

UNIVERSIDAD NACIONAL PEDRO RUÍZ GALLO
FACULTAD DE INGENIERÍA CIVIL, SISTEMAS Y
ARQUITECTURA

ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS



UNPRG | UNIVERSIDAD NACIONAL
PEDRO RUIZ GALLO

**APLICACIÓN MÓVIL UTILIZANDO LA PLATAFORMA
ANDROID PARA LA PLANIFICACIÓN Y VERIFICACIÓN
DEL CUMPLIMIENTO DE HISTORIAS DE USUARIO
BASADO EN LA METODOLOGÍA SCRUM.**

**PROYECTO DE TESIS PARA OPTAR EL TÍTULO
PROFESIONAL DE INGENIERO DE SISTEMAS**

PRESENTADO POR:

KETTY ADALÍ LIVAQUE DELGADO

ERICK JEAN PIERRE BERNILLA MIO

ASESOR:

Dr. Ing. ERNESTO KARLO CELI ARÉVALO

LAMBAYEQUE – PERÚ

2018

**“APLICACIÓN MÓVIL UTILIZANDO LA PLATAFORMA ANDROID PARA LA
PLANIFICACIÓN Y VERIFICACIÓN DEL CUMPLIMIENTO DE HISTORIAS DE
USUARIO BASADO EN LA METODOLOGÍA SCRUM”**

Tesis presentada a la Facultad de Ingeniería Civil, de Sistemas y Arquitectura de la Universidad Nacional “Pedro Ruiz Gallo” para optar el Título Profesional de **INGENIERO DE SISTEMAS** a cargo de los bachilleres:

Ketty Adalí Livaque Delgado

Erick Jean Pierre Bernilla Mío

ASESOR:

Dr. Ing. Ernesto Karlo Celi Arévalo

APROBADA POR:

Ing. Omar Wilton Saavedra Salazar

PRESIDENTE

Mg. Ing. Gilberto Martín Ampuero Pasco

MIEMBRO DEL JURADO

Ing. José Ramón Sandoval Jiménez

MIEMBRO DEL JURADO

LAMBAYEQUE – PERÚ

2018

TESIS:

“APLICACIÓN MÓVIL UTILIZANDO LA PLATAFORMA ANDROID PARA LA PLANIFICACIÓN Y VERIFICACIÓN DEL CUMPLIMIENTO DE HISTORIAS DE USUARIO BASADO EN LA METODOLOGÍA SCRUM”

Ing. Omar Wilton Saavedra Salazar
PRESIDENTE

Mg. Ing. Gilberto Martín Ampuero Pasco
MIEMBRO DEL JURADO

Ing. José Ramón Sandoval Jiménez
MIEMBRO DEL JURADO

Dr. Ing. Ernesto Karlo Celi Arévalo
ASESOR

Bach. Ketty Adalí Livaque Delgado
TESISTA

Bach. Erick Jean Pierre Bernilla Mío
TESISTA

LAMBAYEQUE – PERÚ

2018

DEDICATORIA

Al creador de todas las cosas, por darme la fortaleza necesaria para seguir adelante, por permitirme el haber llegado a este momento tan importante de mi formación profesional y lograr mis objetivos.

A Dios por darme la vida. A mis queridos padres Segundo y Susana por apoyarme en este largo camino de la vida. A mis abuelos quienes estuvieron en todo momento a mi lado dándome consejos. y a mi familia quienes han ido velando por mí durante este arduo camino para convertirme en un profesional.

Erick Jean Pierre Bernilla Mío

A Dios por regalarme la vida. A mi madre Flor por ser una madre ejemplar, siendo el pilar más importante en mi vida. A mi padre Salomón y mi hermano Lening que compartieron conmigo buenos y malos momentos. A toda mi familia por el afecto y apoyo incondicional.

Ketty Adalí Livaque Delgado

AGRADECIMIENTO

Agradecemos a Dios por habernos dado las fuerzas y la protección para seguir adelante superando obstáculos y dificultades a lo largo de toda nuestra vida.

A nuestras familias impulsarnos a ser mejores dándonos una profesión y cultivando en nosotros valores para seguir siendo personas de bien y útiles para la sociedad, gracias por todos sus consejos, comprensión y su apoyo incondicional, ya que ustedes hicieron posible alcanzar esta meta.

Agradecemos de manera muy especial a nuestro asesor, el Ing. Ernesto Karlo Celi Arévalo, por habernos dado su confianza, tiempo, apoyo, paciencia, dedicación en todo momento y también por compartir sus concomimientos, siendo parte fundamental en nuestro desarrollo profesional.

A nuestro jurado de tesis, porque desde un primer momento mostraron disposición a ayudarnos con sus sugerencias y correcciones durante la elaboración de la presente tesis.

A nuestra alma mater la Universidad Nacional “Pedro Ruiz Gallo”, a la Facultad de Ingeniería Civil, Sistemas y Arquitectura, en especial a todos los docentes, por habernos formado profesionalmente para ser personas de bien que aporten a la sociedad.

También queremos agradecer a nuestros amigos y todas las personas que de una u otra forma nos apoyaron, para que sea posible la elaboración de la presente tesis.

CONTENIDO

DEDICATORIA	i
AGRADECIMIENTO	ii
CONTENIDO	iii
ÍNDICE DE GRÁFICOS	vi
ÍNDICE DE TABLAS	vii
RESUMEN	viii
ABSTRACT.....	ix
INTRODUCCIÓN	x
CAPÍTULO I. ASPECTOS DE LA INVESTIGACIÓN.....	4
1.1. DESCRIPCIÓN DE LA REALIDAD PROBLEMÁTICA	4
1.2. PLANTEAMIENTO DEL PROBLEMA.....	6
1.3. DESCRIPCIÓN DEL PROYECTO	6
1.4. OBJETIVOS.....	7
1.5. JUSTIFICACIÓN E IMPORTANCIA	8
1.6. ALCANCE Y LIMITACIONES.....	8
CAPÍTULO II. MARCO TEÓRICO	9
2.1. ANTECEDENTES.....	9
2.2. FUNDAMENTO TEÓRICO.....	13
2.2.1. Aplicación móvil en plataforma Android	13
2.2.2. Modelo de Calidad (ISO / IEC 25010)	16
2.2.3. Modelos de procesos para desarrollo de aplicaciones móviles.....	17
2.2.4. Metodología SCRUM	24
2.2.5. Roles según SCRUM	25
2.2.6. Planificación del Backlog	28
2.2.7. Historias de usuario.....	29
2.2.8. Criterios de aceptación.....	32
2.2.9. Planificación un Sprint.....	35
2.2.10. Segregación de historias de usuario en tareas	36
2.3. GLOSARIO DE TÉRMINOS	37
CAPÍTULO III. DESARROLLO DE LA PROPUESTA.....	40
3.1. PLANIFICACIÓN Y VERIFICACIÓN DE HISTORIAS DE USUARIO.....	40
3.1.1. Preparación de un proyecto SCRUM.....	40
3.1.2. Backlog del producto	41

3.1.3.	Planificación de Sprints	45
3.1.4.	Verificación del cumplimiento de los criterios de aceptación	47
3.2.	CONSTRUCCIÓN	48
3.2.1.	Definición de especificaciones	48
3.2.2.	Diseño conceptual:.....	52
3.2.3.	Desarrollo del prototipo:	56
3.2.4.	Pruebas del usuario:	64
3.2.5.	Implantación:	69
3.2.6.	Auditoría y seguimiento.....	71
3.4.	ESPECIFICACIONES TÉCNICAS DE LA APLICACIÓN MÓVIL	82
3.3.1.	Entorno de desarrollo.....	82
3.3.2.	Lenguaje en el que se desarrolló	82
3.3.3.	Base de datos.....	82
3.4.	DESCRIPCIÓN GENERAL DE LAS EMPRESAS DE CHICLAYO	82
3.4.1.	Empresas de Chiclayo.....	83
3.4.2.	Rubro.....	83
3.4.3.	Metodología de desarrollo de software.....	83
3.4.4.	Proyectos.....	83
	Proyecto 001: Toma de pedidos móvil	84
	Proyecto 002: Proyecto – Venta De Clínica De Nutrición.....	99
3.5.	IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL.....	128
3.5.1.	Desarrollo de proyectos	128
3.5.2.	Planificación de proyectos	131
3.5.3.	Generación de Historias de Usuario	133
3.5.4.	Planificación de Sprints	136
3.5.5.	Desglose de Historias de Usuarios en tareas.....	139
3.5.6.	Verificación del cumplimiento de historias de usuario.....	143
3.5.7.	Monitoreo de planificación y verificación	147
3.5.8.	Reportes y resultados	148
3.6.	FICHA TÉCNICA.....	149
	CAPÍTULO IV. CONTRASTACIÓN DE HIPÓTESIS	152
4.1.	HIPÓTESIS	152
4.2.	IDENTIFICACIÓN DE VARIABLES	152
4.2.1.	Variables	152
4.2.2.	Cuadro de operacionalización de variables.....	152

4.3.	DISEÑO DE CONTRASTACIÓN DE HIPÓTESIS	154
4.4.	POBLACIÓN Y MUESTRA DE ESTUDIO.....	155
4.5.	TÉCNICAS DE RECOPIACIÓN DE DATOS	156
4.5.1.	Técnicas e instrumentos de recolección de datos	156
4.6.	TRATAMIENTO DE LOS DATOS Y DISCUSIÓN DE RESULTADOS	159
4.6.1.	Fiabilidad del instrumento (Encuesta)	159
4.6.2.	Análisis de la regresión múltiple.....	160
CAPÍTULO V. RESULTADOS Y DISCUSIÓN.....		167
CONCLUSIONES Y RECOMENDACIONES		168
RECOMENDACIONES:.....		170
REFERENCIAS BIBLIOGRÁFICAS.....		171
ANEXOS		173
Anexo 01: Formato de Encuestas.		174
Anexo 02: Documento de recolección de datos sobre empresas de desarrollo		176
Anexo 03: Historias de usuario no completadas de acuerdo con el Product Owner.		177
Anexo 04: Reportes de la Aplicación		178
Anexo 05: Simulador y programa usados para el desarrollo de los prototipos		181
Anexo 06: Primer diagrama de secuencias de prototipos		183

ÍNDICE DE GRÁFICOS

<i>Ilustración 1: Arquitectura de Android.</i>	14
<i>Ilustración 2: Modelo de calidad del producto de software.</i>	17
<i>Ilustración 3: Modelo en cascada.</i>	18
<i>Ilustración 4: Proceso de la programación extrema.</i>	19
<i>Ilustración 5: El paradigma de hacer prototipos</i>	20
<i>Ilustración 6: Comparación de metodologías para desarrollo de aplicaciones Android.</i>	20
<i>Ilustración 7: Fases de la Metodología de Prototipado Rápido (MPR)</i>	22
<i>Ilustración 8: Representación de procesos SCRUM.</i>	24
<i>Ilustración 9: Framework Scrum.</i>	25
<i>Ilustración 10: Product Owner.</i>	26
<i>Ilustración 11: Equipo de desarrollo.</i>	27
<i>Ilustración 12: Scrum Master</i>	28
<i>Ilustración 13: Product Backlog.</i>	29
<i>Ilustración 14: Plantilla de historia de usuario y tarjeta.</i>	31
<i>Ilustración 15: Condiciones de satisfacción de la historia del usuario.</i>	33
<i>Ilustración 16: Planificación del Sprint.</i>	35
<i>Ilustración 17: Pila del Sprint detallada en tareas.</i>	36
<i>Ilustración 18: Estructura del Product Backlog.</i>	41
<i>Ilustración 19: Historias de usuario ordenadas por prioridad en los Sprints.</i>	45
<i>Ilustración 20: Historias de usuario segregadas en tareas.</i>	46
<i>Ilustración 21: Base de datos para la construcción de la aplicación móvil.</i>	52
<i>Ilustración 22: Diagrama de procesos en SCRUM</i>	53
<i>Ilustración 23: Diagrama de Verificación de Historias de Usuario</i>	54
<i>Ilustración 24: Diagrama de verificación de Historias de Usuario</i>	55
<i>Ilustración 25: Gráfica de la distribución para arquitectura de solución.</i>	69
<i>Ilustración 26: Diagrama de la Web Service</i>	70
<i>Ilustración 27: Modelo conceptual de la investigación.</i>	152

ÍNDICE DE TABLAS

Tabla 1 Innovación y tecnología para la productividad en empresas.	9
Tabla 2 Tecnología y sistemas de información.....	9
Tabla 3 Desarrollo de software en Latinoamérica	10
Tabla 4 Metodologías ágiles para proyectos	10
Tabla 5 Scrum, metodología ágil más usada.....	11
Tabla 6: Metodologías para desarrollo de aplicaciones móviles en android.	11
Tabla 7: Análisis de la metodología de prototipado rápido.....	12
Tabla 8: Desarrollo de un sistema multimedia con la metodología de prototipado.	12
Tabla 9: Comparación entre metodologías para desarrollo de aplicaciones móviles	21
Tabla 10: Ejemplo Historias de Usuario de un proyecto.....	43
Tabla 11: Ejemplo Historias de Usuario de un proyecto.....	46
Tabla 12: Ejemplo del detalle de las tareas.	46
Tabla 13: Ejemplo del detalle de las tareas.	47
Tabla 14: Resumen de requerimientos funcionales	49
Tabla 15: Diseño de prueba de prototipos.	64
Tabla 16: Prueba de prototipos y observaciones	65
Tabla 17: Descripción del plan de auditoría.....	71
Tabla 18: Auditoría y seguimiento al desarrollo de la aplicación móvil.....	72
Tabla 19: Lista inicial de historias de usuario y criterios de aceptación – Caso 01	85
Tabla 20: Priorización de historias de usuario	89
Tabla 21: Ordenamiento de Historias por Prioridad – Caso 01.....	93
Tabla 14: Primer Sprint – Caso 01.....	94
Tabla 15: Segundo Sprint – Caso 01.....	94
Tabla 26: Desglose de HU en tareas-Caso 01	95
Tabla 27: Porcentaje de avance de tareas – Caso 01.....	96
Tabla 28: Verificación de Criterios de aceptación - Caso 01	98
Tabla 29: Lista inicial de HU y Criterios de aceptación - Caso 02	100
Tabla 30: Priorización de HU por prioridad – Caso 02.....	104
Tabla 31: Orden de HU por prioridad - Caso 02.....	108
Tabla 32: Primer Sprint - Caso 02.....	110
Tabla 33: Segundo Sprint - Caso 02	111
Tabla 34: Tercer Sprint - Caso 02	111
Tabla 35: Cuarto Sprint - Caso 02	111
Tabla 36: Desglose de HU en tareas - Caso 02	112
Tabla 37: Porcentaje de avance de tareas - Caso 02	114
Tabla 38: Verificación de criterios de aceptación - Caso 02.....	118
Tabla 39: Sprint a evaluar - Caso 02.....	122
Tabla 40: Sprint 03 modificado - Caso 02	123
Tabla 41: Verificación de criterios de aceptación - Caso 02.....	124
Tabla 42: Tabla de indicadores.	153
Tabla 43 Trabajadores de Upson Systems – Chiclayo.....	155
Tabla 44 Trabajadores de Despensa peruana S.A. – Chiclayo.....	156
Tabla 45 Total de trabajadores encuestados.....	156
Tabla 46: Matriz de consistencia entre los indicadores y las preguntas de la encuesta	157
Tabla 47 Matriz de reducción de ítems evaluados.....	161

RESUMEN

En el presente proyecto se desarrolló una aplicación móvil que facilite la construcción de un producto de software en empresas locales, dedicadas al desarrollo de software basado en la metodología SCRUM, específicamente para apoyar en la planificación y verificación de historias de usuario, que son fundamentales en todo proyecto de desarrollo de software bajo este enfoque, y que son fundamentales para un mejor control del desarrollo de los proyectos.

El tipo de investigación que se desarrolló en el presente estudio es correlacional porque se estableció la relación entre las variables: Aplicación móvil desarrollada en plataforma Android y la Planificación y verificación de historias de usuario basado en la metodología SCRUM. sin embargo, también se realizó una serie de descripciones como: La metodología SCRUM, los roles, los procesos de planificación del Backlog, Sprint, historias de usuario, tareas, listas de verificación, etc.

Para recopilar la información se aplicó una serie de técnicas e instrumentos para la recolección de datos, como entrevistas, análisis de casos, observación, etc. Y para la contrastación de la hipótesis se aplicó una encuesta.

El aporte práctico de la investigación fue el desarrollo de una aplicación móvil para plataforma Android que gestiona las historias de los usuarios bajo la perspectiva Scrum. Adicionalmente, para el desarrollo de la aplicación se usaron diversas tecnologías como el lenguaje de programación Java, Android Studio como ID de desarrollo, sistema de gestor de base de datos PostGres, SQLite.

De esta manera concluimos que, con el desarrollo de la aplicación móvil, su implementación y prueba en empresas de la región dedicadas al desarrollo de software, apoya en el registro y control de la planificación y verificación de historias de usuario, para la gestión de proyectos bajo la guía de la metodología SCRUM.

PALABRAS CLAVES: Metodología SCRUM, Aplicación móvil, Historias de Usuario, Planificación de historias, criterios de aceptación, Verificación.

ABSTRACT

In the present project, a mobile application was developed to facilitate the construction of a software product in local companies, dedicated to the development of software based on the SCRUM methodology, specifically to support the planning and verification of user stories, which are fundamental in all software development projects under this approach, and which are fundamental for a better control of the development of the projects.

The type of research that was developed in the present study is correlational because the relationship between the variables was established: Mobile application developed on Android platform and Planning and verification of user stories based on the SCRUM methodology. however, a series of descriptions was also made such as: The SCRUM methodology, the roles, the Backlog planning processes, Sprint, user stories, tasks, checklists, etc.

To collect the information, a series of techniques and instruments were applied to collect data, such as interviews, case analysis, observation, etc. And for the test of the hypothesis a survey was applied.

The practical contribution of the research was the development of a mobile application for Android platform that manages the stories of users under the Scrum perspective. Additionally, for the development of the application various technologies were used such as the Java programming language, Android Studio as a development ID, PostGres database manager system, SQLite.

In this way we conclude that, with the development of the mobile application, its implementation and testing in companies in the region dedicated to software development, it supports the registration and control of planning and verification of user stories, for project management under the guidance of the SCRUM methodology.

KEYWORDS: SCRUM Methodology, Mobile Application, User Stories, Story Planning, Acceptance Criteria, Verification.

INTRODUCCIÓN

Las TIC's y Los sistemas de información han sido bases fundamentales para el desarrollo de un gran número de empresas y han sido aplicadas en diversos rubros como: Ingeniería, Informática, educación, industria, medicina, etc. Es así como surge la necesidad de tener un área dedicada al desarrollo tecnológico para automatizar procesos base como logística, operaciones, servicios, etc.

Dentro de este desarrollo tecnológico tenemos proyectos de desarrollo de software, hay empresas grandes que, dentro de su estructura, cuentan con un área especializada en crear productos de software adecuado a sus necesidades, por otro lado, tenemos algunas empresas pequeñas que no son capaces de tener esta área de tecnología por lo que tienen que tercerizar este servicio, en vista a esta necesidad nace en el mercado empresas dedicadas exclusivamente al desarrollo de proyectos de software.

Las empresas de desarrollo de software tienen varias metodologías con las que gestionan proyectos de software, metodologías tradicionales como RUP (Rational Unified Procces), para proyectos estáticos y poco cambiantes, pero para empresas que necesitan mejorar la capacidad de respuesta al cambio en sus productos de software, tenemos las metodologías ágiles como SCRUM, que es adaptable con entregables funcionales de software.

La metodología Scrum para el desarrollo ágil de software es un marco de trabajo diseñado para lograr la colaboración eficaz de equipos en proyectos, que emplea un conjunto de reglas y artefactos y define roles que generan la estructura necesaria para su correcto funcionamiento., los llamados Equipos Scrum son auto - gestionados, multifuncionales y trabajan en iteraciones. La autogestión les permite elegir la mejor forma de hacer el trabajo, en vez de tener que seguir lineamientos de personas que no pertenecen al equipo y carecen de contexto. Navarro Cadavid, Fernández Martínez, & Morales Vélez (2013). Dentro de Scrum tenemos varios procesos como la pila del producto, planificación del Sprint, reuniones diarias, monitoreo, pruebas, etc.

El presente documento describe el desarrollo de una aplicación en plataforma Android, cuya principal finalidad y objetivo de este proyecto ha sido la creación de una aplicación móvil que sea capaz de proporcionar toda la información necesaria para planificar y verificar historias de usuario basado en la metodología SCRUM. Describiremos el problema, y detallaremos las razones para empezar a optar por esta aplicación móvil para el aprovechamiento de este. A lo

largo de los próximos capítulos iremos mostrando, por capítulos, los pasos necesarios para para la elaboración de la solución que proponemos.

En el capítulo I, mostraremos generalidades y el problema como punto de partida de este proyecto, además de describir los objetivos generales y específicos, la justificación e importancia, las limitaciones y alcances del proyecto.

En el capítulo II, mostraremos los conceptos básicos y definición de términos que nos ayuden a entender la base teórica para los antecedentes, fundamento teórico y daremos una descripción detallada para entender términos usados en el proyecto.

Capítulo III, explicaremos de que trata nuestro proyecto, los pasos a seguir para su construcción, describiremos especificaciones básicas, para ver cómo se aplicó, cómo se realizaron las pruebas de la aplicación móvil, para posteriormente presentar una explicación detallada del funcionamiento del producto final.

Capítulo IV, presentaremos la hipótesis, describiendo el tipo de investigación que usaremos, para posteriormente identificar las variables dependientes e independientes, así logramos presentar una contrastación de hipótesis. Explicaremos el tipo de población y muestra que escogimos junto con las técnicas de recopilación de información para probar nuestra hipótesis inicial.

Capítulo V, exploraremos en los datos obtenidos para analizar los estadísticos con los resultados obtenidos, viendo el cumplimiento de las variables y dimensiones propuestas inicialmente.

Finalmente, presentaremos las conclusiones y recomendaciones para un posterior trabajo, basándonos en lo aprendido durante el desarrollo de nuestro proyecto.

CAPÍTULO I. ASPECTOS DE LA INVESTIGACIÓN

1.1. DESCRIPCIÓN DE LA REALIDAD PROBLEMÁTICA

El dinamismo y la globalización en el mundo han hecho que las empresas busquen innovar sus productos y/o servicios con el fin de ser competitivos, tener una posición en el mercado y generar beneficios.

Es así, que cada día la generación de riqueza, la creación de empleo calificado, la mejora de la productividad, la estabilidad económica y el desarrollo de nuevos servicios, se deben cada vez más, a la contribución de las nuevas tecnologías y los servicios de telecomunicaciones en las diversas esferas de la sociedad y la economía (Instituto Nacional de Estadística e Informática, 2002).

La infraestructura de Tecnología de Información brinda las herramientas necesarias para el crecimiento y desarrollo de Sistemas de Información eficaces que procesen información y generen conocimiento para solucionar problemas específicos, ayude a tomar decisiones, gestione recursos, apoye a la creación de nuevos productos o servicios, minimice riesgos, controle procesos críticos y genere valor al negocio.

Los sistemas de Información (SI) y la Tecnología de Información y Comunicación (TIC), han permitido el posicionamiento de la Industria de software en todos los ámbitos de la sociedad. Al punto de representar una actividad económica captando la atención de muchas empresas, llevando así, a la formación de organizaciones encargadas de desarrollar software.

Debido al estado de la industria de software mundial, han surgido organizaciones que definen métodos, técnicas y prácticas asociadas al proceso de desarrollo de software y contribuyen al crecimiento continuo de esta industria, tal es el caso del Instituto de Ingeniería de Software de los Estados Unidos–SEI (Software Engineering Institute), el Instituto Europeo de Software –ESI (European Software Institute), entre los más representativos en este ámbito. En los países latinoamericanos la industria de software, aunque ya se ha empezado un proceso de maduración, las empresas del sector se enfrentan a una indudable desventaja, la falta de competitividad, en precios y calidad (Scarone, 2005). Para monitorear el desarrollo de productos de necesitan seguir procesos, usar herramientas, técnicas, pasos, etc. Es así, que se usaron metodologías existentes en distintas áreas de conocimientos, que con el pasar del tiempo se ajustaron

al desarrollo de software y que fueron llamadas metodologías tradicionales. Inicialmente funcionaron como guía para desarrollar aplicaciones informáticas que daban soporte a procesos de negocios poco cambiantes, en relación con sus políticas, procedimientos, etc. Llevando una documentación exhaustiva, y cuyo plan de proyecto se define en la fase inicial del desarrollo, impidiendo futuros cambios al producto final volviéndose compleja para entornos variables.

Luego de discrepancias en cuanto a los beneficios y desventajas de las metodologías tradicionales, nació un nuevo enfoque denominado Métodos Ágiles, que cubre las deficiencias descritas anteriormente.

Ante esta situación, se reunieron personas experimentadas en temas relacionados al desarrollo de software, donde proponen soluciones descritas en un documento llamado “Manifiesto por el desarrollo ágil de software”, siendo idóneo para cambios en el producto de software, manteniendo la satisfacción del cliente, transformando un proyecto grande en partes más pequeñas, haciendo planificación a corto plazo lo que nos permite tener software funcional en tiempos relativamente cortos y generando al cliente ventaja competitiva.

“Una primera selección surge del manifiesto: Scrum, Extreme Programming [XP], Dynamic System Development Method [DSDM], Crystal, Adaptive Software Development [ASD] y Feature-Driven Development [FDD], están representadas en él, a través de al menos una de las personas que lo suscribieron” (Navarro Cadavid, Fernández Martínez, & Morales Vélez, 2013).

Dentro de este grupo vemos a Scrum como la metodología más usada con un 59%, según una encuesta hecha en el 2015” VersionOne (2015).

La metodología Scrum es ágil, liviana, iterativa e incremental, fácil de entender y difícil de aplicar al inicio de su implantación, ya que es un cambio en el pensamiento de todos los trabajadores de empresas desarrolladoras de software, pero que genera beneficios a corto plazo.

En el afán de gestionar los procesos de Scrum de una manera más eficiente, se han creado distintas aplicaciones móviles como Trello, Agile Knowledge tree, Agile Tasks, Asana, que apoyan a las empresas en el desarrollo de software.

Estas aplicaciones móviles solo cubren una parte de las necesidades de las empresas de desarrollo de software al usar Scrum, se enfocan más en el “Modelo Kanban” (sistema de tarjetas que ayudan a tener un mejor control de las tareas asignadas) siendo una técnica visual que permite ver el estado de los proyectos, así como pautar el desarrollo del trabajo de manera efectiva.

Hasta la fecha no hay una aplicación móvil que integre la planificación, desglose en tareas y verificación de historias de usuarios, ante esto se desarrolló una aplicación que permite relacionar ambas partes.

El presente estudio se enfocó en el análisis de la metodología ágil Scrum, para el desarrollo de una aplicación móvil que ayude a las empresas que desarrollan software y toman a Scrum como metodología, a gestionar la planificación y verificación de los productos de software.

1.2. PLANTEAMIENTO DEL PROBLEMA

¿El desarrollo de una aplicación móvil en plataforma Android, basado en SCRUM, permite desarrollar las actividades de planificación y verificación de historias de usuario en proyectos de desarrollo de software?

1.3. DESCRIPCIÓN DEL PROYECTO

SCRUM es una metodología ágil, que permite entregables funcionales de productos de software generando un beneficio enorme para los empresarios que solicitan un producto en tiempos cortos.

Para lograr el adecuado funcionamiento de SCRUM en proyectos de desarrollo de software, existe una guía de procesos a seguir, entre ellos está la planificación de la pila de producto, la creación de sprint, la estimación de tiempos para realización del desarrollo, las reuniones diarias, las pruebas y la entrega final del producto.

La planificación es un pilar fundamental en el inicio de todo tipo de proyectos y los proyectos de software no están exceptos de principio, existen diversas herramientas, que ayudan a planificar, pero Scrum nos sugiere implementar dentro de la planificación gestionar y describir historias de usuario, permitiendo al cliente un acercamiento con la evolución del desarrollo de su producto final.

Este proyecto facilitó el manejo de la información resultante del planning meeting en el proceso de la planificación en SCRUM, desglosando en historias de usuario, con sus respectivos criterios de aceptación, para posteriormente presentarlos en tareas para ser trabajados por el equipo de desarrollo, y al mismo tiempo permitir al dueño del producto, una vez finalizado el proyecto, verificar que sus requerimientos iniciales fueron cumplidos de acuerdo a sus necesidades empresariales.

La necesidad de planificar y verificar las historias de usuario se evidencia debido a la dificultad de administrar información de las tareas descritas en la planificación, y la necesidad del dueño del producto por comprobar, que su producto de software es tal cual se especificó en un inicio. En la actualidad no hay una aplicación móvil que permita tener al alcance un registro adecuado de la información para planificar y verificar historias de usuario, dentro de la metodología SCRUM.

El desarrollo de esta aplicación móvil en Android, permitirá que muchas empresas que desarrollan software puedan trabajar sus proyectos, siguiendo la metodología SCRUM, maneje su información de una forma más ordenada, sin necesidad de estar en la oficina para ver la planificación inicial, sino que pueda ver la información en su dispositivo móvil cada vez que sea necesario, además de que el dueño del producto sea capaz de ver y darle la aceptación u observación al producto de software que recibe.

1.4. OBJETIVOS

Objetivo general

Desarrollar una aplicación móvil en plataforma Android, basado en SCRUM, permite desarrollar las actividades de planificación y verificación de historias de usuario en proyectos de desarrollo de software.

Objetivo específico

- Evaluar la funcionalidad de la aplicación Android con relación al cumplimiento de las actividades de planificación y verificación.
- Determinar la portabilidad de la aplicación Android a plataformas distintas en cuanto a versiones del Sistema Operativo y equipos.
- Determinar el nivel de usabilidad de la aplicación por parte de los usuarios que planifican proyectos de software.
- Evaluar la trazabilidad de las acciones realizadas en la aplicación.

1.5. JUSTIFICACIÓN E IMPORTANCIA

La presente tesis se justifica, desde las siguientes perspectivas:

Relevancia social

La presente investigación, tiene dos beneficiarios directos, por un lado, tenemos al equipo de desarrollo de software, que tendrá una estimación fiable reduciendo, errores, demoras, costos de producción y tiempo de desarrollo; por otro lado, el otro beneficiario es el cliente, que tendrá un producto de software funcional lo más cercano posible, a lo que inicialmente especificó, ahorrando tiempo, y costos.

Aporte práctico

El aporte práctico de esta investigación es el desarrollo de una aplicación móvil en plataforma Android, basado en SCRUM, permite desarrollar las actividades de planificación y verificación de historias de usuario en proyectos de desarrollo de software.

1.6. ALCANCE Y LIMITACIONES

Como parte de la investigación, se tomarán en cuenta los siguientes puntos:

- Debido a la que no hay un estudio concreto sobre las empresas que desarrollan software bajo la metodología SCRUM en la provincia de Chiclayo, se procederá a seleccionar empresas conocidas que cumpla con los requisitos básicos de la propuesta definida y que a través de una encuesta se generalicen los resultados.
- La información vital para el desarrollo de la aplicación y su uso en proyectos Scrum, es muy escasa o nula en el ámbito de desarrollo de software, razón por la cual nos basaremos en la información proporcionada por la empresa Upson Systems S.A. para trabajar los casos de estudio.
- La aplicación móvil, aún no está disponible en Google Play ya que se espera la aprobación correspondiente de la presente investigación.
- El servicio web que alimentan a la aplicación para su correcto funcionamiento está alojado en un servidor propio.
- El dispositivo donde se instale la aplicación debe contar con acceso a internet para su correcto funcionamiento.

CAPÍTULO II. MARCO TEÓRICO

2.1. ANTECEDENTES

Para la realización de este proyecto se tomó como referencia las siguientes fuentes de información.

Tabla 1: Innovación y tecnología para la productividad en empresas.

Título	La innovación en la empresa: la orientación al mercado como factor de éxito en el proceso de innovación en producto.
Universidad	University Oberta de Catalunya
Fecha	2005
Autor(es)	Carlos A. Scarone Delgado
Resumen	Logro determinar que la innovación es propia y exclusiva de sectores de alto contenido tecnológico, repercutiendo en todos los sectores de actividades.
Análisis de relación con la presente investigación	Este trabajo de investigación se toma como referencia porque explica que la innovación y la Tecnología van de la mano, mejorando la productividad en las empresas. Tiene relación con nuestra tesis como parte inicial de donde surge la necesidad de implantar las TIC en las empresas.

Fuente: Creación propia

Tabla 2: Tecnología y sistemas de información

Título	Sistemas de Información Gerencial
Universidad	-
Fecha	2012
Autor(es)	Kenneth C. Laudon y Jane P. Laudon.
Resumen	Describe como las empresas contemporáneas usan las Tecnologías y los sistemas de información para crecer y lograr sus metas.
Análisis de relación con la presente investigación	Este trabajo de investigación guarda relación con nuestra tesis porque la Tecnología y los Sistemas de información son la base para el desarrollo de software en el cual nos centraremos.

Fuente: Creación propia

Tabla 3: Desarrollo de software en Latinoamérica

Título	Gestión de proyectos de Software
Universidad	Iniciativa Latinoamericana de Libros de Texto Abiertos
Fecha	2014
Autor(es)	Francisco Javier Álvarez Julio Ariel Hurtado Alegría Margarita Mondragón Arellano Jaime Muñoz Arteaga César Eduardo Velázquez Amador Yosly Caridad Hernández Bieliukas
Resumen	Se centra en la industria del desarrollo de software en Latinoamérica, brinda modelos de procesos de software y su uso dentro de la ingeniería de Software.
Análisis de relación con la presente investigación	Este trabajo de investigación guarda relación con nuestra tesis porque se va a desarrollar un software, analizar su uso y plantear una solución.

Fuente: Creación propia

Tabla 4: Metodologías ágiles para proyectos

Título	Revisión de metodologías ágiles para el desarrollo de software
Universidad	Universidad Icesi
Fecha	2013
Autor(es)	Andrés Navarro Cadavid, Juan Daniel Fernández Martínez, Jonathan Morales Vélez
Resumen	Nos da una descripción y un enfoque sobre las metodologías ágiles, como solución al cambio de requerimientos por parte de los clientes, siendo su participación más notoria en el proceso de desarrollo, Propone estrategia de solución como: tener múltiples dueños de historias de usuario. Asevera que la administración de proyectos es donde más enfatiza Scrum.
Análisis de relación con la presente investigación	Este trabajo de investigación guarda relación con nuestra tesis dado que nos enfocaremos en la metodología ágil Scrum, específicamente en la participación del cliente y la creación de historias de usuarios.

Fuente: Creación propia

Tabla 5: Scrum, metodología ágil más usada.

Título	10 th Annual State of agile Report
Universidad	Encuestadora Internacional.
Fecha	2015
Autor(es)	VersionOne
Resumen	Es una compañía de software empresarial dedicada a la gestión del ciclo de vida ágil comprometida con el cliente, planificación y el seguimiento de la integración de código fuente, ayudando a las organizaciones escalar sus iniciativas ágiles más rápido, más fácil y más inteligente.
Análisis de relación con la presente investigación	Este trabajo de investigación guarda relación con nuestra tesis, puesto que nos centraremos específicamente en lo que es Scrum, permitiendo justificar por qué fijarnos en esta metodología siendo la más usada a nivel mundial, y nos permite tomar.
Fuente: Creación propia	

Tabla 6: Metodologías para desarrollo de aplicaciones móviles en android.

Título	Propuesta Metodológica para Desarrollo de Aplicaciones Móviles para Dispositivos Android
Universidad	Universidad de la Sierra, México
Fecha	2014
Autor(es)	Ulises Ponce Mendoza, Soto Bernal Rafael Armando y Yáñez Moreno Víctor.
Resumen	Este trabajo de investigación realizó un análisis de las características de trabajo de la empresa modelo con las de varias metodologías que permitiera hacer una transición sencilla y ágil para el equipo de desarrollo, al mismo tiempo que se empataba con prácticas recomendadas de desarrollo.
Análisis de relación con la presente investigación	Este trabajo de investigación guarda relación con nuestra tesis, puesto que nos guiaremos de este análisis de distintas metodologías para escoger la más conveniente para el desarrollo de nuestra aplicación móvil.
Fuente: Creación propia	

Tabla 7: Análisis de la metodología de prototipado rápido

Título	Análisis Comparativo de la Metodología de Prototipado Rápido de Aplicaciones (RAP) para Desarrollo de SW en Móviles.
Universidad	Universidad de la Sierra, México
Fecha	2015
Autor(es)	Ulises Ponce Mendoza, Jesús Miguel García Gorrostieta, José David Madrid Monteverde y Aldo José Juárez de Haro.
Resumen	Este trabajo de investigación se centra en el análisis de la metodología de Prototipado Rápido de Aplicaciones Móviles como una alternativa para asegurar la calidad de aplicaciones desarrolladas
Análisis de relación con la presente investigación	Este trabajo de investigación guarda relación con nuestra tesis, puesto que nos guiaremos de este enfoque a la metodología del prototipado para poder desarrollar nuestra aplicación móvil.
Fuente: Creación propia	

Tabla 8: Desarrollo de un sistema multimedia con la metodología de prototipado.

Título	Prototipo de sistema multimedia interactivo para la visualización del funcionamiento de los algoritmos evolutivos
Universidad	Euro hispanoamericana
Fecha	2008
Autor(es)	José de Jesús Ávalos Gonzáles
Resumen	Desarrolla modelo de prototipo, abarcando en las diferentes iteraciones que se llevaron a cabo para el desarrollo correcto de un sistema iterativo.
Análisis de relación con la presente investigación	Este trabajo de investigación guarda relación con nuestra tesis, puesto que nos ayuda a entender e implementar correctamente la metodología de prototipado que usaremos para el desarrollo de nuestra aplicación móvil.
Fuente: Creación propia	

2.2. FUNDAMENTO TEÓRICO

Para el desarrollo del presente proyecto de tesis, es necesario tener en cuenta los siguientes fundamentos teóricos:

2.2.1. Aplicación móvil en plataforma Android

Aplicación móvil

Según diversos expertos, definen aplicación móvil como:

Según, Gómez Muñoz, Herrera Cárdenas, & Santiago Álvarez (2009), una aplicación, es un tipo de programa informático diseñado para facilitar al usuario la realización de un determinado tipo de trabajo. Esto lo diferencia principalmente de otros tipos de programas como los sistemas operativos (que hacen funcionar al ordenador), las utilidades (que realiza tareas de mantenimiento o de uso general), y los lenguajes de programación (con lo cual se crean programas informáticos), que realizan tareas más avanzadas y no pertinentes al usuario común.

Según, Avilés (2011), es un Software desarrollado para correr bajo Smartphones. Está diseñada para educar, entretener o ayudar en la vida diaria a sus usuarios/consumidores. Estas funcionalidades que antes sólo estaban disponibles en PC ahora pueden ser ejecutadas en los dispositivos móviles (González Mérida, 2012).

Para Rojas Lizarazo, Roa Castañeda, & Alarcón Aldana (2011) Con el auge de los dispositivos móviles el desarrollo de aplicaciones ha avanzado con fines lucrativos, de investigación y de satisfacción de necesidades, entre otros. Las aplicaciones son creadas mediante herramientas y kits de desarrollo específicos para cada plataforma. Por lo general, cada plataforma ofrece un simulador para probar las aplicaciones, sin embargo, la mejor prueba es en el dispositivo real.

Finalmente podemos asegurar que una aplicación móvil es un programa informático desarrollado, para ser ejecutado en equipos móviles (celulares inteligentes, tablets, etc.) y que pueden ser usadas en diversas situaciones por los usuarios y se encuentran disponibles en varias plataformas como Android, iOS, BlackBerry OS, Windows Phone, etc. Cada una de estas ofrece beneficios distintos, pero con un mismo fin la satisfacción del usuario final.

Aplicación nativa

Las aplicaciones nativas es un programa informático que tiene archivos ejecutables que pueden ser descargados desde un repositorio o tienda virtual de aplicaciones como App store de Apple, Marketplace de Android o App World de BlackBerry u otros; se instalan directamente al dispositivo y se ejecuta como otro servicio del dispositivo móvil

Plataforma Android

En informática, una plataforma de desarrollo es el entorno de software común en el cual se desenvuelve la programación de un grupo definido de aplicaciones. Comúnmente se encuentra relacionada directamente a un sistema operativo; sin embargo, también es posible encontrarla ligada a una familia de lenguajes de programación o a una interfaz de programación de aplicaciones (Miguel, 2011).

Android plataforma para aplicaciones móviles, es totalmente abierta tanto los fabricantes de dispositivos móviles y para los desarrolladores de aplicaciones que funcionan en tales dispositivos. Android incluye básicamente un sistema operativo, middleware y aplicaciones base. Inicialmente desarrollado por Google y luego en conjunto con Open Handset Alliance, Android permite a los desarrolladores escribir código gestionado en lenguaje de programación Java y controlar los dispositivos por medio de bibliotecas desarrolladas o adaptadas por Google (Miguel, 2011).

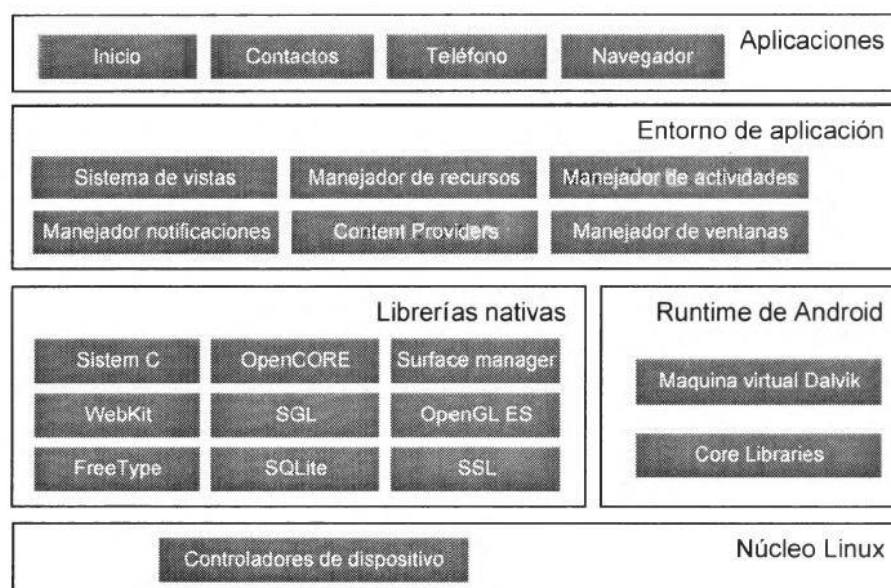


Ilustración 1: Arquitectura de Android.

Fuente: (Tomás Gironés, 2012)

El Núcleo Linux

El núcleo de Android está formado por el sistema operativo Linux, versión 2.6. Esta capa proporciona servicios como la seguridad, el manejo de la memoria, el multiproceso, la pila de protocolos y el soporte de para dispositivos.

Runtime De Android

Está basado en el concepto de máquina virtual utilizado en Java. Google tomo la decisión de crear una nueva, la máquina virtual Dalvik, que respondiera mejor y sea óptima de recursos. También se incluye en el Runtime de Android el “Core libraries” con la mayoría de las librerías disponibles en el lenguaje Java.

Librerías Nativas

Incluye un conjunto de librerías en C/C++ usadas en varios componentes de Android. Están compiladas en el código nativo del procesador. Muchas de las librerías utilizan proyectos de código abierto. Algunas de estas librerías son:

- **System C Library:** una derivación de la librería BSD de C estándar (libc), adaptada para dispositivos embebidos basados en Linux.
- **Media Framework:** librería basada en PacketVideos OpenCORE; soporta codecs de reproducción y grabación de multitud de formatos de audio, video e imágenes MPEG4, H.264, MP3, AAC, AMR, JPG y PNG.
- **Surface Manager:** maneja el acceso al subsistema de representación gráfica en 2D y 3D.
- **WebKit:** soporta un moderno navegador web utilizado en el navegador Android y en la vista webview. Se trata de la misma librería que utiliza Google Chrome y Safari de Apple.
- **SGL:** moto de gráficos 2D.
- **Librerías 3D:** implementación basada en OpenGL ES 1.0 API. Las librerías utilizan el acelerador hardware 3D si está disponible, o el software altamente optimizado de proyección 3D.
- **FreeType:** fuentes en bitmap y renderizado vectorial.

- **SQLite:** potente y ligero motor de bases de datos relacionados disponible para todas las aplicaciones.
- **SSL:** proporciona servicios de encriptación Secure Socket Layer (Tomás Gironés, 2012).

Entornos de desarrollo

Android estudio es un IDE Entorno de desarrollo Integrado apoyado por el SDK (software development kit) que son herramientas de desarrollo de software disponibles en la página oficial de Android, y que ayuda al desarrollador a crear aplicaciones basadas en Android:

- IntelliJ IDE + Estudio plugin de Android
- Android SDK Tools
- Android plataforma de herramientas
- Una versión de la plataforma Android
- Emulador de Android una imagen del sistema Android que incluye servicios de Google Play

Entonces podemos decir que Android además de ser una plataforma es un sistema operativo basado en software libre Linux, diseñado para desarrollo de aplicaciones móviles, usando como lenguaje de programación.

2.2.2. Modelo de Calidad (ISO / IEC 25010) ¹

Existen un conjunto de normas que formaron la base para obtener esta nueva versión que detalla la calidad de los Sistemas Informáticos, productos de Software, calidad de datos y el uso de estos.

Las Normas Internacionales que conforman esta división actuales modelos detallados de calidad para los sistemas informáticos y productos de software, calidad en el uso y datos. También se proporciona una guía práctica sobre el uso de los modelos de calidad.

¹ ISO/IEC: International Organization for Standardization / International Electrotechnical Commission

Modelo de calidad del producto

Este modelo de calidad nos ayudará a evaluar la calidad del producto de software, determinando características importantes del producto al momento de evaluar sus propiedades.

Según la ISO/IEC 25000 (2014): Un modelo de calidad es un conjunto definido de características, y las relaciones entre ellas, lo que proporciona un marco para la especificación de requisitos de calidad y evaluación de la calidad. El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las ocho características de calidad que se muestran en la siguiente figura:

(Sub)Characteristic	Reliability
Functional suitability	Maturity
Functional completeness	Availability
Functional correctness	Fault tolerance
Functional appropriateness	Recoverability
Performance efficiency	Security
Time behaviour	Confidentiality
Resource utilization	Integrity
Capacity	Non-repudiation
Compatibility	Accountability
Co-existence	Authenticity
Interoperability	Maintainability
Usability	Modularity
Appropriateness recognizability	Reusability
Learnability	Analysability
Operability	Modifiability
User error protection	Testability
User interface aesthetics	Portability
Accessibility	Adaptability
	Installability
	Replaceability

Ilustración 2: Modelo de calidad del producto de software

Fuente: (ISO 25010, 2011).

2.2.3. Modelos de procesos para desarrollo de aplicaciones móviles

Tomaremos información de distintos autores para hacer un análisis de metodologías a usar en el desarrollo de nuestra aplicación móvil.

Modelo en cascada:

Según Pressman (2010) El modelo en cascada es llamado a veces ciclo de vida clásico, sugiere un enfoque sistemático, secuencial hacia el desarrollo de software, que inicia con la especiación de requerimientos, seguido por el modelado, la construcción y el despliegue para culminar en el soporte de software terminado.

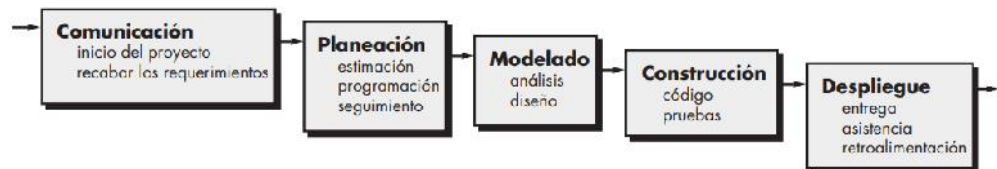


Ilustración 3: Modelo en cascada

Fuente: (Pressman R. S., Ingeniería de Software: Un enfoque práctico 7ma edición, 2010)

Móvil – D

Según (Spataru A. C., 2010), citado por Ponce Mendoza, Madrid Monteverde, García Gorrostieta, & Juárez de Haro (2015) Mobile-D, es una metodología que hace su aparición en 2004 y que incorpora elementos de Programación Extrema, el Proceso Unificado de Rational (RUP) y Metodologías Crystal. Tiene ciclos de desarrollo cortos, es decir realizar entregas de producto cada diez semanas, cuenta un equipo de máximo 10 desarrolladores. Típicamente se implementa en cinco fases las cuales son:

- Exploración
- Inicialización
- Producción
- Estabilización
- Prueba y Corrección de Defectos.

Incluye el uso de nueve elementos que se encuentran directamente relacionados con prácticas de desarrollo de software los cuáles son:

- Definición de Fases y Actividades
- Definición de arquitecturas tipo
- Desarrollo basado en pruebas
- Integración Continua
- Programación en pares

- Métricas
- Mejora de los procesos ágiles
- Fuera del sitio del cliente
- Enfoque centrado en el usuario

XP – Programación Extrema

Según Pressman (2010) La programación extrema usa un enfoque orientado a objetos como paradigma preferido de desarrollo, y engloba un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades estructurales: planeación, diseño, codificación y pruebas.

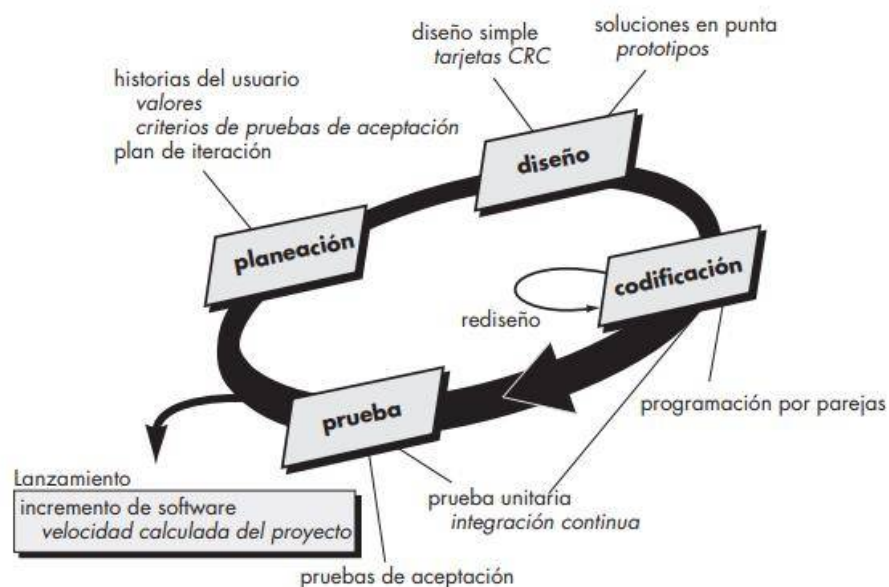


Ilustración 4: Proceso de la programación extrema.

Fuente: (Pressman R. S., Ingeniería de Software: Un enfoque práctico 7ma edición, 2010)

Six - sigma

Según Corral, Sillitti, & Giancarlo, (2013), citado por Ponce Mendoza, Madrid Monteverde, Garcia Gorrostieta, & Juárez de Haro (2015), Scrum con Lean Six Sigma, es una metodología híbrida entre un enfoque de planeación y control estadístico y una metodología ágil como Scrum. Tomando de Scrum elementos como los Sprints. Fue diseñada especialmente para desarrollar aplicaciones empotradas como las que se incorporan por los fabricantes de dispositivos y/o operadores telefónicos.

Modelo de prototipado

Según Pressman (2010) de manera ideal, el prototipo debería servir como un mecanismo para identificar los requisitos de software. A pesar de que la construcción de prototipos se puede utilizar como un modelo de proceso independiente, se emplea más comúnmente como una técnica susceptible de implementarse dentro del contexto de cualquiera de los modelos de procesos.

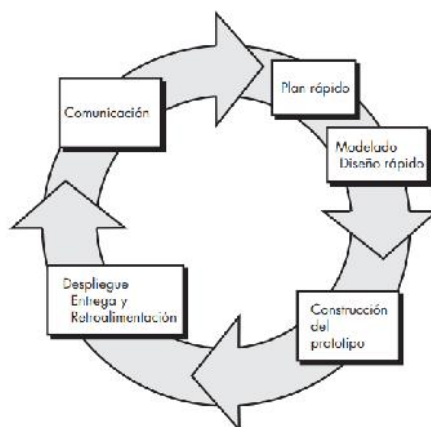


Ilustración 5: El paradigma de hacer prototipos

Fuente: (Pressman R. S., Ingeniería de Software: Un enfoque práctico 7ma edición, 2010)

Según Yáñez Moreno, Ponce Mendoza, & Soto Bernal (2014) Es una metodología iterativa basada en la metodología de prototipos y que toma algunos elementos de Scrum como la definición de historias para cada uno de los casos de uso, la interacción permanente con un representante del cliente y la realización de sprint sucesivos que tienen como objetivo crear prototipos de la aplicación móvil. Propone el uso del paradigma Modelo-Vista-Controlador como modelo arquitectónico para separar las funcionalidades esenciales de las diferentes vistas requeridas por los distintos dispositivos en las fases de diseño, definición de tareas y construcción del prototipo.

Cascada	Scrum	Prototipos
El equipo de desarrollo es mayor de 5 integrantes.	El equipo de desarrollo es mayor a 5 integrantes pero menor a 9.	El equipo de desarrollo puede ser menor de 3 integrantes.
Requerido contar con requerimientos funcionales antes del desarrollo.	Individuos e interacciones más que procesos y herramientas.	El cliente no identifica los requisitos detallados.
Para que el proyecto tenga éxito deben desarrollarse todas las fases.	Software que funciona más que documentación exhaustiva.	Se reducen tiempos y costos.
Las fases continúan hasta que los objetivos se han cumplido.	Responder ante el cambio más que el seguimiento de un plan.	El cliente prueba el prototipo y redefine requisitos.
La calidad del producto resultante es alta.	Colaboración con el cliente más que negociación de contratos	Existe mayor comunicación entre los desarrolladores y el cliente

Ilustración 6: Comparación de metodologías para desarrollo de aplicaciones Android.

Fuente: (Yáñez Moreno, Ponce Mendoza, & Soto Bernal, 2014).

Tabla 9: Comparación entre metodologías para desarrollo de aplicaciones móviles

CARACTERÍSTICA	MOBILE-D	SCRUM	SCRUM LEAN SIX SIGMA
Ciclos Cortos de Entrega del Producto	Es la que contempla de forma explícita los ciclos más cortos con una duración de 10 semanas	No especifica una duración determinada máxima para la entrega del producto	No especifica duración fija máxima para la entrega del producto
Diversidad de Dispositivos/Pantallas	Específica el uso de HCI para el diseño de interfaces. Sugiere casos de uso para identificar las variantes de cada dispositivo	Se considera de forma implícita en la etapa de determinación de requerimientos	En el caso de estudio abordado se limita exclusivamente a un número reducido de dispositivos de un solo fabricante lo que atenúa esta característica particular del desarrollo móvil
Entornos de Operación	Los considera en la fase de exploración y sugiere el uso AOP como técnica para abordar la funcionalidad transversal al contexto	No se considera de forma explícita, aunque pudiesen ser considerados como historias específicas que deban realizarse con el costo del incremento en el tiempo de desarrollo	No se considera de forma explícita, aunque pudiesen ser considerados como historias específicas que deban realizarse con el costo del incremento en el tiempo de desarrollo
Verificación de la Calidad del Producto	Se considera de forma explícita en las etapas tempranas de desarrollo y se sugiere el uso de pruebas de caja blanca	Se considera de forma explícita en etapas finales de cada iteración no se especifican ningún tipo de pruebas	Se considera de forma explícita en etapas finales de cada iteración no se especifican ningún tipo de pruebas
Frecuencia de Entrega	Considera una entrega de avances cada dos semanas	Si considera entregas intermedias, una por cada sprint, mas no se especifica duración	Si considera entregas intermedias, una por cada sprint, mas no se especifica duración

Fuente: Adaptado de (Ponce Mendoza, Madrid Monteverde, Garcia Gorrostieta, & Juárez de Haro, 2015)

Es así, que tomando en cuenta el análisis comparativo con otras metodologías hemos optado por desarrollar la aplicación móvil con la metodología de prototipado rápido ya que la aplicación desarrollará un estándar que es SCRUM.

Procesos de la metodología del prototipado rápido

Para poder entender cómo será desarrollada nuestra aplicación móvil nos guiaremos del modelo descrito por Pressman R. S., (2002) citado en Dapena, García-Naya, Castro, & Pan, (2010) como un resumen de las fases que son parte de esta metodología.

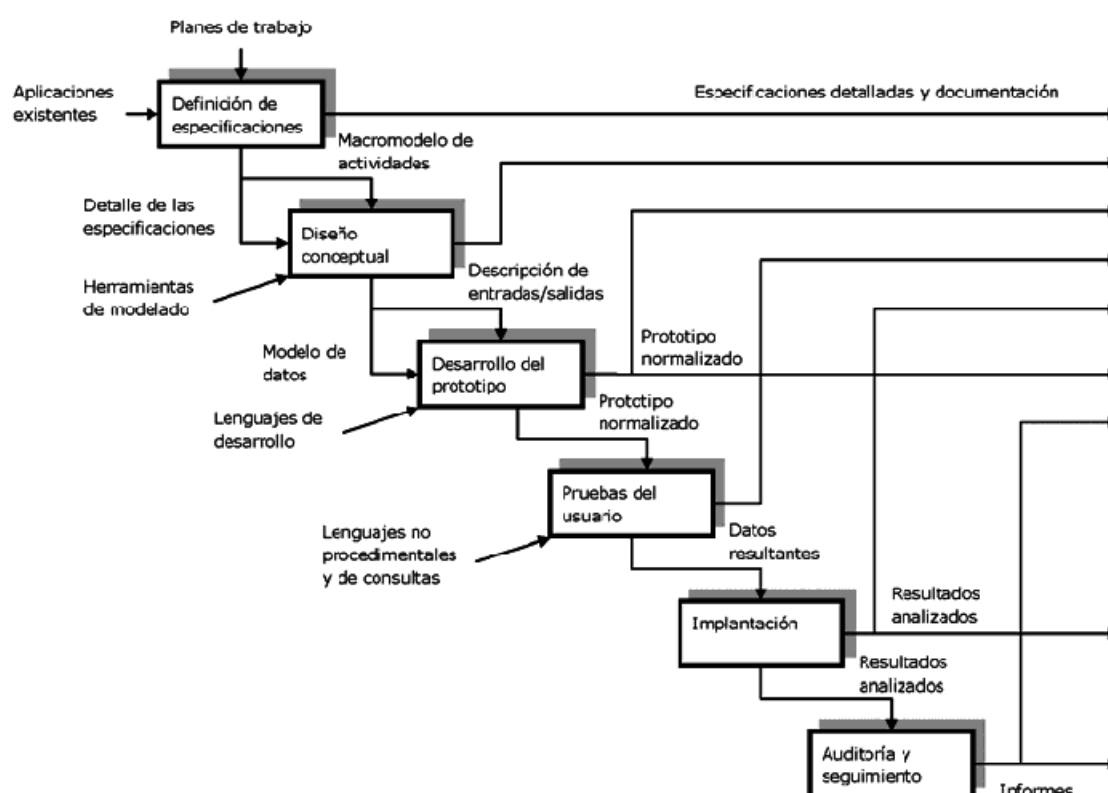


Ilustración 7: Fases de la Metodología de Prototipado Rápido (MPR)

Fuente: (Pressman R. S., 2002)

Como vemos en la imagen anterior la evolución es cíclica del prototipo en la que se involucra al usuario final en la mejora de la captura de requisitos para obtener versiones mejoradas en cada iteración con la vista de prototipos para poder evaluar que tanto se adapta el producto final a los requisitos iniciales.

- **Definición de especificaciones:** El proceso comienza con la recolección y refinamiento de requisitos, en donde se realiza la aplicación de alguna herramienta para recolectar los datos que darán lugar a la definición de especificaciones.
- **Diseño conceptual:** De acuerdo con estos requisitos, se realice el diseño conceptual usando herramientas de modelado que nos ayuden a ver de manera global lo que se pretende lograr.
- **Desarrollo del prototipo:** Ahora bien, una vez realizado el diseño conceptual, se lleva a cabo la construcción del prototipo, donde se pueden llevar a cabo con lenguajes de desarrollo ya formando una secuencia de interfaces; aquí se desarrolla el prototipo, el cual es una idea un poco más real al que se pretende llegar a realizar.
- **Pruebas del usuario:** Una vez construido, se procede a mostrársele al cliente, con las llamadas pruebas de usuario en el que se realizará una evaluación del prototipo, esto para ver si cumple con los requerimientos.
- **Implantación:** Una vez finalizado este punto de evaluación del prototipo, se lleva a cabo el proceso de implantación donde ya se obtiene un producto.
- **Auditoría y seguimiento:** En la parte de auditoría y seguimiento se busca realizar un refinamiento de este, en donde se modificarán algunos puntos que el usuario crea conveniente para que éste realice las actividades quizás de manera diferente ajustándose a los requerimientos.

Siguiendo estas fases se realizó el desarrollo de nuestra aplicación, ya que “es una técnica que facilita al programador la creación de un modelo de software, a través de la recopilación de información, con la finalidad de crear un sistema que cumpla con un ciclo de vida de desarrollo” Pressman R. S., Ingeniería de Software: Un enfoque práctico 7ma edición, (2010).

2.2.4. Metodología SCRUM

Según, Toapanta Chancusi, Vergara Ordoñez, & Campaña Ortega (2012), SCRUM es un método de gestión de proyectos, el cual puede adaptarse a cualquier tipo de proyecto y no únicamente a los de desarrollo de software.

Aplicada al desarrollo de software, está basado en el modelo de las metodologías ágiles, incrementales, basadas en iteraciones y revisiones continuas.

El objetivo principal es elevar al máximo la productividad del equipo de desarrollo. Reduce al máximo las actividades no orientadas a producir software funcional produce resultados en periodos cortos de tiempo.

Según Navarro Cadavid, Fernández Martínez, & Moreales Vélez (2013) La metodología Scrum para el desarrollo ágil de software es un marco de trabajo diseñado para lograr la colaboración eficaz de equipos en proyectos, que emplea un conjunto de reglas y artefactos y define roles que generan la estructura necesaria para su correcto funcionamiento.

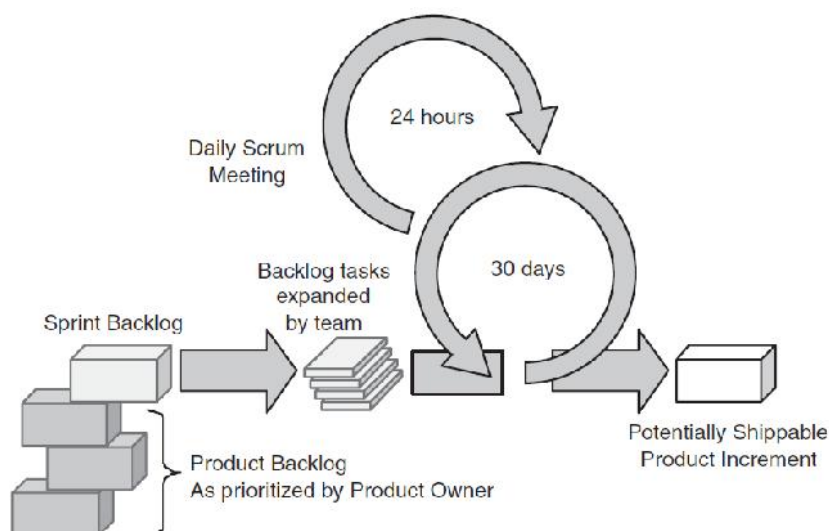


Ilustración 8: Representación de procesos SCRUM.

Fuente: (Cohn, 2009).

Aseveramos diciendo que Scrum es una metodología ágil que ayuda a las empresas dedicadas al desarrollo de Software, a agilizar sus procesos, de una forma interactiva e incremental, dando como resultado final un producto de software funcional, creando valor para el cliente, formando equipos

autoorganizados que con ayuda del Dueño del Producto trabajarán a la par durante todo el proceso que dure el desarrollo del proyecto.

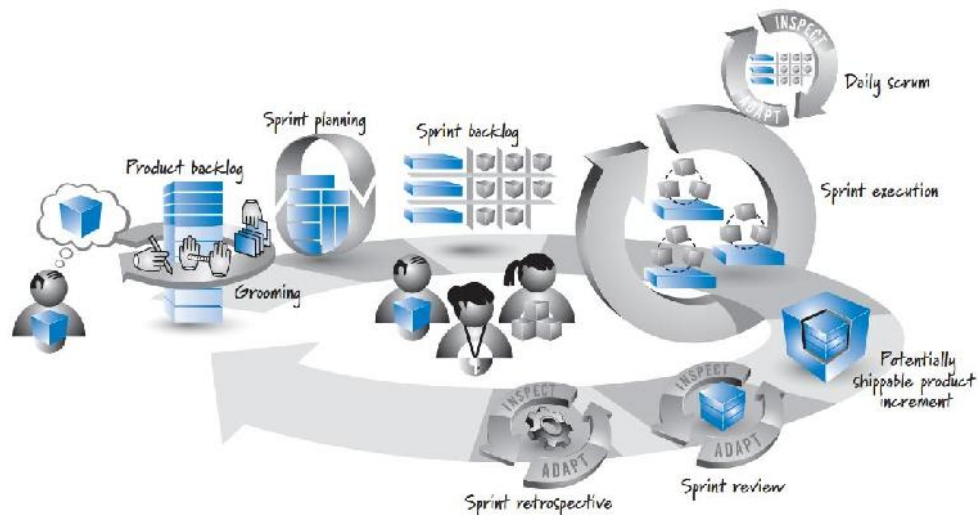


Ilustración 9: Framework Scrum

Fuente: (Kennet, 2012)

2.2.5. Roles según SCRUM

Según Scrum tenemos personas dentro del equipo SCRUM que desempeñan diversos roles: Dueño de Producto (Product Owner), el Equipo de Desarrollo (Development Team) y un Scrum Master. Cabe recalcar que los Equipos Scrum son auto organizados y multifuncionales.

Dueño del producto: El Dueño de Producto es el responsable de generar valor al producto y transmitir de forma clara, lo que el cliente realmente desea para que del Equipo de Desarrollo lleve a cabo el proyecto de la mejor manera posible, acercándose a las necesidades del cliente.

El Dueño de Producto es la única persona responsable de gestionar la Lista del Producto (Product Backlog).

La gestión de la Lista del Producto incluye:

- Expresar claramente los elementos de la Lista del Producto.
- Ordenar los elementos en la Lista del Producto para alcanzar los objetivos y misiones de la mejor manera posible.
- Optimizar el valor del trabajo desempeñado por el Equipo de Desarrollo.

- Asegurar que la Lista del Producto es visible, transparente y clara para todos, y que muestra aquello en lo que el equipo trabajará a continuación.
- Asegurar que el Equipo de Desarrollo entiende los elementos de la Lista del Producto al nivel necesario.

Es por eso que es muy importante que todo el grupo involucrado en el proyecto respete las decisiones del producto owner, dado que él es el que conoce los requerimientos del cliente final. (Ken & Jeff, 2013).

El Product Owner es la persona responsable del éxito del producto desde el punto de vista de los stakeholders.



Ilustración 10: Product Owner

Fuente: Adaptado de (Alaimo, 2013).

Equipo de desarrollo:

Son un grupo de profesionales, escogidos por la propia empresa para realizar el trabajo de entregar un producto funcional al final de cada Sprint.

El número de los miembros de este equipo, no deben ser muy robustos o muy escasos, según Ken & Jeff (2013) nos sugiere no tener menos de 3 ni más que nueve, para una mejor coordinación entre todos y así asegurar un mejor producto. Además de sugerir un número promedio nos indica los siguientes puntos a tomar en cuenta:

- Son autoorganizados. Nadie (ni siquiera el Scrum Master) indica al Equipo de Desarrollo cómo convertir elementos de la Lista del Producto en Incrementos de funcionalidad potencialmente desplegables.
- Los Equipos de Desarrollo son multifuncionales, contando como equipo con todas las habilidades necesarias para crear un Incremento de producto.
- Scrum no reconoce títulos para los miembros de un Equipo de Desarrollo, todos son Desarrolladores, independientemente del trabajo que realice cada persona; no hay excepciones a esta regla.

- Scrum no reconoce sub-equipos en los equipos de desarrollo, no importan los dominios particulares que requieran ser tenidos en cuenta, como pruebas o análisis de negocio; no hay excepciones a esta regla.
- Los Miembros individuales del Equipo de Desarrollo pueden tener habilidades especializadas y áreas en las que estén más enfocados, pero la responsabilidad recae en el Equipo de Desarrollo como un todo.

El equipo de desarrollo tiene tres responsabilidades tan fundamentales como indelegables. La primera es proveer las estimaciones de cuánto esfuerzo para cada una de las características del producto. La segunda responsabilidad es comprometerse al comienzo de cada Sprint a construir. Y finalmente, también es responsable por la entrega del producto terminado al finalizar cada Sprint.



Ilustración 11: Equipo de desarrollo

Fuente: Adaptado de (Alaimo, 2013).

Scrum Master:

Según, Ken & Jeff (2013) el Scrum Master ayuda a todos los grupos de la siguiente manera:

El Servicio del Scrum Master al Dueño de Producto

El Scrum Master da servicio al Dueño de Producto de varias formas, incluyendo:

- Encontrar técnicas para gestionar la Lista de Producto de manera efectiva.
- Ayudar al Equipo Scrum a entender la necesidad de contar con elementos de Lista de Producto claros y concisos.
- Entender la planificación del producto en un entorno empírico.
- Asegurar que el Dueño de Producto conozca cómo ordenar la Lista de Producto para maximizar el valor.
- Entender y practicar la agilidad.
- Facilitar los eventos de Scrum según se requiera o necesite.

El Servicio del Scrum Master al Equipo de Desarrollo

- El Scrum Master da servicio al Equipo de Desarrollo de varias formas, incluyendo:
- Guiar al Equipo de Desarrollo en ser autoorganizado y multifuncional.
- Ayudar al Equipo de Desarrollo a crear productos de alto valor.
- Eliminar impedimentos para el progreso del Equipo de Desarrollo.
- Facilitar los eventos de Scrum según se requiera o necesite.
- Guiar al Equipo de Desarrollo en el entorno de organizaciones en las que Scrum aún no ha sido adoptado y entendido por completo.

El Servicio del Scrum Master a la Organización

- El Scrum Master da servicio a la organización de varias formas, incluyendo: Liderar y guiar a la organización en la adopción de Scrum.
- Planificar las implementaciones de Scrum en la organización.
- Ayudar a los empleados e interesados a entender y llevar a cabo Scrum y el desarrollo empírico de producto.
- Motivar cambios que incrementen la productividad del Equipo Scrum.
- Trabajar con otros Scrum Masters para incrementar la efectividad de la aplicación de Scrum en la organización.



Ilustración 12: Scrum Master

Fuente: Adaptado de (Alaimo, 2013).

2.2.6. Planificación del Backlog

La Lista de Producto es una lista ordenada de todos los requisitos necesarios para construir un producto funcional, y es la única fuente de requisitos para que el equipo de desarrollo pueda realizar cualquier avance o cambio en el producto.

Cabe recalcar que el Dueño de Producto (Product Owner) es el único responsable de la Lista de Producto, incluyendo su contenido, disponibilidad y ordenación. Tanto el Scrum Master como el equipo de desarrollo pueden aportar

ideas para refinar las historias de usuario, pero el trabajo de escribir, priorizar y todo lo que involucre la planificación del Backlog recae en el Product Owner.

Se define el plan de trabajo, que se va a entregar y como se lograra. El dueño del sistema y la estimación de cantidad de trabajo. Esta actividad dura 8 horas para un sprint de un mes.

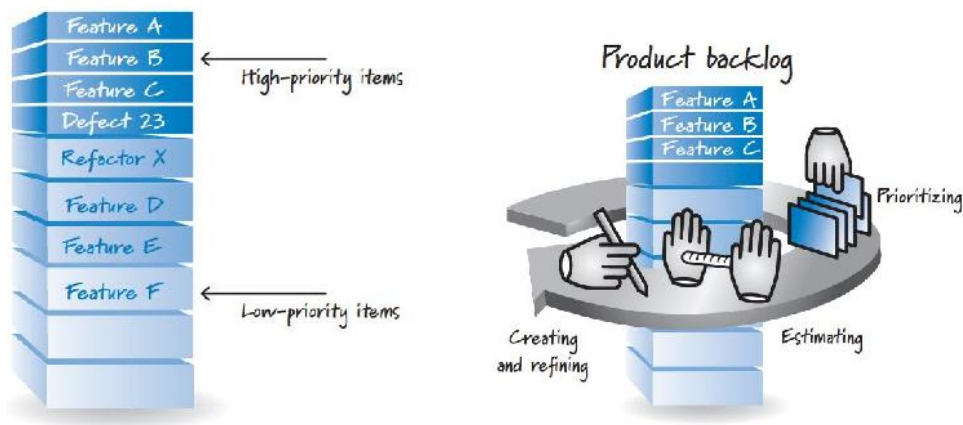


Ilustración 13: Product Backlog

Fuente: (Kennet, 2012)

2.2.7. Historias de usuario

Historias de usuario, que son una técnica ligera para expresar los requisitos de software. Una historia de usuario es una breve descripción de la funcionalidad como se ve por un usuario o cliente del sistema. Historias de usuario son de forma libre y no hay sintaxis obligatoria. Sin embargo, puede ser útil pensar en una historia general de montar el formulario: ". Como <tipo de usuario>, quiero <capacidad de> manera que <valor de negocio>" (Mike, 2005).

En Scrum el dueño del producto es el que escribe las historias de usuarios, se recomienda sean descritas en un lenguaje fácil de entender para los clientes y no en un lenguaje muy técnico, para que se pueda describir fácilmente el producto y se pueda priorizar las historias.

En un inicio el dueño del producto describe sus necesidades, se recomienda describir en una pequeña tarjeta para no escribir demasiado, se llega a un acuerdo con el grupo de desarrollo y se hace un armado de un conjunto de historias, que posteriormente el grupo de desarrollo hará sus estimaciones en tiempo y esfuerzo para culminarla.

Una historia de usuario es valiosa para el comprador del Sistema o Software es por eso por lo que se planifica una descripción detallada de lo que va a contener, que se espera y que se puede probar.

Atributos de las Historias de Usuario

Según Cohn (2009) Una buena historia de usuario debe cumplir con seis atributos:

- Independiente
- Negociable
- Valiosa para los usuarios o clientes
- Estimable
- Pequeña
- Testeable

Una historia de usuario debe ser independiente, no en su totalidad, pero si se puede dar el caso de desligarla de un grupo sería mucho mejor ya que permite que se trabaje sola sin ser un pre-requisito para iniciar con la otra historia de usuario. Llevando esto al campo de desarrollo es muy difícil que una historia no dependa de otra, pero por recomendación se debería trabajar de tal modo que sea lo más independiente posible.

Debe ser negociable en el sentido que debe ser discutido a detalle entre el dueño del producto y el grupo de desarrollo para llegar a un acuerdo y un mejor entendimiento de ambas partes y no que se imponga algo visto en un solo sentido.

Una historia de usuario debe ser valorada por sus clientes, y por el dueño del producto, si una historia no genera ningún valor al cliente, es inservible, y en vez de apoyar al avance con el proyecto generaría retraso porque se estaría haciendo algo irrelevante a lo que realmente se necesita.

Si una historia se puede estimar, hace que el proyecto siga su curso, se pueden presentar algunos inconvenientes, pero serán mínimos, es por eso, que es muy importante la experiencia que tenga el grupo de desarrollo para poder determinar qué cantidad de tiempo se necesita para terminar el proyecto.

Además, una historia de usuario debe ser pequeña, es decir no debe ser grande, porque nos podremos dar con la sorpresa de que tenga dentro varias historias más; pero tampoco debe ser muy pequeña, porque carecería de entendimiento y sería difícil de estimar, sabiendo que se puede agrupar con otras de su mismo rubro.

User Story Title	Find Reviews Near Address
As a <user role> I want to <goal> so that <benefit>.	As a typical user I want to see unbiased reviews of a restaurant near an address so that I can decide where to go for dinner.
Template	

Ilustración 14: Plantilla de historia de usuario y tarjeta.

Fuente: (Kennet, 2012)

Finalmente, una Historia de Usuario debe ser comprobable, para que se desarrolle con éxito, permitiendo además a los desarrolladores saber cuándo esa historia está lista para salir del Sprint y volverse un producto funcional.

Priorización de historias de usuario

Parte importante de la planificación es la priorización de historias de usuario y esta gran responsabilidad se comparte entre todos los asistentes a la reunión, para hacerlo se hace una comparación entre historias y además Cohn (2009) nos dice que se debe considerar 4 factores cuando se prioriza el desarrollo:

- El valor financiero de tener las características.
- El coste de desarrollo (y quizás de apoyo) las nuevas características.
- La cantidad y la importancia del aprendizaje y nuevos conocimientos en vías de creación por las características.
- La cantidad de riesgo eliminado mediante el desarrollo de las características.

Este establecimiento de prioridades se comparte entre todo el equipo, pero la mayor parte de responsabilidad recae sobre el propietario del producto, ya que es el quien mejor conoce el negocio y que es lo primero que se necesita o tiene más valor para la organización.

2.2.8. Criterios de aceptación

Para poder verificar las historias de usuario debemos usar pruebas de aceptación.

Según Cohn (2009) Las pruebas de aceptación es el proceso de verificar que las historias se han desarrollado de tal manera que cada uno trabaja exactamente de la forma en que el equipo cliente espera que funcione. La prueba funciona mejor como un proceso de dos pasos:

En primer lugar, señala sobre las pruebas futuras están apuntadas en la parte posterior de las cartas de historia. Esto puede hacerse en cualquier momento que alguien piense en una nueva prueba.

En segundo lugar, las notas de las pruebas se convierten en pruebas de pleno derecho que se utilizan para demostrar que la historia ha sido codificada correcta y completamente.

Nos recomienda escribir al reverso de la tarjeta de Historia de Usuario, no como una lista de requisitos, sino como una manera de describir los detalles de lo que el cliente espera de esa historia.

En Scrum dentro del grupo de desarrollo puede estar una persona que cumpla una doble función, tanto de desarrollador como de téster, esta persona debe ser dedicado y calificado no solo para interactuar con el cliente sino también para comunicarse dentro del grupo con los desarrolladores.

Estas pruebas de desarrollo deben hacerse en todo momento del desarrollo, no debemos esperar a que se termine la iteración para recién realizar el testeo, sino que las pruebas de aceptación muestras más temprano se hagan es mucho mejor, si es posible antes de que empiece el Sprint cuando se reúnen todos para especificar tantas pruebas iniciales de les ocurra, ya que estaría llena de supuestos y explicaciones de lo que el cliente en realidad espera del producto y ayudaría al grupo de desarrollo a tener una visión más clara del producto final.

Cabe recalcar que estas reuniones pueden ser una vez por semana mientras dure el Sprint que en promedio es de 4 semanas, sin dejar de lado las reuniones diarias para ver los avances.

Tener estos criterios o pruebas de aceptación nos permite realizar el proyecto de la mejor manera, evitando que hagamos demasiado, muy poco, desperdiciemos tiempo, esfuerzo y costos en ello. Además, nos proporciona una gran cantidad de información que el grupo de desarrollo puede usar antes de empezar a programar evitando imprevistos o cambios en el proceso de codificación de la historia.

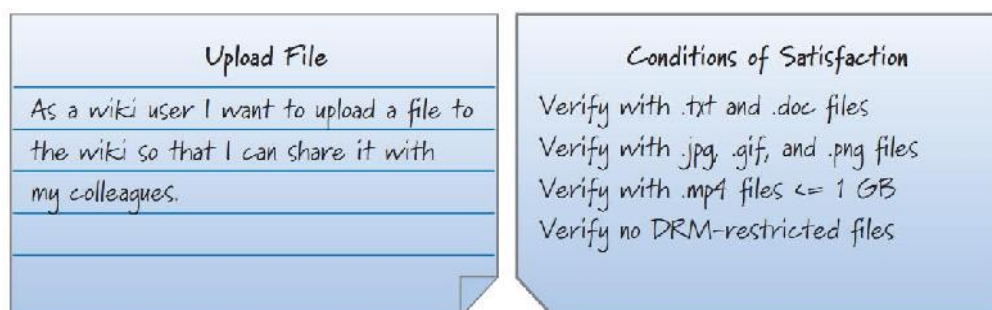


Ilustración 15: Condiciones de satisfacción de la historia del usuario.

Fuente: (Kennet, 2012)

Como lo hemos mencionado anteriormente el cliente ya tiene una visión clara de lo que espera que haga el producto, así que él estas pruebas de aceptación deben ser especificadas por el cliente, el cual puede trabajar con un probador o téster para crear las pruebas y para saber cuándo el desarrollo de una historia ha finalizado con éxito.

Una vez descritas estas pruebas, los programadores ejecutan la codificación y el probador sería el encargado de validar que el programa funcione tal y como se pidió en un inicio, para esto el programa pasa por una serie de pruebas, siguiendo las herramientas que mejor se acomoden a la situación.

Tanto las historias de usuario como las pruebas son imprescindibles como parte del proceso de desarrollo, muchos por equivocación creen que las pruebas se deben realizar al final de toda la codificación, cuando en realidad las pruebas están presentes en todo el proceso y a cada instante, tanto en el grupo de desarrollo para ver internamente como se debe codificar en un lenguaje de bajo nivel, como en el lado del cliente que debe ver si el producto está avanzando como el en inicio específico.

Cohn (2009) Nos menciona que hay tipos de pruebas y el equipo de desarrollo como el cliente deben trabajar juntos para asegurar que se realicen las pruebas adecuadas, en su mayoría las pruebas de historia son las funcionalidades sin embargo hay otros tipos de pruebas a considerar. Por ejemplo, es posible que desee considerar cualquiera o todos de los siguientes:

- Pruebas de la interfaz de usuario, lo que garantiza que todos los componentes de la interfaz de usuario se comportan como se esperaba.
- Las pruebas de usabilidad, se realiza para asegurar que una aplicación puede ser utilizado fácilmente.
- Las pruebas de rendimiento, que se realiza para medir qué tan bien funcionan la aplicación bajo diversas cargas de trabajo.
- Las pruebas de tensión, en el que la aplicación se somete a valores extremos de los usuarios, transacciones, o cualquier otra cosa que pueda poner la aplicación en virtud estrés.

Como vemos las pruebas de aceptación son muy importantes es así que se según García Ramírez (2011) Scrum como metodología involucra:

- Pruebas iterativas.
- Backlog (repositorio) de los procesos o módulos a ser testeados.
- Planificación y desarrollo de Sprints (conjunto de procesos o módulos a ser probados) durante un periodo de 15 o 30 días máximo.
- Seguimiento continuo del progreso de cada uno de los Sprints planificados (Burn Down Chart).
- Reuniones Diarias (15 a 30 minutos) con el equipo de pruebas (Daily Scrum Meeting), con el fin de discutir: Que se ha realizado desde la última reunión, Que se necesita realizar hasta la próxima reunión y que impedimentos existen, si los hay.
- Reunión de Planificación de los Sprints (Sprint Planning Meeting), con el equipo de pruebas, con el fin de distribuir los módulos o procesos a ser probados.

- Reunión de Revisión de los Sprints (Sprint Review Meeting), con el propósito de discutir aspectos que permitan optimizar los Sprints a realizar o pendientes.

2.2.9. Planificación un Sprint

Esta reunión se lleva a cabo al inicio de cada sprint y los involucrados en son el dueño del producto, el Scrum Master y el grupo de desarrollo, el tiempo promedio no dura más de 8 horas.

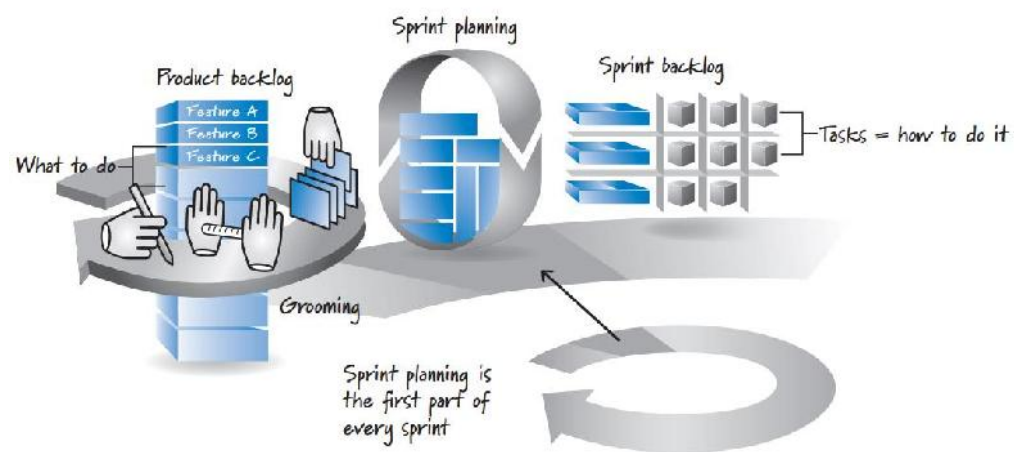


Ilustración 16: Planificación del Sprint.

Fuente: (Kennet, 2012)

Una vez descritas las historias de usuario, clasificamos las historias en varias columnas llamadas pilas que representa una iteración. Cada Sprint tendrá un número de historia, un tamaño, la complejidad con respecto a otras historias y la estimación de la velocidad de desarrollo. Las historias de más alta prioridad serán tomadas para entrar en el primer sprint, y las de menos prioridad estarán programadas para el segundo sprint y así sucesivamente. Esta planificación del Sprint se realiza en 2 fases:

Primera fase: Se describen las necesidades del dueño de producto de una manera superficial y prioriza las que más le interesa, se establece que historias se ponen el primer lugar para el primer Sprint y las restantes para las demás iteraciones.

Segunda Fase: Todo el grupo en conjunto determina que elementos se pueden mover de la pila de producto hacia el sprint backlog para empezar el desarrollo

siempre tomando en cuenta los objetivos del sprint y el nivel de compromiso por parte del equipo de desarrollo.

2.2.10. Segregación de historias de usuario en tareas

Una vez que estén definidos los Sprints estos pasan con sus respectivas historias de usuario para ser segregadas en partes más pequeñas que son las tareas.

Si las tareas seleccionadas son terminadas dentro del sprint (en combinación con otros artículos apuntados para el mismo sprint), se procede a trabajar las historias de usuario con sus respectivas tareas del siguiente Sprint, es así como se repite el ciclo hasta que el equipo se quede sin capacidad para hacer más trabajo.

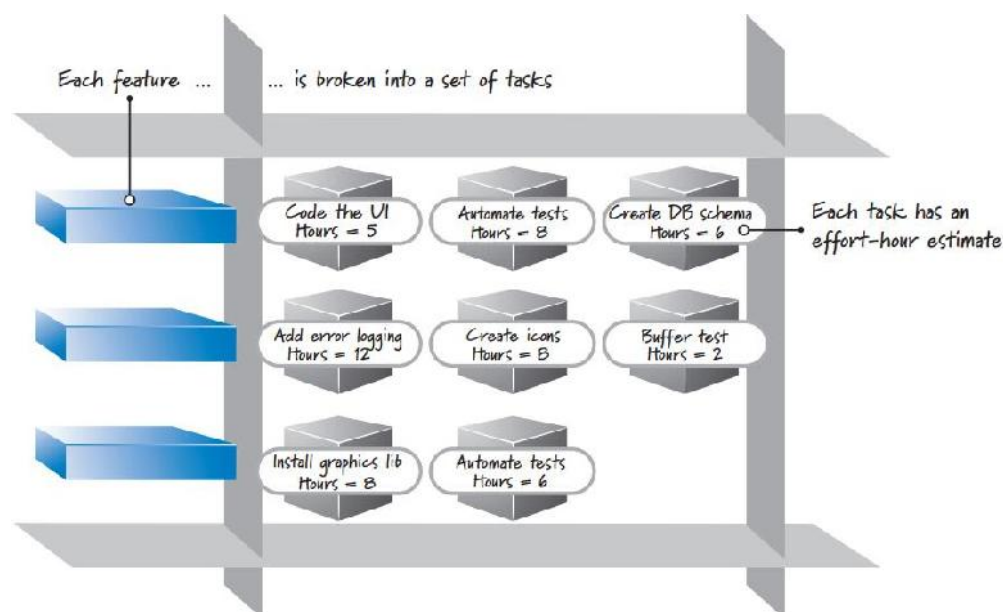


Ilustración 17: Pila del Sprint detallada en tareas.

Fuente: (Kennet, 2012)

El equipo de desarrollo es el encargado de segregar las historias de usuario en tareas descritas en un lenguaje técnico para que todos en el grupo sepan que se tiene que hacer, o qué tareas faltan desarrollar para completar una o más historias de usuario y poder completar los Sprints hasta su finalización.

2.3. GLOSARIO DE TÉRMINOS

TÉRMINO	SIGNIFICADO
Aplicación móvil	: Software implementado para ser ejecutado en dispositivos móviles como celulares inteligentes, Smartphone, Tablet, etc. y que pueden ser usadas en diversas situaciones por los usuarios.
Android	: Plataforma para aplicaciones móviles, es totalmente abierta tanto los fabricantes de dispositivos móviles y para los desarrolladores de aplicaciones que funcionan en tales dispositivos.
Backlog	: Pila de producto, generalmente es un conjunto de historias de usuario, que se ordenan por prioridad en orden descendente y que determina el orden en que se debe desarrollar cada historia de usuario.
Criterios de aceptación	: Detalles específicos de lo que la historia de usuario de hacer o generar, para más entendimiento de lo que realmente se desea según la necesidad del Product Owner.
Desarrollo de software	: Procedimiento para crear un producto de software.
Dueño de producto	: Rol que cumple una persona en un proyecto Scrum, por lo general es alguien que pertenece a la empresa de desarrollo, es el que transmite las necesidades de los Stakeholders al equipo Scrum, para desarrollar un producto de software funcional.
Equipo de desarrollo	: Conjunto de personas encargadas de ejecutar las tareas establecidas en el proyecto de software, son personas conocedoras y expertas en todo lo que involucra desarrollo de software.
Historia de usuario	: Descripción corta y precisa para detallar requerimientos que se irán desarrollando en el proyecto.

Iteración	: Acto de repetir el proceso de desarrollar historias de usuario agrupadas por prioridad.
Metodología ágil	: Conjunto de procesos y procedimientos referidos a ingeniería del software basados en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan o cambian con el tiempo según la necesidad del proyecto.
Metodología tradicional	: Conjunto de procesos y procedimientos referidos a ingeniería del software basados en requisitos y soluciones que se mantienen y no cambian a través del tiempo en un proyecto.
Modelo de calidad	: Conjunto de prácticas relacionadas a los procesos de gestión y el desarrollo de proyectos que cumplan con los objetivos fijados en lo referente a la calidad del producto o servicio y que son tomadas como referencias para otros proyectos.
Plataforma	: Es el entorno de software común en el cual se desenvuelve la programación de un grupo definido de aplicaciones.
Planificación	: Reunión de los actores involucrados en un proyecto con metodología Scrum para elaborar el plan conforme al que se ha de desarrollar
Producto de software	: Resultado final y funcional que se genera después de desarrollar y testear un conjunto de componentes lógicos que permiten que una computadora realice distintas tareas.
Prototipo	: Los prototipos son modelos preliminares de interfases que permiten estudiar y probar aspectos específicos del producto final de software.

Proyecto de software	: Planificación de actividades que se encuentran interrelacionadas y coordinadas que establecen un modo determinado y un conjunto de medios necesarios, para posteriormente desarrollar un producto de software.
Release Backlog	: Es un retraso en la liberación, formado por un conjunto de historias de usuario seleccionadas para ser desarrolladas en los Sprints, generalmente se toma como un proceso subjetivo antes de llevar a cabo el desarrollo de las historias de usuario.
Scrum	: Nombre de una metodología ágil, usada básicamente para desarrollo de software, esta metodología se caracteriza por ser incremental, adaptativa y es una de las usadas a nivel mundial.
Scrum master	: Persona conocedora de la metodología Scrum, sus procesos y procedimientos, ayuda en todo momento al equipo Scrum a implementar la metodología dentro de la empresa, es el que guía a todos a un buen desarrollo de la metodología durante la realización de todo el proyecto.
Sprint	: Es una iteración de corta duración que permite a los desarrolladores, identificar qué historias de usuario necesitan desarrollar para proporcionar un resultado completo con un producto que sea potencialmente entregable.
Tarea	: Parte más pequeña que resulta de desglosar historias de usuario, son descripciones cortas de la parte técnica que involucra desarrollar las historias, siguen una secuencia dependiente una de la otra.
Verificación	: Comprobación que realiza el Product owner a las historias de usuario para ver si fueron entregadas tal cual las solicitó al inicio del proyecto.

CAPÍTULO III. DESARROLLO DE LA PROPUESTA

3.1. PLANIFICACIÓN Y VERIFICACIÓN DE HISTORIAS DE USUARIO

En este proyecto se tomó como base la metodología SCRUM, como un proceso iterativo y que trabaja con equipos auto gestionados, usa una secuencia de pasos que permiten que las empresas que desarrollan software, entregar producto de calidad en cada iteración. Identificaremos los actores que son: El Product owner, el Scrum Master y el equipo de desarrollo.

3.1.1. Preparación de un proyecto SCRUM

Empezamos por realizar una adecuada preparación de un proyecto SCRUM, y es cuando se inicia a trabajar con un proyecto, para esto identificamos los actores involucrados, en esta fase solo es importante describir el papel del Product owner, él tiene que presentar el caso de negocio, el cual es la comprensión negocio y el valor del proyecto.

el Product owner, cumple un rol muy importante ya que la visión del proyecto debe ser lo más corta y accesible posible para ser transmitida de forma clara y entendible, a tal punto que los miembros del equipo sepan cual es la visión y el objetivo del proyecto, ya que en base a eso se harán las estimaciones del esfuerzo y se tendrá una visión más clara de lo que se necesita,

Más adelante se involucró los demás actores que son el equipo de desarrollo y el Master Scrum que actuará en todo momento como mediador en la reunión.

Una vez definidos el caso de negocio y la visión del proyecto, el equipo en conjunto entra reunión, para esto debe tener suficientes ítems que pueden haber surgido durante el trabajo de crear el caso de negocio, transformados de un documento de requerimientos, o creados directamente por el Dueño Del Producto, para luego ser enumerados y priorizados en el Backlog Del Producto.

El Backlog Del Producto contiene toda la funcionalidad que el producto debería tener, SCRUM propone dividir el Backlog en una colección de entregables que pueden ser cambiados por distintos motivos como:

- Cambio en el negocio.
- El usuario encuentra que la funcionalidad de un entregable es más importante para entrar en producción cuanto antes.
- Demasiadas características para un entregable.

Además de eso, en la preparación se debería tomar en cuenta el alcance del proyecto, aportaciones de temas técnicos, acuerdos de como trabajara el equipo.

Roles en SCRUM

- Dueño Del Producto (representa la voz del cliente)
- Scrum Master (el facilitador de Scrum asegura y guía en el proceso de Scrum, y quita escollos)
- Miembros Del Equipo De Scrum (los responsables de crear el producto)

3.1.2. Backlog del producto

El Backlog del producto en Scrum es el artefacto más importante de Scrum, ya que dirige la construcción. Hasta el equipo más efectivo y eficiente fallará si construye lo que no es necesario.

Representa todo aquello que esperan los clientes, usuarios y los interesados en el producto; no es un documento de requisitos, sino una herramienta de referencia para el equipo.

PRODUCT BACKLOG



Ilustración 18: Estructura del Product Backlog.

Fuente: Adaptado de (Axosoft, 2016)

Historias de usuario

Las historias de usuario son la unidad funcional del Product Backlog, para fines del proyecto usamos la siguiente estructura:

- Código.
- Título.
- Descripción:
- Prioridad.

Existen varias estructuras de historias de usuario, pero para fines del proyecto usamos la siguiente estructura.

- **Código:** Representado por un número o un valor alfanumérico único.
- **Título:** Título descriptivo de la historia de usuario
- **Descripción:** Descripción sintetizada de la historia de usuario siguiendo la siguiente secuencia:
 - Como – Rol
 - Yo quiero – Algo
 - Para – beneficiarme
- **Criterios de aceptación:** Lista detallada de requerimientos funcionales, pueden ser varios criterios por historia de usuario.
- **Prioridad:** Prioridad en la implementación de la historia de usuario respecto al resto de las historias de usuario, está considerado en 3 rangos: Alta, media y baja. Siempre y cuando las Historias de usuario tengan registrados los criterios de aceptación.

Criterios de aceptación:

Una vez establecidas las historias de usuario, se definieron los criterios de aceptación.

Los criterios de aceptación se traducen en pruebas, descritas en pocas palabras, en un lenguaje entendible para que el equipo de desarrollo pueda especificar requerimientos funcionales, y es por ello que toman tanta importancia en las historias de usuario.

Los criterios de aceptación son muy importantes porque ayudan al equipo de desarrollo a entender el funcionamiento del producto, de manera que estimarán mejor el tamaño de la historia de usuario.

Derivado de esta sintaxis los elementos de los criterios de aceptación son:

- Código de escenario: Número (ejemplo 01, 02, 03 ó 04), que identifica al escenario asociado a la historia.
- Descripción del escenario: Describe el contexto del escenario que define un comportamiento.

A continuación, mostraremos la estructura de los criterios de aceptación con los que se va a trabajar en el proyecto:

Se escriben en cuanto se obtengan historias de usuario preparadas para entrar en un sprint y se refinan en la planificación de sprint.

Tabla 10: Ejemplo Historias de Usuario de un proyecto.

Entrega 1. Comercializar Eventos				
Prio.	Como	Necesito	Para	Criterios Aceptación
1	Comercial	Crear un evento confirmado	Hacer el seguimiento del mismo	<ul style="list-style-type: none"> - Debe tener Nombre, Fecha, Descripción, Destinatarios, Programa, - Instructor, Lugar, Ciudad, País, - Capacidad, Precios y Promociones: - SEB (Super Early Bird), EB (Early Bird). - Las promociones son opcionales.
2	Comercial	Ver listado de eventos confirmados	No superponer eventos	<ul style="list-style-type: none"> - Mostrar Nombre, Ciudad y País - Muestra solo los futuros - Ordenado por fecha ascendente

3	Comercial	Modificar evento confirmado	Corregir cualquier error o reprogramarlo	- Permite modificar todos los campos.
4	Comercial	Cancelar evento confirmado	Dejar de seguirlo	- Desaparece del listado de eventos confirmados.

Fuente: Adaptado de (Alaimo, 2013)

Priorización de historias de usuario

El Dueño Del Producto es el responsable de mantener el contenido del Backlog del producto, y asegurar que están continuamente priorizados para alinearse a las necesidades del negocio.

Plantear entrega del producto al usuario: el cliente debe tener en mente que cuando más cubra y mejor estructurada su historia de usuario. más rendimiento le sacará a su inversión.

Estimar el costo del primer Sprint: cuantas más historias de usuario tenga el proyecto, mayor coste terminarlo. Un buen contrapeso al tiempo total de desarrollo es poder determinar la fecha de implementación de la primera etapa productiva del producto.

Valorar cada historia de usuario: conocer los elementos más valorados por el usuario ayudaran al equipo de desarrollo a priorizar el resto de las historias.

Para priorizar usamos la siguiente escala: Alta, media y baja.

Técnica de priorización:

Podemos priorizar de la siguiente manera:

- Proponer al dueño de producto y demás involucrados que ubiquen las tarjetas del Backlog sobre la mesa (una tarjeta por cada historia).
- Explicar que se trata de establecer un orden de prioridad para que el equipo pueda dar el máximo valor en el menor tiempo.
- Explicar que la dinámica es de 30 minutos.
- Pedir a los participantes que dividan las tarjetas en dos columnas de igual tamaño -prioridad 1 y prioridad 2-, si queda un número impar de cartas, ubicar la carta extra en la columna de prioridad 1.

- Una vez hecho esto, cambiar el nombre de la columna de prioridad 2 como prioridad 3 y dejar a un lado.
- Ahora debería tener 3 columnas (en orden de prioridad) de tarjetas, con no más de 10 tarjetas en cada una de las dos primeras columnas.

Si durante la reunión todavía hay tiempo, puede pedir a los participantes que ordene las tarjetas de prioridad 1. Una alternativa es seguir dividiendo la columna 1 hasta tener el número deseado de tarjetas, asignado a las primeras con prioridad alta, a las siguientes con prioridad media y a las últimas con prioridad baja.

3.1.3. Planificación de Sprints

El propósito de un Sprint es convertir un set de ítems del Backlog del Producto en un incremento de funcionalidad potencialmente productiva.

Además de ir segregando historias de usuario de acuerdo a la prioridad asignada por el Product Owner para ser desarrolladas por el equipo de desarrollo siendo tomadas en orden de prioridad desde las historias de usuario con prioridad ALTA, pasando por la prioridad MEDIA y culminando con las historias de usuario de prioridad BAJA hasta que se termine el producto completo de software.



Ilustración 19: Historias de usuario ordenadas por prioridad en los Sprints.

Fuente: Adaptado de (Axosoft, 2016)

Una vez priorizadas las historias de usuario y sus respectivos criterios de aceptación, se establecen los sprint de acuerdo al orden escogido.

Tabla 11: Ejemplo Historias de Usuario de un proyecto.

Entrega 1. Comercializar Eventos				
Prio.	Como	Necesito	Para	Criterios Aceptación
SPRINT 1				
1	Comercial	Crear un evento confirmado	Hacer el seguimiento del mismo	A definir
2	Comercial	Ver listado de eventos confirmados	No superponer eventos	A definir
SPRINT 2				
5	Comercial	Crear un texto con fechas y valores	Pegarlos en los e-mails de respuesta	A definir

Fuente: Adaptado de (Alaimo, 2013)

Ahora estas historias de usuario se desglosarán en tareas de la siguiente manera:



Ilustración 20: Historias de usuario segregadas en tareas.

Fuente: Adaptado de (Axosoft, 2016)

Para una mejor visualización del desglose en tareas y la asignación respectiva.

Tabla 12: Ejemplo del detalle de las tareas.

TAREA	ENCARGADO
Código de la pantalla de búsqueda básica	Susan
Código de búsqueda avanzada	Susan
Pantalla de resultado del código	Jay

Fuente: Adaptado de (Alaimo, 2013)

Es fácil rastrear las tareas, el desarrollador que realiza cada tarea, y las estimaciones en la Pizarra blanca.

3.1.4. Verificación del cumplimiento de los criterios de aceptación

Este proyecto permite que, al finalizar el proceso de desarrollo, tanto el equipo de desarrollo como el Product owner puedan validar y verificar que las historias de usuario y sus respectivos criterios de aceptación fueron cumplidos tal cual el cliente solicitó. Para esto haremos un proceso inverso:

- Todas las tareas deben estar en estado finalizado.
- Se amarrarán a las historias de usuario a la que pertenecen.
- El Product owner como el equipo de desarrollo, al ir testeando el producto de software irá registrando en un checklist si los criterios de aceptación que se especificaron en un inicio cumplen con lo especificado.
- Se valorizará el nivel de satisfacción del cliente.
- Si el cliente tiene alguna observación la podrá agregar al final como un cambio o una mejora para el próximo sprint.

Tabla 13: Ejemplo del detalle de los criterios de aceptación.

Historia de Usuario	Criterios de Aceptación	Observaciones
HU 0001	Criterio por definir	- OB001
		- OB002
		- OB003
	Criterio por definir	Sin Observaciones
HU 0002	Criterio por definir	- OB004
		- OB005
		- OB006
	Criterio por definir	- OB007
		- OB008

Fuente: Desarrollo Propio

3.2. CONSTRUCCIÓN

Para poder construir nuestra aplicación usamos la metodología de prototipado rápido anteriormente planteada, seguiremos las fases para el desarrollo de la aplicación:

3.2.1. Definición de especificaciones

Comenzaremos con la recolección y refinamiento de requisitos, en donde se realiza la aplicación de alguna herramienta para recolectar los datos, dado que la metodología no nos restringe una técnica de recolección de especificaciones, hemos considerado listarlos de la siguiente manera:

En esta sección se muestra la lista de requerimientos identificados para resolver el problema planteado, tanto funcionales como no funcionales.

La información sobre SCRUM y el apoyo de diversas fuentes dieron como resultado una lista de funcionalidades deseadas que se listan a continuación:

- Facilitar de descarga e instalación.
- Facilitar de autenticación de usuarios.
- Registrar un único usuario con el rol de Administrador.
- Permitir al administrador el registro de usuarios y asignarles un rol (Product owner, Scrum Master y Equipo de desarrollo).
- Crear Proyectos y asignarles fecha de inicio.
- Crear el Product Backlog por cada proyecto.
- Crear y listar las historias de usuario dentro del backlog.
- Agregar criterios de aceptación por cada historia de usuario.
- Priorizar historias de usuario según la evaluación del Product owner.
- Crear Sprints de acuerdo a la prioridad de historias de usuarios seleccionadas por el Product Owner.
- Asignar tiempo (semanas) a cada uno de los Sprint.
- Calcular el tiempo del proyecto en base a la suma del tiempo de todos los Sprint.
- Asignar personal del Development Team al proyecto en el que trabajará.
- Cada usuario del Development Team debe tener el permiso de agregar tareas a las historias de usuario.
- Verificar el avance de las tareas hasta terminarlas (100%).
- Verificar el cumplimiento de Historias de Usuario, de acuerdo a los criterios ingresados.

- Mostrar las tareas por desarrollar, unidas a las historias de usuarios y sus respectivos criterios de aceptación.
- Permitir al usuario comprobar que las historias de usuarios estén terminadas en base al cumplimiento de sus criterios de aceptación.
- Ingresar observaciones en las historias que no cumplan con ciertos criterios de aceptación.
- Permite ingresar las historias de usuario que no se cumplieron con los criterios de aceptación.
- Permite terminar los Sprints siempre y cuando se hayan terminado las historias de usuario.
- Restringir las interfaces de acuerdo a los roles de usuario.
- Visualizar reportes del Product Backlog, Tareas por Historias de usuario y listado de criterios de aceptación.

Descripción de Requerimientos Funcionales

A continuación, se presenta una breve descripción por cada requerimiento funcional, basada en un código, nombre que identifica la funcionalidad, una descripción breve que explica su función. Los requerimientos funcionales para la construcción de la aplicación móvil son:

Tabla 14: Resumen de requerimientos funcionales

Código	Nombre	Descripción
R-001	Inicio de Sesión	El administrador al ingresar por primera vez tendrá que registrar los datos de la empresa, y crear una cuenta de administrador.
R-002	Cuenta de Administrador	El administrador podrá tener acceso a las siguientes interfaces: Perfil de Empresa, Gestión de roles, gestión de usuarios.
R-003	Perfil de Empresa	Se mostrará la información general de la empresa y beneficios de HUS, para todos los usuarios.
R-004	Gestión de usuarios	El administrador podrá crear, editar y eliminar usuarios para 3 roles: Product Owner, Scrum Master y Development Team

R-005	Cuenta de Product Owner	El Product Owner podrá tener acceso a Perfil de empresa, gestión de proyectos checklist y reportes.
R-006	Gestión de proyecto	El Product owner podrá crear, editar y eliminar proyectos, así como asignar a usuarios de Development Team a proyectos creados.
R-007	Creación de HU en el backlog	el Product owner podrá registrar, editar y eliminar historias de usuario en el backlog.
R-008	Creación de Criterios de Aceptación.	el Product Owner podrá agregar los criterios de aceptación a las historias de usuario en el backlog.
R-009	Priorización de Historias de Usuario	el Product Owner podrá priorizar (alta, media y baja) y ver los criterios de aceptación de las historias de usuario en el backlog.
R-010	Creación de Sprints.	el Product Owner podrá crear Sprints de acuerdo a la prioridad de las historias de usuario, las HU de más alta prioridad irán en el Primer Sprint y así sucesivamente.
R-011	Duración por Sprint	el Product Owner podrá asignarle un tiempo de desarrollo (semanas) a cada sprint.
R-012	Duración por proyectos	La aplicación debe ser capaz de calcular la duración del proyecto (semanas) en base al tiempo especificado en cada sprint.
R-013	Generación de Reportes	En la lista de reportes se debe mostrar para todos los usuarios, excepto el administrador.
R-014	Checklist	El Product Owner podrá ver los proyectos a verificar, así como los Sprints relacionados para realizar la verificación de criterios.
R-015	Validación de Criterios de Aceptación.	Se cargarán las historias de usuario y los criterios de aceptación por historia para seleccionar las que cumplen y las que no cumplen.

R-016	Agregar observaciones	Si el Product owner no está conforme con alguno de los criterios, podrá agregar observaciones, las que se mostrarán al visualizar una historia y en los reportes.
R-017	Cuenta de Equipo de desarrollo	El Development Team podrá tener acceso a Perfil de empresa, gestión de tareas y reportes.
R-018	Gestión de tareas	El Development Team podrá visualizar los proyectos al cual ha sido asignado, visualizar los Sprints del proyecto y las historias de usuario por sprint.
R-019	Creación de tareas	El equipo de desarrollo podrá crear, editar y terminar tareas por cada historia de usuario.
R-020	Avance de tareas	Se visualizará las historias de usuario con su porcentaje de avance de acuerdo a las tareas terminadas.
R-021	Cuenta de Scrum master	El scrum master podrá tener acceso a todo para poder visualizar, monitorear y hacer seguimiento a los procesos alineándolos a SCRUM.

Fuente: Desarrollo Propio

Requerimientos no funcionales

Los requerimientos no funcionales son los siguientes:

- R 1.** La aplicación debe ser desarrollado para dispositivos móviles como celulares y tablets con sistema operativo Android mayores a la versión 4.0.
- R 2.** La aplicación debe ser desarrollada con el IDE de Android Studio y en lenguaje de programación Java.
- R 3.** La aplicación debe de tener conexión a internet, para un buen funcionamiento.
- R 4.** La aplicación de mantener conexión constante con una web service que la proveerá de todos los datos.
- R 5.** El dispositivo debe de tener un mínimo de memoria de 50MB.

3.2.2. Diseño conceptual:

De acuerdo con estos requisitos, realizamos el diseño conceptual usando herramientas de modelado que nos ayuden a ver de manera global lo que se pretende lograr.

Diagrama de base de datos

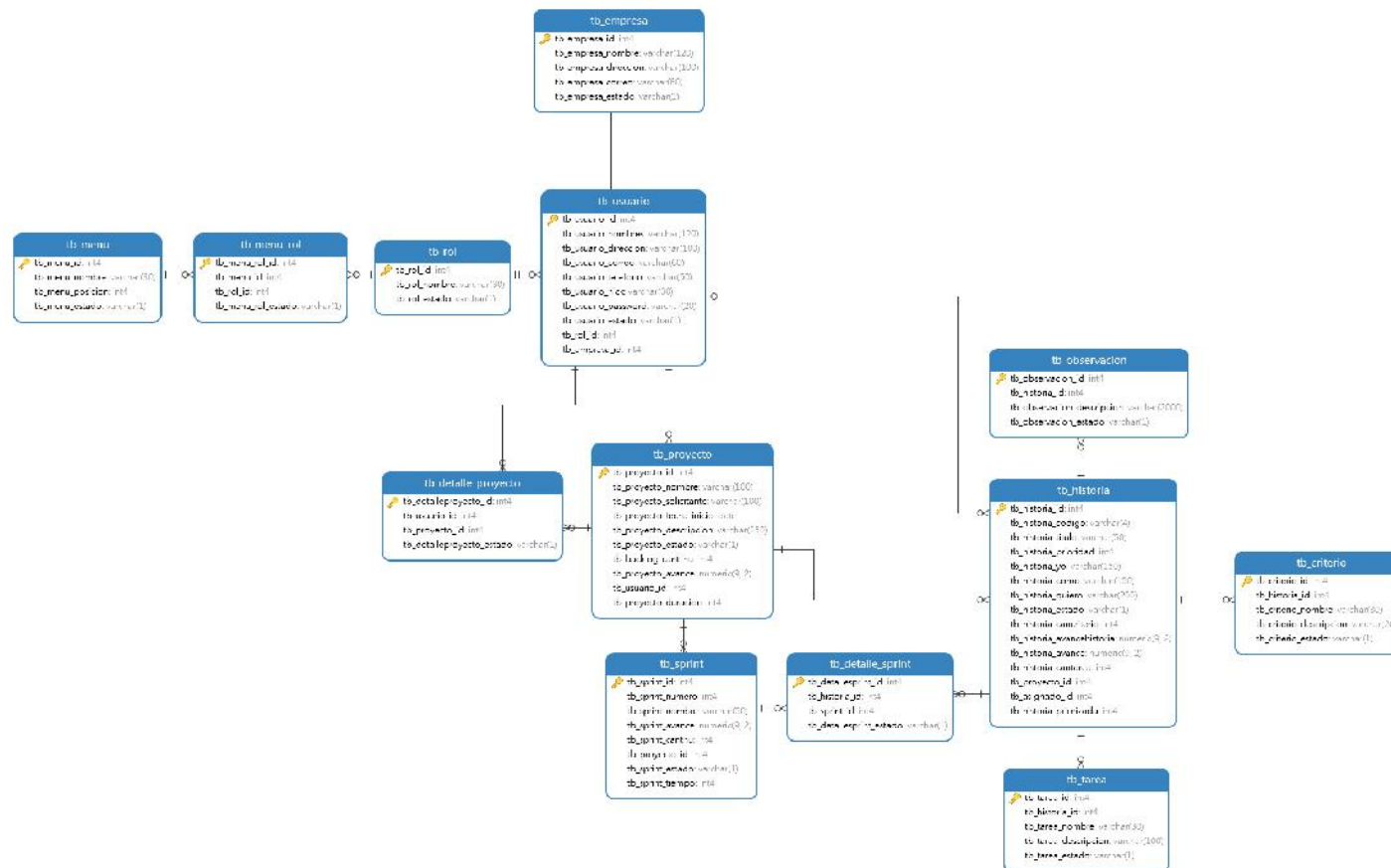


Ilustración 21: Base de datos para la construcción de la aplicación móvil.

Fuente: Desarrollo Propio

Modelado de procesos

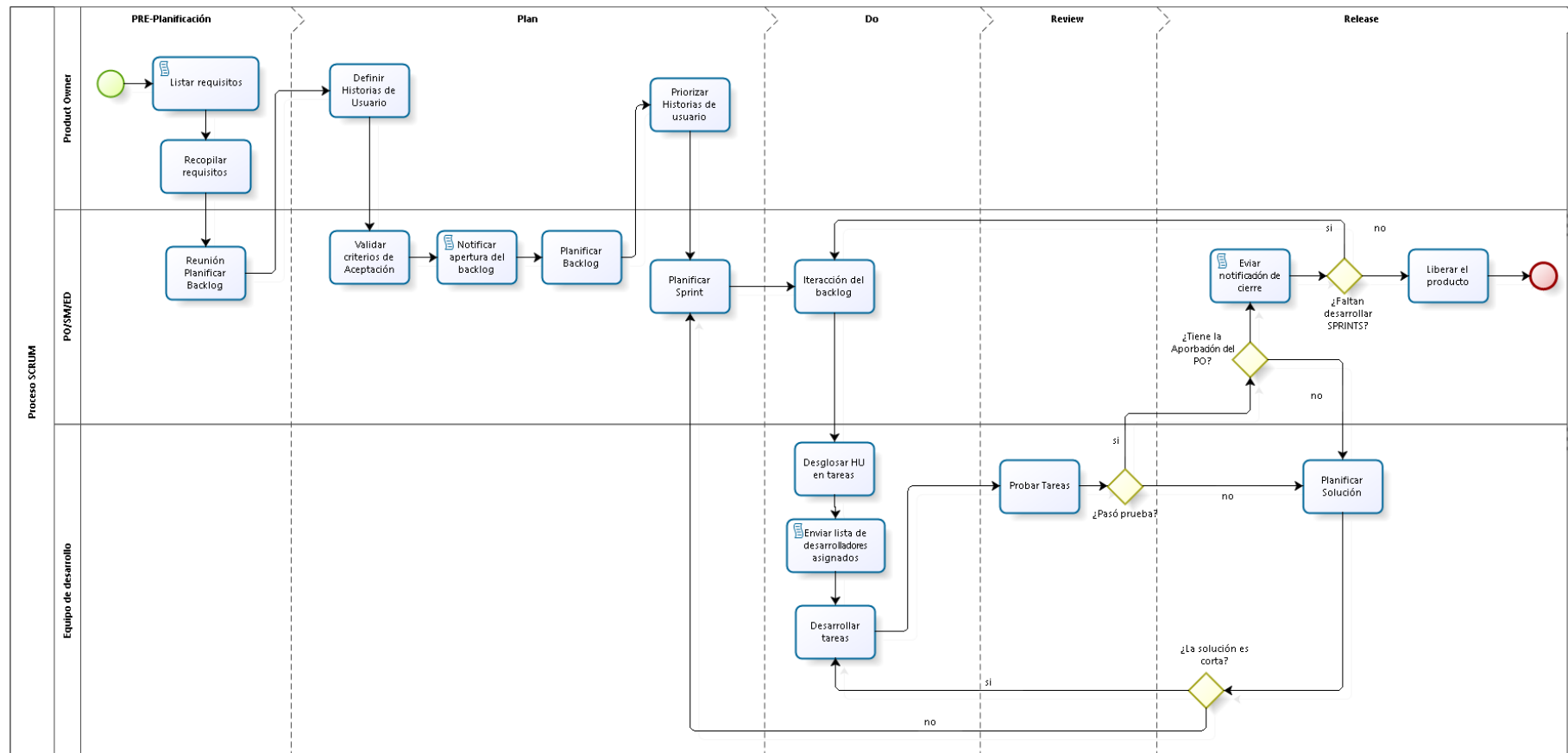


Ilustración 22: Diagrama de procesos en SCRUM

Fuente: Elaboración Propia.

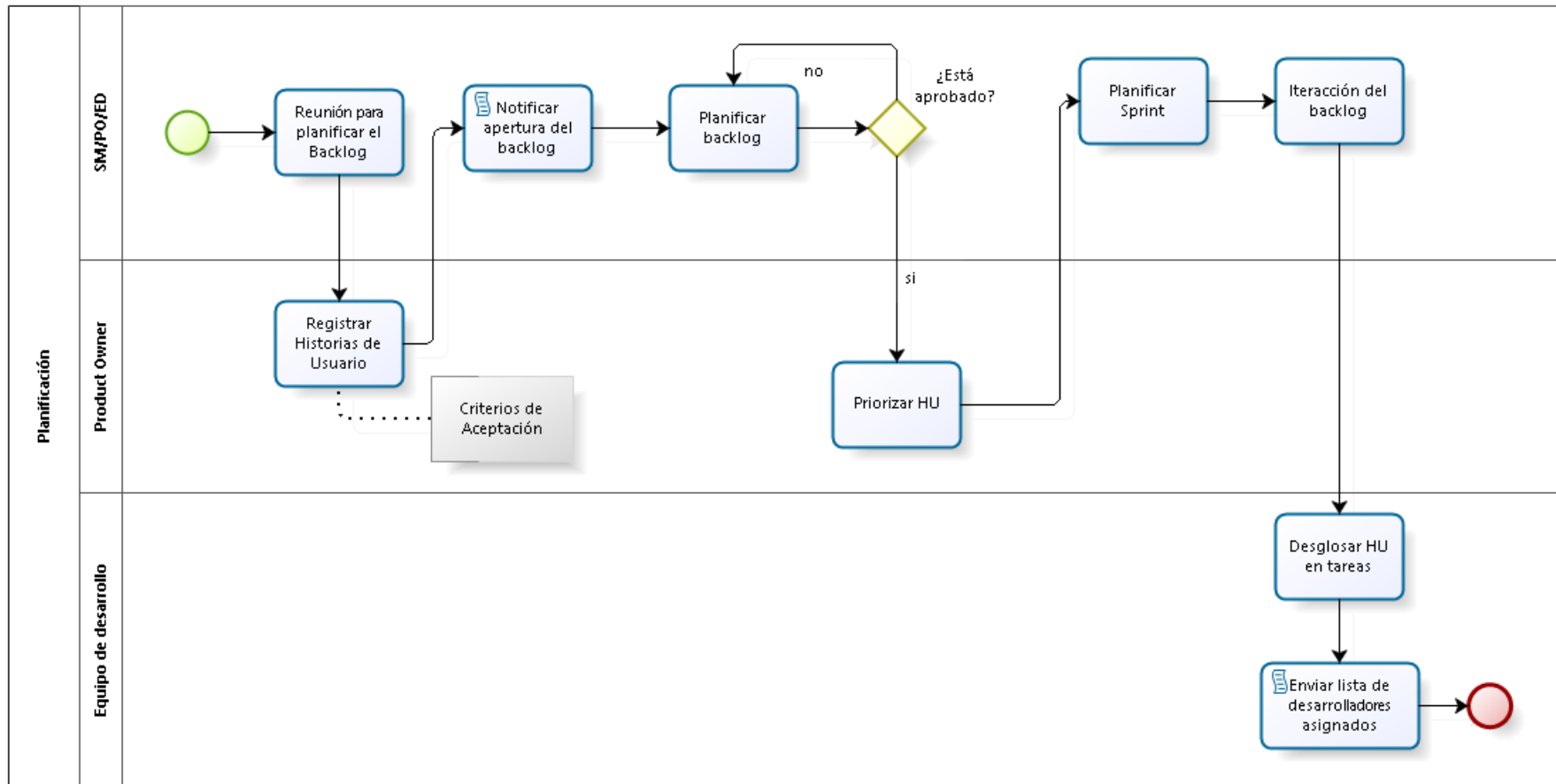


Ilustración 23: Diagrama de Verificación de Historias de Usuario

Fuente: Elaboración Propia.

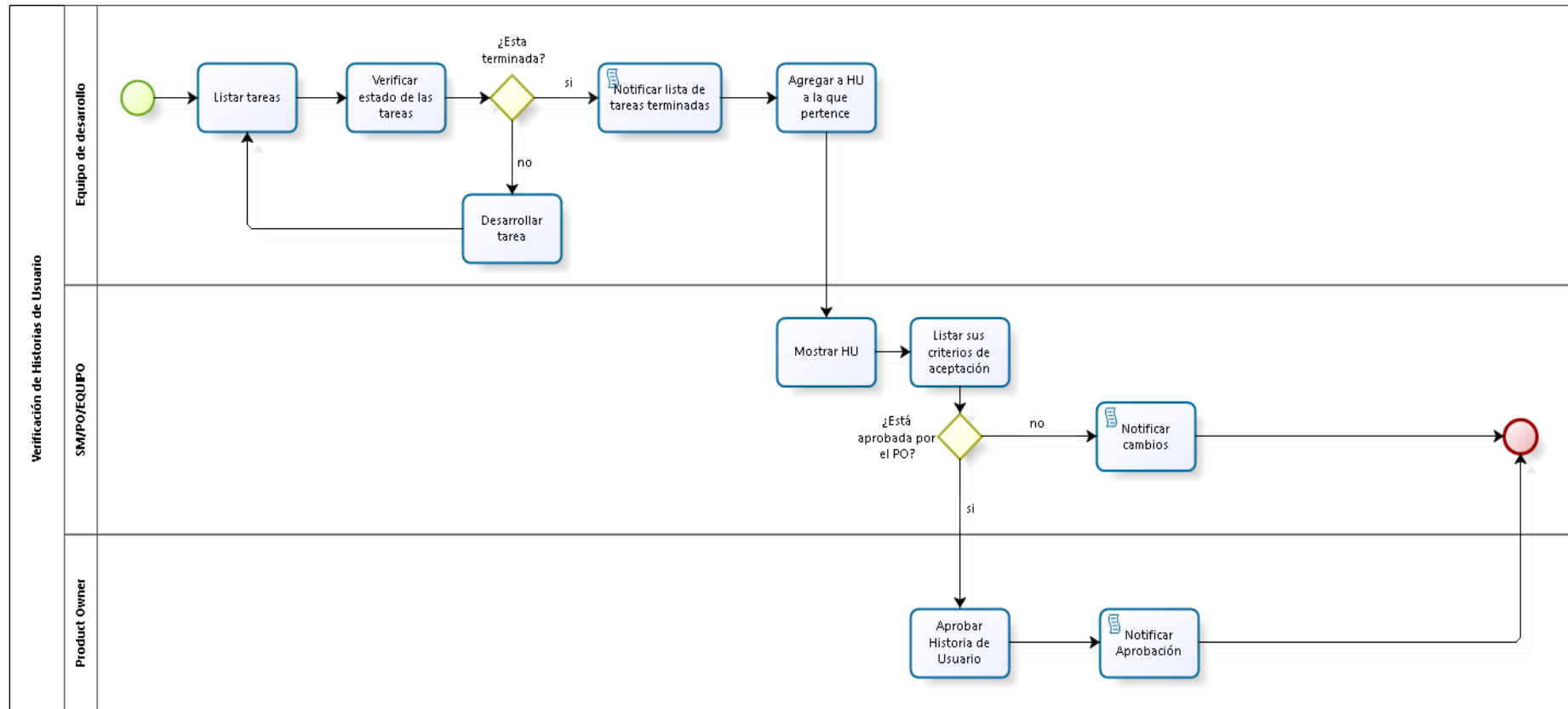


Ilustración 24: Diagrama de verificación de Historias de Usuario

Fuente: Elaboración Propia.

3.2.3. Desarrollo del prototipo:

Llevamos a cabo la realización de los prototipos tomando en cuenta la herramienta JustinMind para la creación de prototipos de aplicaciones web y móviles ya formando una secuencia de interfaces; aquí se desarrolla el prototipo, el cual es una idea un poco más real al que se pretende llegar a realizar.

Para realizar los prototipos se realizaron varios diseños y secuencias de las mismas, siendo esta última la que más se asemejó a nuestro proyecto final, tomando en cuenta cada uno de los requerimientos anteriormente descritos.

Código: 001

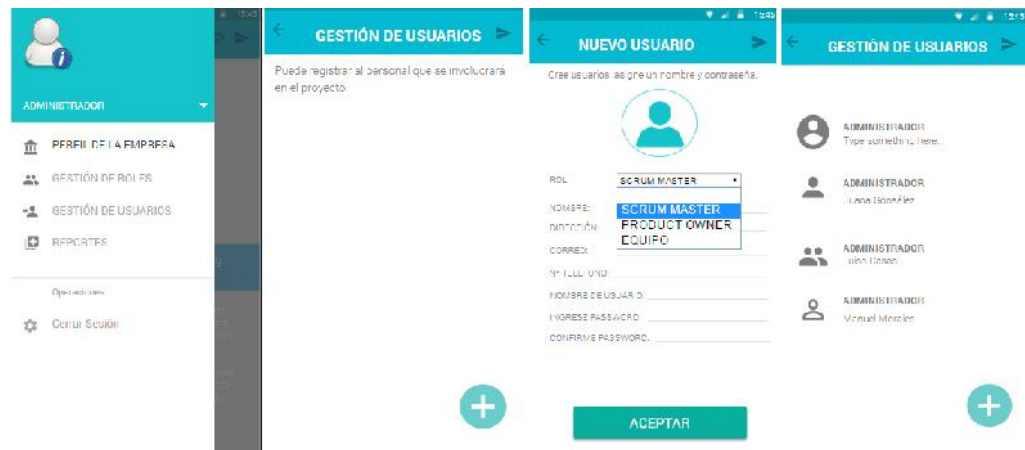
Nombre: Inicio de sesión - Registro

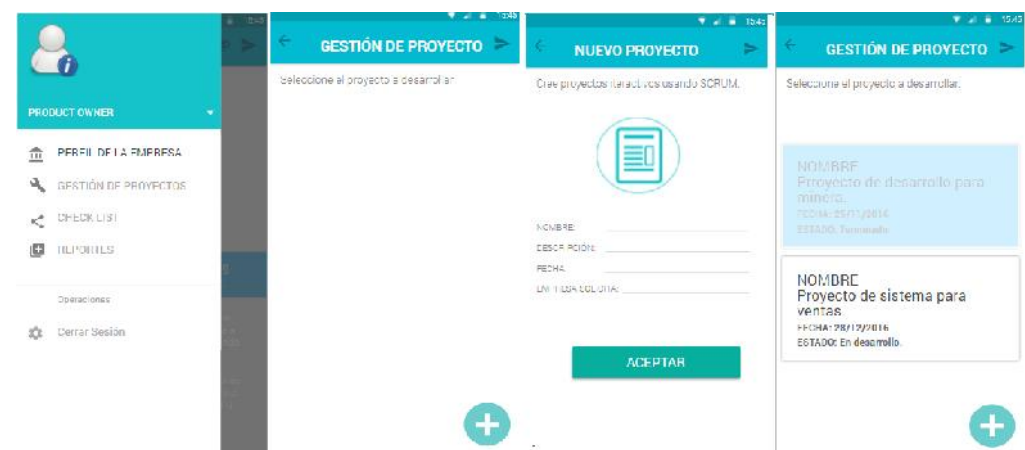


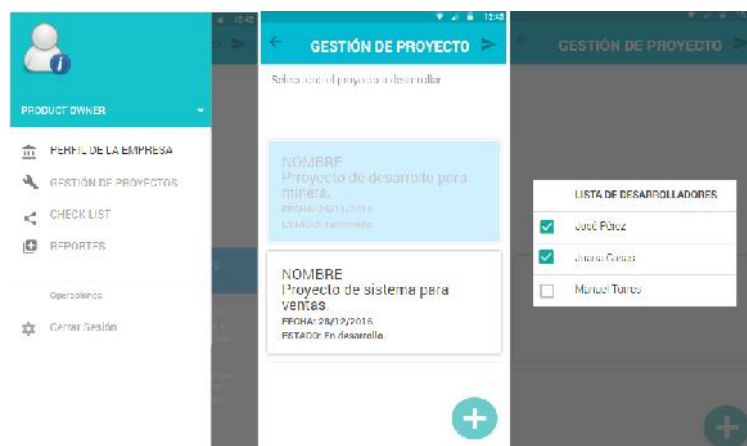
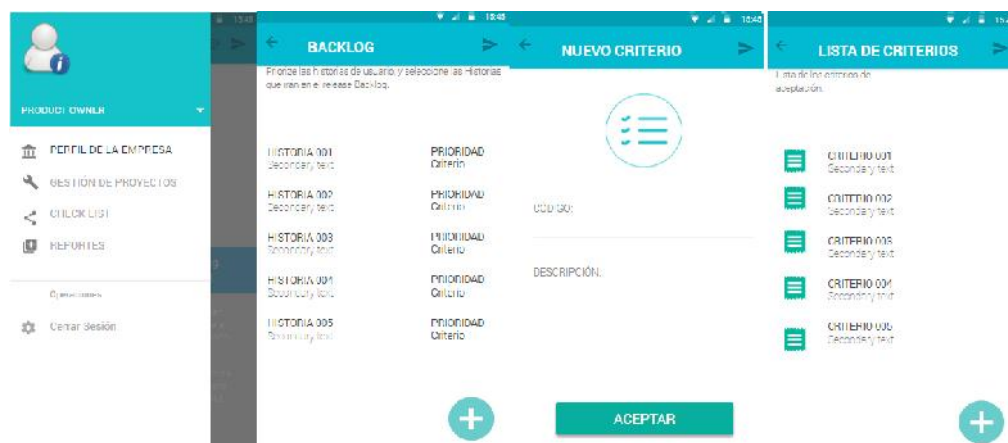
Código: 002

Nombre: Gestión de Cuenta de Administrador – Perfil de empresa y gestión de roles.



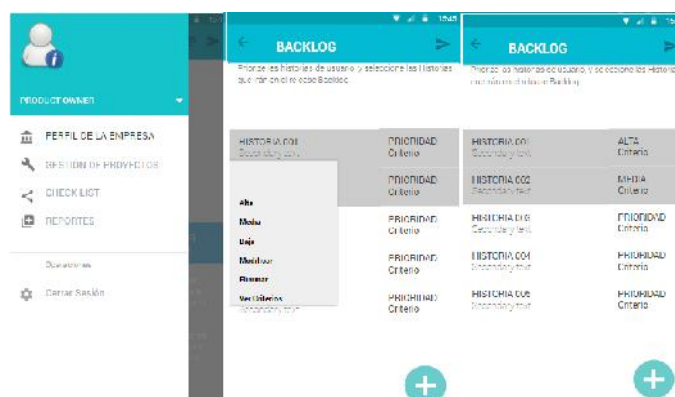
Código: 003**Nombre:** Gestión de Cuenta de Administrador – Gestión de usuarios.

Código: 004**Nombre:** Gestión de cuenta de Product Owner

Planificación de proyectos**Código: 005****Nombre:** Gestión de cuenta de Product Owner - Gestión de proyecto.


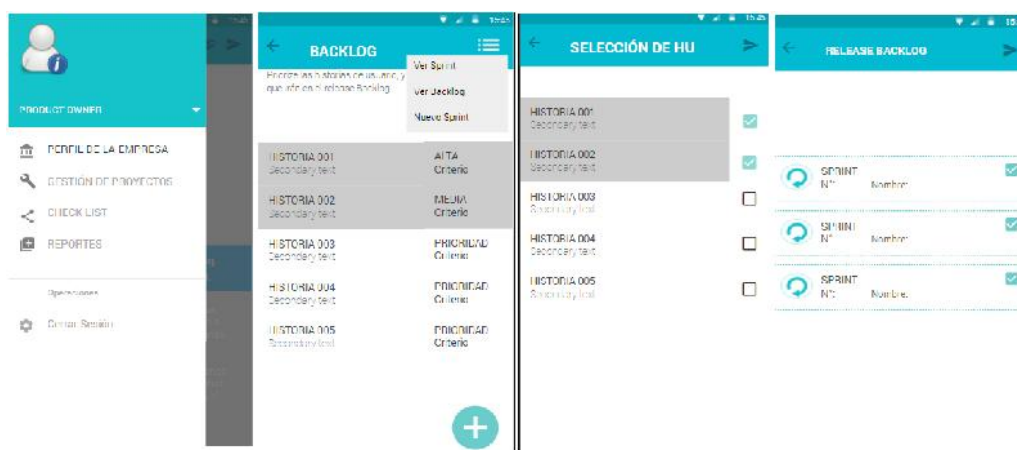
Código: 006**Nombre: Gestión de cuenta de Product Owner – Asignar personal****Generación de Historias de Usuario****Código: 007****Nombre: Gestión de cuenta de Product Owner – Creación de HU en el backlog****Código: 008****Nombre: Gestión de cuenta de Product Owner – Gestión de Historias de Usuario – Criterios de Aceptación.**

Código: 009

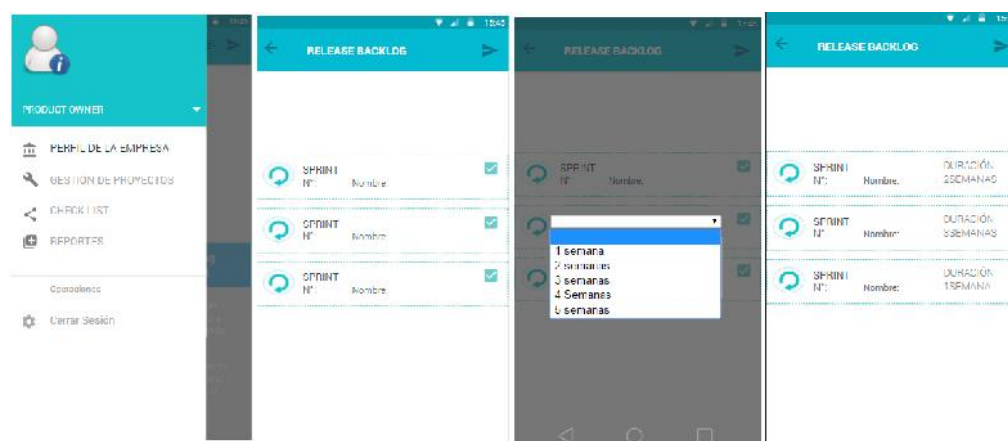
Nombre: Gestión de cuenta de Product Owner – Gestión de Historias de Usuario - Priorización.

**Planificación de Sprints****Código: 010**

Nombre: Gestión de cuenta de Product Owner – Gestión de Sprints – Creación de Sprints.

**Código: 011**

Nombre: Gestión de cuenta de Product Owner – Gestión de Sprints – Duración de Sprints.



Código: 012

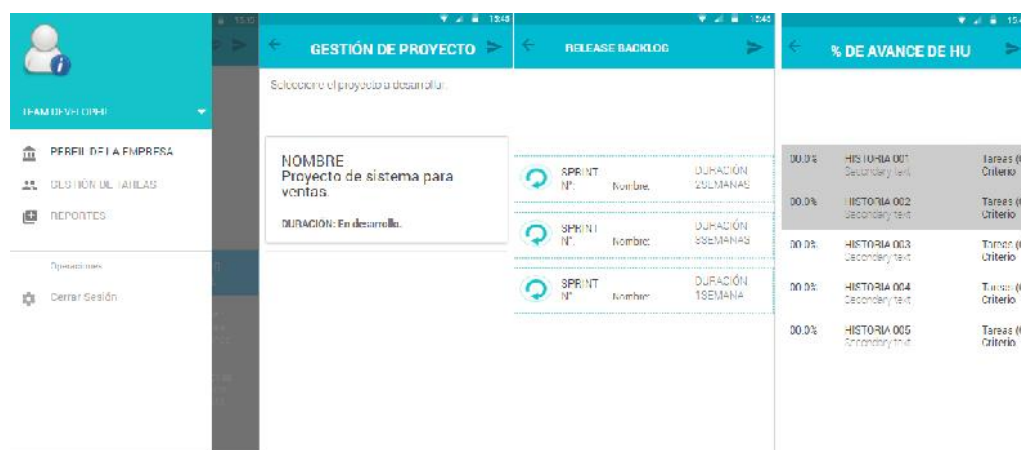
Nombre: Gestión de cuenta de Product Owner – Gestión de Proyecto – Duración por proyecto.

**Desglose de Historias de Usuarios en tareas****Código: 013**

Nombre: Gestión de cuenta de Equipo de desarrollo

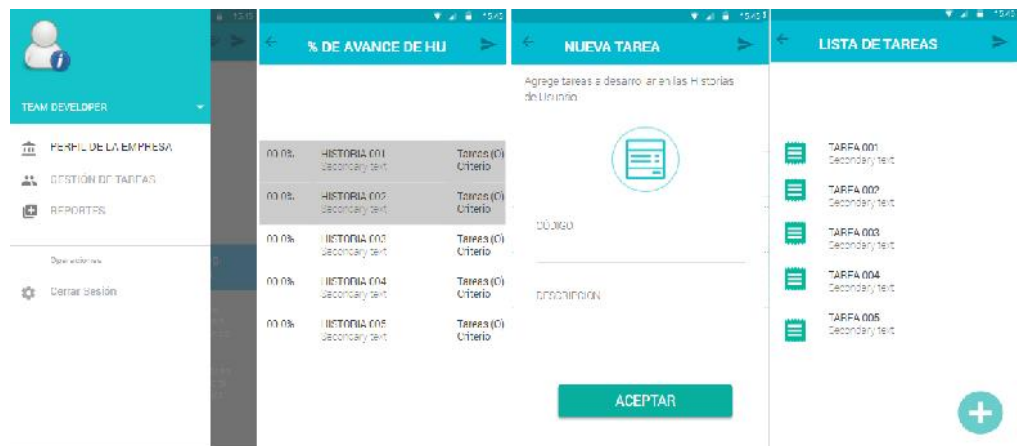
**Código: 014**

Nombre: Gestión de cuenta de Equipo de desarrollo - Gestión de Sprints

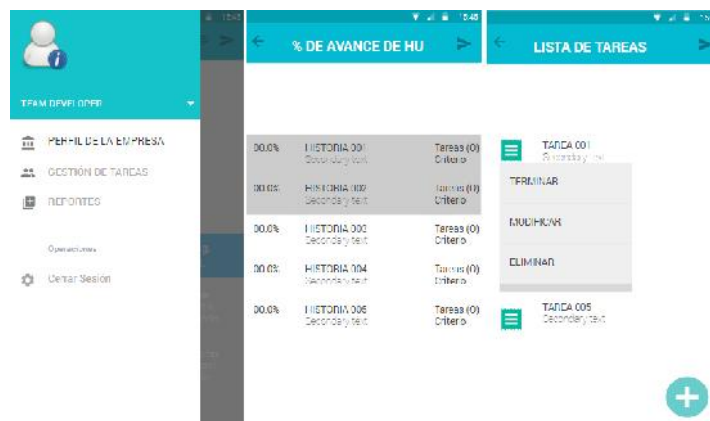


Código: 015

Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de Sprints – Creación de tareas.

**Código: 016**

Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de tareas

**Código: 017**

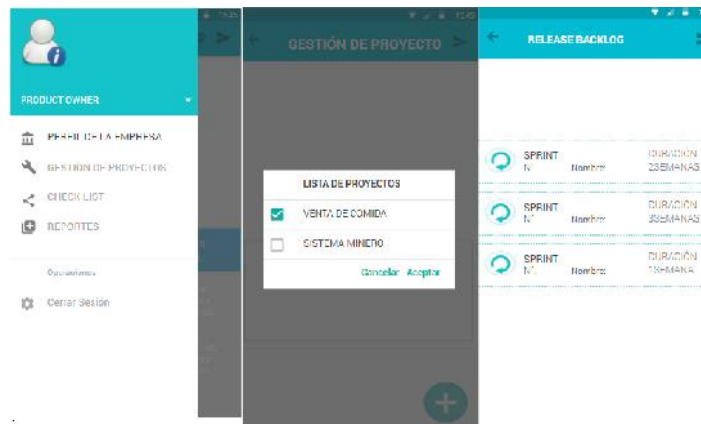
Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de tareas - Avance



Verificación del cumplimiento de historias de usuario

Código: 018

Nombre: Gestión de cuenta de PO – checklist



Código: 019

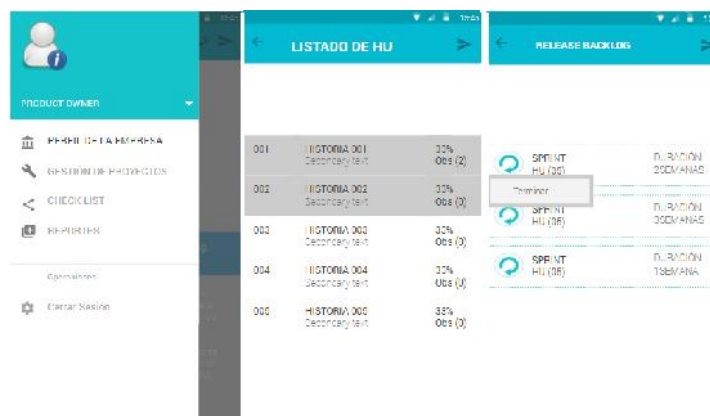
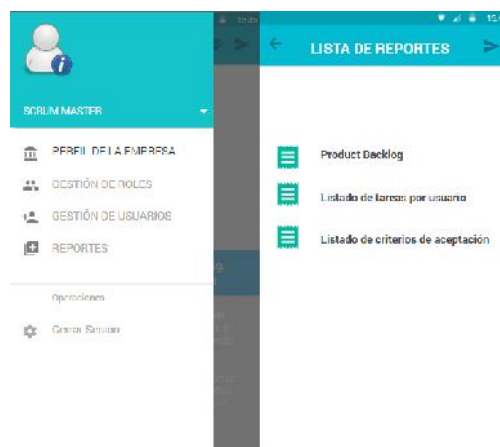
Nombre: Gestión de cuenta de Product Owner – checklist – criterios de aceptación



Código: 020

Nombre: Gestión de cuenta de PO – checklist – Observaciones en los criterios de aceptación



Código: 021**Nombre:** Gestión de cuenta de PO – checklist – Sprints terminados**Monitoreo de planificación y verificación****Código: 022****Nombre:** Gestión de cuenta de SM.**Reportes y resultados****Código: 023****Nombre:** Reportes para todos los usuarios

3.2.4. Pruebas del usuario:

Una vez construido, se procede a mostrársele al cliente, con las llamadas pruebas de usuario en el que se realizará una evaluación del prototipo, esto para ver si cumple con los requerimientos. Para garantizar que los diferentes módulos se integren con la aplicación nos basamos en la prueba de integración generando nuestra propia técnica basado y adaptado del modelo de la caja negra que detallaremos a continuación:

Para poder seleccionar a las personas encargadas de supervisar y evaluar los prototipos hemos tomado como referencias, distintas fuentes de búsqueda, como búsquedas en internet, preguntas a compañeros, e ingenieros del medio que desarrollan software, logrando obtener la recomendación de dos profesionales encargados de trabajar con la metodología Scrum los cuales nos brindaron la información necesaria para la evaluación de nuestros prototipos.

Se presentó un modelo inicial de prototipos, creados por nosotros mismos en base a los conocimientos previos sobre la metodología, posteriormente mediante correo se envió la información y las secuencias de los prototipos iniciales, proponiendo este diseño para que el evaluador pueda darnos sus observaciones y acotaciones.

Tabla 15: Diseño de prueba de prototipos.





Entrada	Proceso	Salida	Resultado
Ubicación o descripción de la interfaz	Interfaz del prototipo propuesto inicialmente	Observaciones	Interfaz de prototipo adaptado

Fuente: Elaboración Propia

En base a este diseño se obtuvimos respuestas de regreso por parte de los evaluadores, logrando así estructurar adecuadamente los prototipos y los cambios que realizamos fueron en base a las sugerencias descritas en cada una de las observaciones, pudiendo mostrar finalmente un prototipo mejorado y posteriormente aprobado por los evaluadores.

Se aprobaron la mayoría de los prototipos iniciales, así que solo mostraremos la secuencia y la evaluación de prototipos en los cuales se agregaron observaciones y se efectuaron cambios.

Tabla 16: Prueba de prototipos y observaciones

Entrada	Proceso	Salida	Resultado
Inicio de sesión - Pantalla de Bienvenida		<ul style="list-style-type: none"> La bienvenida debe ser corta, se sugiere retirar la lista de opciones. Crear un panel de opciones para cada tipo de usuario delimitando las interfases con las que puede interactuar. Por ejemplo, el P.O como verá los avances, al igual que el S.M. el equipo de desarrollo solo verá las Tasklist, etc. 	
Nuevo Usuario		<ul style="list-style-type: none"> Dirección de usuario, se puede obviar. Nombre de usuario aparece 2 veces, imagino para tomarlo como ID. El rol se debe poner en la parte superior para que el usuario tenga una mejor visión del registro. 	

Gestión de Roles



- Cambiar Colores u omitirlos.
- Mantener la descripción de cada uno para un mejor entendimiento del rol.



Gestión de Proyectos



- Se puede quitar la parte de Nuevo Backlog, debido a que cada proyecto tiene su backlog, en lugar de eso se propone que al momento de hacer clic en proyecto se abra el backlog ya que por cada proyecto hay un backlog definido.
- Quitar los botones del proyecto.



Backlog



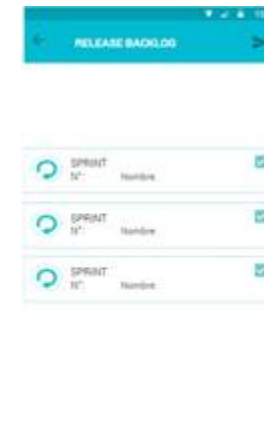
- La parte que dice Release Backlog lo puedes cambiar a simplemente Product Backlog, se sobreentiende que en la APP se colocarán los requerimientos ya priorizados y luego pasa directo al Sprint.



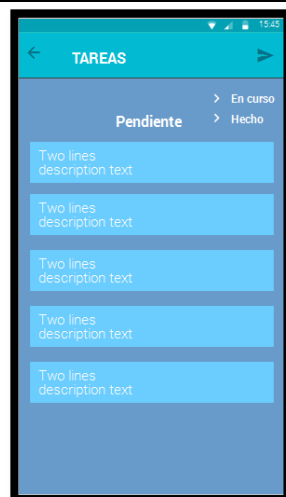
Sprint – Vista del Product Owner



- Omitir el botón de búsqueda ya que los Sprints no debe ser demasiados porque se trata de proyecto agilizados.
- Borrar el botón iniciar Sprint para el Product Owner ya que el que desarrolla esto es el Development Team.



Listado de Tareas



- No abarcar tanto centrarse en planificación y verificación, las tareas se pueden listar.
- Agregarle la opción de terminar tarea para el Development Team en lugar de abarcar todos los procesos desde el inicio hasta el final de la realización de la tarea.



Observaciones



- En la valoración como coincidimos sería mejor con si cumple o no cumple el criterio.
- Se recomienda optar por permitir al usuario seleccionar el criterio e ir evaluando cada uno de acuerdo a su criterio para agregar o no observaciones.



Fuente: Elaboración Propia

3.2.5. Implantación:

Una vez finalizado las pruebas de nuestros prototipos, se llevó a cabo el proceso de implantación donde se pone en marcha todo lo necesario para la creación de nuestra aplicación móvil.

Implementación de una Aplicación Móvil

MVP es otro patrón de diseño que tiene como objetivo separar la interfaz de usuario de la lógica de las aplicaciones.

Básicamente este patrón consiste en 3 componentes:

- La vista. Compuesta de las ventanas y controles que forman la interfaz de usuario de la aplicación.
- El modelo. Que es donde se lleva a cabo toda la lógica de negocio.
- El presentador. Escucha los eventos que se producen en la vista y ejecuta las acciones necesarias a través del modelo. Además, puede tener acceso a las vistas a través de las interfaces que la vista debe implementar. (Arrivi, 2010)

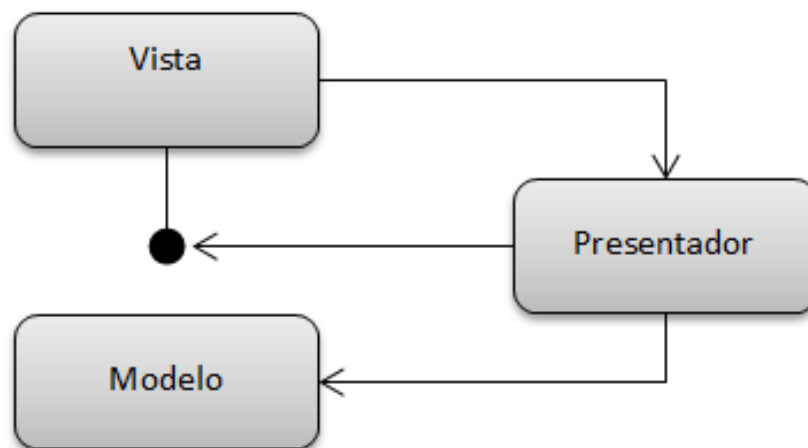


Ilustración 25: Gráfica de la distribución para arquitectura de solución.

Fuente: (Arrivi, 2010).

Lo que hace este patrón es sencillo, delega toda la lógica del negocio a una clase presentador el cual se encarga de atender las peticiones solicitadas desde la vista y dirigirlas al modelo para que este reaccione ante los eventos del usuario.

El modelo se encarga de la lógica de negocio, persistencia de datos locales (SQLite), conexión con APIS (Servicio Web), etc. La vista son todas las Interfaces (Activity, fragments) que son mostradas al usuario.

El beneficio es tener un código más ordenado e independiente del framework de Android, además se realizan pruebas más sencillas.

Implementación de una Web Service

La solución planteada requiere que se mantenga la fiabilidad y seguridad de la información en todo momento a fin de evitar posibles riesgos; como, por ejemplo, el nivel de recuperación de información ante fallas o pérdidas, o saber que usuario registro cambios en el sistema, así como el nivel de no repudio de transacciones por los usuarios. Debido a esto, se propone la implementación de un servicio web con tres capas.



Ilustración 26: Diagrama de la Web Service

Fuente: Elaboración propia.

3.2.6. Auditoría y seguimiento

Las pruebas nos sirvieron como una retroalimentación constante para corroborar el correcto funcionamiento de la aplicación y fueron ejecutadas a lo largo de todo el proceso de implementación de cada uno de los módulos. Éstas se fueron consolidando de acuerdo a las observaciones y recomendaciones de los evaluadores, siendo consideradas como exitosas si todas las pruebas son aceptadas.

El plan de Auditoria incluye los siguientes aspectos:

Tabla 17: Descripción del plan de auditoría

Título	Descripción
Tipo de auditoría	Evaluación
Fecha	03 de mayo 2018
Objetivo	Comprobar de manera eficiente el desarrollo de la aplicación móvil
alcance de la auditoria:	Primero se desarrollará una evaluación funcional, pasando por análisis de procesos y corroborando si los resultados mostrados se cumplen o no de acuerdo con los requisitos iniciales.
Duración de la auditoría.	1 semana
Equipo Auditor.	Ing. Jesús Miguel Quispe Paredes CIP: 178881 Ing. Fátima Romero Medina - Scrum Master en los proyectos de Data – BCP - Lima.
Procesos y actividades que auditar.	Todos los procesos que involucra el desarrollo de la aplicación móvil para la planificación y verificación de Historias de Usuario basado en metodología Scrum.

Fuente: Adaptado de (Giraldo Rivera & Martínez Pérez, 2015)

Especificaremos en detalle cada proceso auditado y estructurado de la siguiente manera:

Tabla 18: Auditoría y seguimiento al desarrollo de la aplicación móvil.

Requerimientos iniciales			Prueba de requerimientos en la aplicación	Cumple Si / No
Código	Nombre	Descripción		
R-001	Inicio de Sesión	El administrador al ingresar por primera vez tendrá que registrar los datos de la empresa, y crear una cuenta de administrador.	Nombre: Inicio de sesión - Registro Entradas: Datos de la empresa. Proceso: Se determinará a la persona que desarrollará el papel de administrador, para poder realizar el registro, ya definido su rol deberá ir a la Opción “Registrarse”, e ingresará los datos de la empresa solicitados como Nombre de empresa, dirección, correo, Nombre de usuario y contraseña para poder completar el registro. Resultados esperados: Completados todos los datos se muestra un mensaje de “Registro exitoso”, como una confirmación de cuenta y acceso a la aplicación móvil.	Si cumple
			Nombre: Gestión de Cuenta de Administrador – Perfil de empresa y gestión de roles Entradas: Usuario y contraseña de administrador Proceso: Desempeñando el rol de administrador se habilitará 3 pantallas: Gestión de usuarios, perfil de empresa y gestión de roles, las dos últimas sólo son informativas y sólo podrán ser visualizadas sin poder realizar cambios. Resultados esperados: Visualización del perfil de empresa, gestión de roles y gestión de usuarios.	
R-002	Cuenta de Administrador	El administrador podrá tener acceso a las siguientes interfaces: Perfil de Empresa, Gestión de roles, gestión de usuarios.		Si cumple

R-003	Perfil de Empresa	Se mostrará la información general de la empresa y beneficios de HUS, para todos los usuarios.	<p>Nombre: Gestión de Cuenta de Administrador – Perfil de empresa.</p> <p>Entradas: Usuario y contraseña de administrador</p> <p>Proceso: El administrador y todos los usuarios al momento de iniciar sesión se les mostrará una pantalla de bienvenida.</p> <p>Resultados esperados: Se muestra una interfaz de bienvenida para todos los usuarios.</p>	Si cumple
R-004	Gestión de usuarios	El administrador podrá crear, editar y eliminar usuarios para 3 roles: Product Owner, Scrum Master y Development Team	<p>Nombre: Gestión de Cuenta de Administrador – Gestión de usuarios.</p> <p>Entradas: Datos de los encargados que van a trabajar en los proyectos, Puede crear solo un Product Owner, 01 Scrum Master y un máximo de 5 personas en el equipo de desarrollo.</p> <p>Proceso: El administrador agregará los datos de los usuarios uno a uno tomando en cuenta, el Rol, nombres, apellidos, dirección, correo, nombre de usuario y contraseña.</p> <p>Resultados esperados: Lista de usuarios que participarán en los proyectos con sus respectivos roles, nombres de usuario y contraseña.</p>	Si cumple
R-005	Cuenta de Product Owner	El Product Owner podrá tener acceso a Perfil de empresa, gestión de proyectos checklist y reportes.	<p>Nombre: Gestión de cuenta de Product Owner</p> <p>Entradas: Nombre de usuario y contraseña.</p> <p>Proceso: El Product Owner determinado como tal, podrá tener acceso a Perfil de empresa (modo Visualización), Gestión de proyectos y checklist.</p> <p>Resultados esperados: Tener visualización del perfil de empresa, poder gestionar proyecto y gestionar checklist.</p>	Si cumple

R-006	Gestión de proyecto	El Product owner podrá crear, editar y eliminar proyectos, así como asignar a usuarios de Development Team a proyectos creados.	Nombre: Gestión de cuenta de Product Owner - Gestión de proyecto. Entradas: Datos del proyecto. Proceso: El Product Owner va a la interfaz de Gestión de Proyectos, donde deberá registrar los datos solicitados, seleccionar una fecha de inicio y finalmente crear el proyecto. Resultados esperados: El Product owner crea proyectos.	Si cumple
			Nombre: Gestión de cuenta de Product Owner – Asignar personal Entradas: Tener proyectos creados y seleccionados. Proceso: En este proceso el PO seleccionará un proyecto y se desplegará la lista de desarrolladores para ser asignados. Resultados esperados: El PO podrá asignar personal del Equipo de desarrollo a un proyecto, también podrá editar y eliminar proyectos.	
R-007	Creación de HU en el backlog	El Product owner podrá registrar, editar y eliminar historias de usuario en el backlog.	Nombre: Gestión de cuenta de Product Owner – Creación de HU en el backlog Entradas: Código, título, seguido de las premisas Yo Como, necesito, para; según lo establecido anteriormente. Proceso: En este proceso el PO selecciona el ícono de agregar historias de usuario, seguirá ingresando la secuencia antes señalada e irá agregando las historias las historias de usuario una a una. Resultados esperados: El PO generar una lista de historias de usuario en el backlog.	Si cumple

R-008	Creación de Criterios de Aceptación.	el Product Owner podrá agregar los criterios de aceptación a las historias de usuario en el backlog.	<p>Nombre: Gestión de cuenta de Product Owner – Gestión de Historias de Usuario - Criterios de Aceptación.</p> <p>Entradas: Código y descripción</p> <p>Proceso: En este proceso el PO deberá seleccionar una historia de usuario, “Ver criterios”, se desplegará una pantalla que permitirá agregar criterios de aceptación siguiendo la siguiente estructura: Código y Descripción. Los criterios de aceptación se irán guardando internamente y aparecerán como números en las Historias de usuario.</p> <p>Resultados esperados: Product Owner agregará los criterios de aceptación de las historias de usuario en el backlog además de poder editar, eliminar y posteriormente priorizar Historias de Usuario.</p>	Si cumple
R-009	Priorización de Historias de Usuario	el Product Owner podrá priorizar (alta, media y baja) y ver los criterios de aceptación de las historias de usuario en el backlog.	<p>Nombre: Gestión de cuenta de Product Owner – Gestión de Historias de Usuario - Priorización.</p> <p>Entradas: Para priorizar las Historias de Usuario debemos seleccionar la historia de usuario.</p> <p>Proceso: En este proceso el PO deberá seleccionar una historia de usuario, para poder asignarle una prioridad, que vaya de acorde al rango establecido: “Alta”, “Media” o “Baja”. Además de poder modificar, eliminar y poder visualizar los Criterios de Aceptación en cada HU.</p> <p>Resultados esperados: El PO podrá priorizar una lista de historias de usuario en el backlog de acuerdo con el nivel de priorización que el PO considere. Además de poder modificar, eliminar y poder visualizar los Criterios de Aceptación.</p>	Si cumple

R-010	Creación de Sprints.	el Product Owner podrá crear Sprints de acuerdo a la prioridad de las historias de usuario, las HU de más alta prioridad irán en el Primer Sprint y así sucesivamente.	<p>Nombre: Gestión de cuenta de Product Owner – Gestión de Sprints – Creación de Sprints.</p> <p>Entradas: Historias de Usuario priorizadas con sus criterios de aceptación.</p> <p>Proceso: En este proceso el PO deberá seleccionar “Nuevo Sprint”, se cargará las historias de usuario en orden desde la más alta a baja, podrá seleccionar las HU que irán en el primer Sprint y así podrá ir seleccionando HU de acuerdo a la prioridad hasta que todas estén asignadas a un sprint.</p> <p>Resultados esperados: El PO creará y visualizará los Sprints con las historias de usuario priorizadas desde las HU con prioridad alta, media y baja secuencialmente.</p>	Si cumple
R-011	Duración por Sprint	el Product Owner podrá asignarle un tiempo de desarrollo (semanas) a cada sprint.	<p>Nombre: Gestión de cuenta de Product Owner – Gestión de Sprints – Duración de Sprints.</p> <p>Entradas: Sprints asignados por proyecto.</p> <p>Proceso: En este proceso el PO deberá seleccionar sprint por sprint para poder asignarle un tiempo entre el siguiente rango: 1 – 5 semanas, una vez asignada internamente se guardará la información y se cargará los Sprints con la duración seleccionada.</p> <p>Resultados esperados: El PO asignará una duración cada uno de los Sprint, se visualizará la lista de Sprint y el tiempo de duración por cada uno de ellos.</p>	Si cumple

R-012	Duración por proyectos	La aplicación debe ser capaz de calcular la duración del proyecto (semanas) en base al tiempo especificado en cada sprint.	<p>Nombre: Gestión de cuenta de Product Owner – Gestión de Proyecto – Duración por proyecto.</p> <p>Entradas: Lista de Sprints con una duración asignada en semanas.</p> <p>Proceso: En este proceso el PO al tener una lista de Sprints con su respectiva duración en semanas regresará a la pantalla del proyecto donde internamente se realizará el cálculo sumando el total de semanas por sprint contenidos en ese proyecto, logrando así la estimación total del proyecto en semanas.</p> <p>Resultados esperados: El PO visualizará la duración del proyecto en semanas.</p>	Si cumple
R-013	Generación de Reportes	En la lista de reportes se debe mostrar para todos los usuarios, excepto el administrador.	<p>Nombre: Reportes para todos los usuarios</p> <p>Entradas: Usuario y contraseña como SM, PO y TD respectivamente.</p> <p>Proceso: En este proceso el SM, PO y TD luego de iniciar sesión tendrá la opción de ver reportes en el panel lateral donde podrán visualizar, un listado de tres reportes para poder conocer todo el proceso de desarrollo del proyecto, se restringe el acceso al administrador ya que es el encargado de gestionar usuarios, más no de evaluar ni desarrollar actividades propias de Scrum.</p> <p>Resultados esperados: El SM, PO y TD Visualizarán una lista de reportes como: Product Backlog, Listado de tareas por usuario y listado de criterios de aceptación, todos estos reportes están en formato PDF y se podrán descargar, imprimir, o abrir de acuerdo crean conveniente los usuarios.</p>	Si cumple

R-014	Checklist	El Product Owner podrá ver los proyectos a verificar, así como los Sprints relacionados para realizar la verificación de criterios.	Nombre: Gestión de cuenta de PO – checklist Entradas: Inicio de sesión del PO Proyecto, historias de usuario y Sprints. Proceso: En este proceso el PO accediendo al checklist se desplegará un menú con la lista de proyectos, se podrá seleccionar el proyecto que desea verificar, y se mostrarán los Sprints del proyecto.	Si cumple
			Resultados esperados: Visualización de los Sprints en el proyecto con la cantidad de historias de usuario que contiene.	
R-015	Validación de Criterios de Aceptación.	Se cargarán las historias de usuario y los criterios de aceptación por historia para seleccionar las que cumplen y las que no cumplen.	Nombre: Gestión de cuenta de Product Owner – checklist– criterios de aceptación Entradas: Sprint, historias de usuario y criterios de aceptación. Proceso: En este proceso el PO accediendo al checklist se desplegará el sprint seleccionado, se mostrarán los criterios con la opción de validar con checklist los criterios que para el PO llegaron a cumplirse.	Si cumple
			Resultados esperados: Visualización de los Sprints en el proyecto con la cantidad de historias de usuario y aprobación del cumplimiento de los criterios de aceptación establecidos.	
R-015	Validación de Criterios de Aceptación.	Se cargarán las historias de usuario y los criterios de aceptación por historia para seleccionar las que cumplen y las que no cumplen.	Nombre: Gestión de cuenta de PO – checklist – Sprints terminados Entradas: Historias de usuario y criterios de aceptación terminados. Proceso: En este proceso el PO dará la conformidad del desarrollo de los Sprints pudiendo dar por terminados los que crea convenientes.	Si cumple
			Resultados esperados: El PO podrá dar por terminado Sprints.	

R-016	Agregar observaciones	Si el Product owner no está conforme con alguno de los criterios, podrá agregar observaciones, las que se mostrarán al visualizar una historia y en los reportes.	<p>Nombre: Gestión de cuenta de PO – checklist –Observaciones en los criterios de aceptación</p> <p>Entradas: Historias de usuario y criterios de aceptación terminados.</p> <p>Proceso: En este proceso el PO dará la conformidad del desarrollo de los criterios de aceptación y tendrá la posibilidad de agregar una observación de ser necesario si no cumplió del todo lo que inicialmente pidió, de no ser considerados los criterios de aceptación como aceptados, se debe considerar la historia de usuario para el próximo Sprint.</p> <p>Resultados esperados: El PO podrá dar por aprobado los criterios de aceptación y además podrá agregar observaciones adicionales.</p>	Si cumple
R-017	Cuenta de Equipo de desarrollo	El Development Team podrá tener acceso a Perfil de empresa, gestión de tareas y reportes.	<p>Nombre: Gestión de cuenta de Equipo de desarrollo</p> <p>Entradas: Nombre de usuario y contraseña.</p> <p>Proceso: En este proceso el usuario asignado como parte del equipo de desarrollo deberá iniciar sesión, una vez validados sus datos podrá tener acceso al panel lateral con las opciones de Perfil de empresa (modo Visualización), Gestión de tareas restringida sólo al Development Team y lista de reportes que serán mostrados a los demás usuarios que desempeñen un rol en Scrum.</p> <p>Resultados esperados: Podrá visualizar Perfil de empresa, gestión de proyecto al que está asignado y los reportes (Product Backlog, Listado de tareas por usuario y listado de criterios de aceptación)</p>	Si cumple

R-018	Gestión de tareas	El Development Team podrá visualizar los proyectos al cual ha sido asignado, visualizar los Sprints del proyecto y las historias de usuario por sprint.	<p>Nombre: Gestión de cuenta de TD - Gestión de Sprints</p> <p>Entradas: Nombre de usuario y contraseña.</p> <p>Proceso: En este proceso el TD deberá seleccionar el Sprint a trabajar, se visualizará el Porcentaje de avance, la historia de Usuario con su código y nombre, la cantidad de criterios de aceptación por HU y finalmente la cantidad de tareas por HU.</p> <p>Resultados esperados: Podrá visualizar los Sprints, las historias de usuario, el Porcentaje de avance, cantidad de criterios y podrá agregar tareas.</p>	Si cumple
R-019	Creación de tareas	El equipo de desarrollo podrá crear, editar y terminar tareas por cada historia de usuario.	<p>Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de Sprints – Creación de tareas.</p> <p>Entradas: Código y descripción de las tareas.</p> <p>Proceso: En este proceso el TD deberá seleccionar la historia de usuario a trabajar, se desplegará una interfaz para agregar tareas, seleccionamos el ícono de agregar y llenaremos los datos para registrar las tareas.</p> <p>Resultados esperados: Podrá crear y visualizar la lista de tareas asignadas a cada HU.</p>	Si cumple
			<p>Nombre: Gestión de cuenta de TD – Gestión de tareas</p> <p>Entradas: Lista de Tareas.</p> <p>Proceso: En este proceso el TD podrá seleccionar una tarea para poder “terminar” si el desarrollo de la misma se aprobó y se da por</p>	Si cumple

			<p>finalizado; del mismo modo la tarea puede ser modificada y eliminada siempre y cuando no haya sido terminada.</p> <p>Resultados esperados: Podrá terminar, modificar y eliminar las tareas asignadas a cada HU.</p>	
R-020	Avance de tareas	Se visualizará las historias de usuario con su porcentaje de avance de acuerdo a las tareas terminadas.	<p>Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de tareas - Avance</p> <p>Entradas: Lista de HU con sus respectivas tareas.</p> <p>Proceso: En este proceso el usuario asignado como parte del equipo de desarrollo podrá seleccionar una tarea para poder “terminar”, de acuerdo al avance de las tareas el porcentaje irá cambiando hasta que las tareas de las historias de usuario hayan sido totalmente completadas.</p> <p>Resultados esperados: Visualización general de la historia de usuario, número de criterios, número de tareas y el porcentaje de avance de las tareas con sus respectivas historias de usuario.</p>	Si cumple
R-021	Cuenta de Scrum master	El scrum master podrá tener acceso a todo para poder visualizar, monitorear y hacer seguimiento a los procesos alineándolos a SCRUM,	<p>Nombre: Gestión de cuenta de SM.</p> <p>Entradas: Usuario y contraseña.</p> <p>Proceso: En este proceso el SM podrá visualizar, editar y eliminar historias de usuario, Sprints, criterios que considere alejados de los conceptos manejados por SCRUM.</p> <p>Resultados esperados: El SM tendrá acceso a todos los procesos, menos a los procesos realizados por el administrador.</p>	Si cumple

Fuente: Desarrollo Propio

3.4. ESPECIFICACIONES TÉCNICAS DE LA APLICACIÓN MÓVIL

3.3.1. Entorno de desarrollo

Para el desarrollo de este proyecto, se usó el entorno de desarrollo Android Studio, es el entorno de desarrollo integrado oficial para la plataforma Android llegando a reemplazar a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. Es un entorno flexible que nos permite un mejor desarrollo de aplicaciones Android.

3.3.2. Lenguaje en el que se desarrolló

El lenguaje de programación que se usó es Java, lo usamos dado que es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Permitiendo tener un alcance más amplio.

Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo, permitiendo crear código que puede ser ejecutado en una plataforma y no tiene que ser recompilado para correr en otra. Además de eso que Android es el lenguaje oficial para plataformas Android.

3.3.3. Base de datos

La base de datos que se usó fue PostgreSQL, ya que es un Sistema de gestión de bases de datos relacional orientado a objetos y libre, publicado bajo la licencia PostgreSQL.

3.4. DESCRIPCIÓN GENERAL DE LAS EMPRESAS DE CHICLAYO

Las empresas de desarrollo en la ciudad de Chiclayo están en auge, ya que la necesidad de implementar sistemas informáticos va a en aumento por parte de los empresarios en esta ciudad. Es importante tomar en cuenta que las empresas en las que probamos nuestra aplicación son empresas desarrolladoras de software y empresas que tienen un área de desarrollo integrado y que trabajan sus proyectos bajo la metodología SCRUM, para poder hacer un análisis de planificación y verificación de las historias de usuario. Por lo que hemos tomado proyectos y los analizamos como casos para tener una mejor visión de la importancia de nuestra aplicación en empresas reales.

Se realizó una búsqueda exhaustiva para poder localizar empresas del medio que desarrollen software siguiendo los procesos de la metodología Scrum, pero no se logró a muchas siendo estas dos las indicadas para implementar y probar la aplicación final. Ver ANEXO 02.

3.4.1. Empresas de Chiclayo

Las empresas con las que trabajamos son las siguientes, analizamos el modo en el que usa Scrum dentro del desarrollo de sus proyectos.

Datos de las empresas:

NOMBRE : Upson Systems S. A.
DIRECCIÓN : Los pinos de la plata Bloque Y N° 502
AÑOS DE CREACIÓN : 5 años.
SECTOR DE ACTIVIDAD: Desarrollo de Software

NOMBRE : Despensa Peruana S. A.
DIRECCIÓN : Av. Salaverry 1145
AÑOS DE CREACIÓN : 25 años.
SECTOR DE ACTIVIDAD: Distribución y abastecimiento de abarrotes.

3.4.2. Rubro

Para poder analizar Scrum, se decidió tomar empresas de desarrollo de software como empresas que tienen un área de sistemas integrada para desarrollar sus propios proyectos, ya que necesitamos ver como aplica Scrum en sus productos de Software y proyectos, los beneficios que trae la planificación y la verificación por parte del cliente.

3.4.3. Metodología de desarrollo de software

Hemos tomado como base a estas empresas dado que usan SCRUM como metodología ágil, para el desarrollo de proyectos ágiles.

3.4.4. Proyectos

Los proyectos que analizamos son los siguientes y expondremos el uso de la metodología a detalle en cada uno de ellos.

Proyecto 001: Toma de pedidos móvil

Descripción del proyecto:

Este es un proyecto corto desarrollado por el equipo de desarrollo de Despensa Peruana, que sirve de apoyo a los vendedores en el registro de sus pedidos en el punto de venta (cliente).

Requerimientos iniciales: Estos requerimientos son los que el Product Owner tiene como referencia para el desarrollo del proyecto. En la mayoría de los casos estas son anotaciones básicas de las características que debe tener este sistema de venta de comida rápida.

La lista de requerimientos es la siguiente:

- Tener un listado de los clientes de la ruta asignada al vendedor por día.
- Tener un listado de las condiciones de pago.
- Tener una lista de todos los artículos que vende la empresa.
- Buscar clientes y artículos.
- Consultar los pedidos realizados en el día.
- Dar la opción de anular el pedido.
- Registro del porque un cliente no realizo un pedido.

Product backlog

Una vez el Product Owner tuvo una lista de los requerimientos iniciales, se hizo una reunión con el Scrum Master y el equipo de desarrollo, mostró la lista, se realizó un dialogo y se diseñó una lista mejorada de estos requerimientos, que pasaron a ser Historias de Usuario, del mismo modo se enumeraron, y se delimitaron los criterios de aceptación de cada historia. Para este caso vamos a tomar las historias de usuario establecidas en cuadro para un mejor entendimiento la estructura física.

Tabla 19: Lista inicial de historias de usuario y criterios de aceptación – Caso 01

Código	Título	Historia de Usuario	Criterio de Aceptación
001	Inicio de sesión de los vendedores.	- Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos.	<ul style="list-style-type: none"> - Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.
002	Módulo de registro de pedidos.	- Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta.	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.
003	Módulo de registro de no pedido.	- Como vendedor necesito registrar el motivo de porque un cliente no realiza un pedido para tener un mejor histórico del cliente.	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar el motivo del no pedido. - En caso el motivo en la lista, digitarlo. - Registro del no pedido.
004	Módulo de consulta de pedidos registrados en el día.	- Como vendedor necesito consultar mis pedidos para visualizar una lista de todos los pedidos registrados en el día.	<ul style="list-style-type: none"> - Buscar los pedidos por fechas. - Visualizar el monto total a pagar de cada pedido, cuantos pedidos se registraron y una sumatoria del monto total a pagar de los pedidos.

005	Anulación de pedidos mal registrados.	<ul style="list-style-type: none"> - Como vendedor necesito anular los pedidos mal registrados para evitar problemas con los clientes. 	<ul style="list-style-type: none"> - Listar los pedidos registrados. - Opción de anular por cada pedido.
001	Inicio de sesión de los vendedores.	<ul style="list-style-type: none"> - Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos. 	<ul style="list-style-type: none"> - Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.
002	Módulo de registro de pedidos.	<ul style="list-style-type: none"> - Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta. 	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.
003	Módulo de registro de no pedido.	<ul style="list-style-type: none"> - Como vendedor necesito registrar el motivo de porque un cliente no realiza un pedido para tener un mejor histórico del cliente. 	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar el motivo del no pedido. - En caso el motivo en la lista, digitarlo. - Registro del No pedido.
004	Módulo de consulta de pedidos registrados en el día.	<ul style="list-style-type: none"> - Como vendedor necesito consultar mis pedidos para visualizar una lista de todos los pedidos registrados en el día. 	<ul style="list-style-type: none"> - Buscar los pedidos por fechas.

			<ul style="list-style-type: none"> - Visualizar el monto total a pagar de cada pedido, cuantos pedidos se registraron y una sumatoria del monto total a pagar de los pedidos.
005	Anulación de pedidos mal registrados.	<ul style="list-style-type: none"> - Como vendedor necesito anular los pedidos mal registrados para evitar problemas con los clientes. 	<ul style="list-style-type: none"> - Listar los pedidos registrados. - Opción de anular por cada pedido.
001	Inicio de sesión de los vendedores.	<ul style="list-style-type: none"> - Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos. 	<ul style="list-style-type: none"> - Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.
002	Módulo de registro de pedidos.	<ul style="list-style-type: none"> - Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta. 	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.
003	Módulo de registro de no pedido.	<ul style="list-style-type: none"> - Como vendedor necesito registrar el motivo de porque un cliente no realiza un pedido para tener un mejor histórico del cliente. 	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar el motivo del no pedido. - En caso el motivo en la lista, digitarlo. - Registro del No pedido.

004	Módulo de consulta de pedidos registrados en el día.	<ul style="list-style-type: none"> - Como vendedor necesito consultar mis pedidos para visualizar una lista de todos los pedidos registrados en el día. 	<ul style="list-style-type: none"> - Buscar los pedidos por fechas. - Visualizar el monto total a pagar de cada pedido, cuantos pedidos se registraron y una sumatoria del monto total a pagar de los pedidos.
005	Anulación de pedidos mal registrados.	<ul style="list-style-type: none"> - Como vendedor necesito anular los pedidos mal registrados para evitar problemas con los clientes. 	<ul style="list-style-type: none"> - Listar los pedidos registrados. - Opción de anular por cada pedido.
001	Inicio de sesión de los vendedores.	<ul style="list-style-type: none"> - Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos. 	<ul style="list-style-type: none"> - Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.
002	Módulo de registro de pedidos.	<ul style="list-style-type: none"> - Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta. 	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.

Fuente: Desarrollo Propio.

Tabla 20: Priorización de historias de usuario

Código	Título	Historia de Usuario	Criterio de Aceptación	Prioridad
001	Inicio de sesión de los vendedores.	- Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos.	- Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.	ALTA
002	Módulo de registro de pedidos.	- Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta.	- Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.	ALTA
003	Módulo de registro de no pedido.	- Como vendedor necesito registrar el motivo de porque un cliente no realiza un pedido para tener un mejor histórico del cliente.	- Seleccionar cliente. - Seleccionar el motivo del no pedido. - En caso el motivo en la lista, digitarlo. - Registro del No pedido.	MEDIA
004	Módulo de consulta de pedidos registrados en el día.	- Como vendedor necesito consultar mis pedidos para visualizar una lista de todos los pedidos registrados en el día.	- Buscar los pedidos por fechas. - Visualizar el monto total a pagar de cada pedido, cuantos pedidos se registraron y una sumatoria del monto total a pagar de los pedidos.	ALTA

005	Anulación de pedidos mal registrados.	- Como vendedor necesito anular los pedidos mal registrados para evitar problemas con los clientes.	- Listar los pedidos registrados. - Opción de anular por cada pedido.	MEDIA
001	Inicio de sesión de los vendedores.	- Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos.	- Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.	ALTA
002	Módulo de registro de pedidos.	- Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta.	- Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.	ALTA
003	Módulo de registro de no pedido.	- Como vendedor necesito registrar el motivo de porque un cliente no realiza un pedido para tener un mejor histórico del cliente.	- Seleccionar cliente. - Seleccionar el motivo del no pedido. - En caso el motivo en la lista, digitarlo. - Registro del No pedido.	MEDIA
004	Módulo de consulta de pedidos	- Como vendedor necesito consultar mis pedidos para visualizar una lista de todos los pedidos registrados en el día.	- Buscar los pedidos por fechas. - Visualizar el monto total a pagar de cada pedido, cuantos pedidos se registraron y una	ALTA

	registrados en el día.		sumatoria del monto total a pagar de los pedidos.	
005	Anulación de pedidos mal registrados.	- Como vendedor necesito anular los pedidos mal registrados para evitar problemas con los clientes.	- Listar los pedidos registrados. - Opción de anular por cada pedido.	MEDIA
001	Inicio de sesión de los vendedores.	- Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos.	- Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.	ALTA
002	Módulo de registro de pedidos.	- Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta.	- Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.	ALTA
003	Módulo de registro de no pedido.	- Como vendedor necesito registrar el motivo de porque un cliente no realiza un pedido para tener un mejor histórico del cliente.	- Seleccionar cliente. - Seleccionar el motivo del no pedido. - En caso el motivo en la lista, digitarlo. - Registro del No pedido.	MEDIA

004	Módulo de consulta de pedidos registrados en el día.	- Como vendedor necesito consultar mis pedidos para visualizar una lista de todos los pedidos registrados en el día.	- Buscar los pedidos por fechas. - Visualizar el monto total a pagar de cada pedido, cuantos pedidos se registraron y una sumatoria del monto total a pagar de los pedidos.	ALTA
005	Anulación de pedidos mal registrados.	- Como vendedor necesito anular los pedidos mal registrados para evitar problemas con los clientes.	- Listar los pedidos registrados. - Opción de anular por cada pedido.	MEDIA
001	Inicio de sesión de los vendedores.	- Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos.	- Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas.	ALTA
002	Módulo de registro de pedidos.	- Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta.	- Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido.	ALTA

Fuente: Desarrollo Propio.

Tabla 21: Ordenamiento de Historias por Prioridad – Caso 01

Código	Título	Historia de Usuario	Prioridad
001	Inicio de sesión de los vendedores.	- Como vendedor necesito poder ingresar en la aplicación con mi usuario y clave respectiva para registrar los pedidos.	ALTA
002	Módulo de registro de pedidos.	- Como vendedor necesito registrar los pedidos de los clientes que se encuentren en mi ruta para cubrir la necesidad diaria de venta.	ALTA
004	Módulo de consulta de pedidos registrados en el día.	- Como vendedor necesito consultar mis pedidos para visualizar una lista de todos los pedidos registrados en el día.	ALTA
003	Módulo de registro de no pedido.	- Como vendedor necesito registrar el motivo de porque un cliente no realiza un pedido para tener un mejor histórico del cliente.	MEDIA
005	Anulación de pedidos mal registrados.	- Como vendedor necesito anular los pedidos mal registrados para evitar problemas con los clientes.	MEDIA

Fuente: Desarrollo Propio.

Separación por Sprints

Como detallamos anteriormente tomaremos las siguientes historias de usuario de acuerdo con la prioridad asignada por el Product Owner. Una vez tenemos establecidos los Sprints le asignaremos un tiempo, este tiempo lo consignaremos en semanas para poder calcular cuánto tiempo demora realizar el proyecto.

SPRINT 01

Ya se cuenta con una base de datos para poder iniciar con el desarrollo, para lo cual en este diseño se contará con el personal que labora en la empresa (supervisores, vendedores), quienes informarán sobre los datos que son importantes para la empresa.

Tabla 22: Primer Sprint – Caso 01

Código	Título	Prioridad
001	Inicio de sesión de los vendedores.	ALTA
002	Módulo de registro de pedidos.	ALTA
004	Módulo de consulta de pedidos registrados en el día.	ALTA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 3 semanas.

SPRINT 02

Esta iteración trabajaremos las siguientes historias de usuario, relacionadas con el registro de no pedido.

Tabla 23: Segundo Sprint – Caso 01

Código	Título	Prioridad
003	Módulo de registro de no pedido.	MEDIA
005	Anulación de pedidos mal registrados.	MEDIA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 2 semanas

Desglose en tareas a cargo del equipo de desarrollo:

El equipo de desarrollo teniendo acceso a la información y lista de Historias de usuario, desglosara las HU en tareas de acuerdo a su experiencia en desarrollo.

Tabla 24: Desglose de HU en tareas-Caso 01

N° SPRINT	CÓDIGO	NOMBRE HU	TAREAS
1	001	Inicio de sesión de los vendedores.	Creación de la interfaz de acceso a la aplicación.
			Carga los datos del vendedor, así como la ruta asignada.
	002	Módulo de registro de pedidos.	Creación de la interfaz para seleccionar un cliente.
			Carga de clientes en base a la ruta asignada.
			Carga de lista de precios y condición de pago en base al cliente seleccionado.
			Crear interfaz para seleccionar y agregar los artículos al carrito de ventas, así como la cantidad solicitada.
			Visualizar el pedido completo antes de guardarlo.
			Creación de interfaz para listar los pedidos registrados.
	004	Módulo de consulta de pedidos registrados en el día.	Buscar por fechas los pedidos.
			Mostar la suma del monto total de pedidos registrados.
2	003	Módulo de registro de no pedido.	Creación de interfaz para registrar los no pedidos.
			Listar los motivos por los que no se realizan los pedidos.
			Buscar y seleccionar el cliente.
			Registrar el no pedido.

005	Anulación de pedidos mal registrados.	Creación de la interfaz de listado de pedidos registrados.
		Filtrar pedidos por fecha y que los pedidos se anulen solo si están con estado ingresado.
		Opción para anular el pedido y mostrar el pedido anulado.

Fuente: Desarrollo Propio.

Avance de tareas: El equipo de desarrollo será el encargado de desarrollar las tareas y delimitar el avance y el estado de cada una de ellas como se detalla a continuación.

Tabla 25: Porcentaje de avance de tareas – Caso 01

N° SPRINT	CÓDIGO	NOMBRE HU	TAREAS	ESTADO	% DE AVANCE
1	001	Inicio de sesión de los vendedores.	Creación de la interfaz de acceso a la aplicación.	T	100%
			Carga los datos del vendedor, así como la ruta asignada.	T	
	002	Módulo de registro de pedidos.	Creación de la interfaz para seleccionar un cliente.	T	100%
			Carga de clientes en base a la ruta asignada.	T	
			Carga de lista de precios y condición de pago en base al cliente seleccionado.	T	
			Crear interfaz para seleccionar y agregar los artículos al carrito de ventas, así como la cantidad solicitada.	T	
			Visualizar el pedido completo antes de guardarlo.	T	

004	Módulo de consulta de pedidos registrados en el día.	Creación de interfaz para listar los pedidos registrados.	T	100%
		Buscar por fechas los pedidos.	T	
		Mostar la suma del monto total de pedidos registrados.	T	
003	Módulo de registro de no pedido.	Creación de interfaz para registrar los no pedidos.	T	100%
		Listar los motivos por los que no se realizan los pedidos.	T	
		Buscar y seleccionar el cliente.	T	
		Registrar el no pedido.	T	
005	Anulación de pedidos mal registrados.	Creación de la interfaz de listado de pedidos registrados.	T	100%
		Filtrar pedidos por fecha y que los pedidos se anulen solo si están con estado ingresado.	T	
		Opción para anular el pedido y mostrar el pedido anulado.	T	

Fuente: Desarrollo Propio.

Checklist: Es una lista detallada de los criterios de aceptación fijados por el PO en un inicio junto con las historias de usuario, el PO deberá dar la conformidad y aprobación si algo no resulta como lo solicitó debe agregar observaciones detalladas del cambio, para que el equipo de desarrollo sepa si eso fue lo que se solicitó o se deben hacer cambios. Para este caso los criterios de aceptación son los siguientes:

Tabla 26: Verificación de Criterios de aceptación - Caso 01

Sprint	Código de HU	Nombre de HU	Criterio de Aceptación	Cumplimiento		Observaciones
				si	no	
	001	Inicio de sesión de los vendedores.	<ul style="list-style-type: none"> - Campos para digitar el usuario y clave. - Botón para ingresar a la interfaz de registro de pedidos. - Validar si el usuario tiene rutas asignadas. 	x		
	002	Módulo de registro de pedidos.	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar condición de pago. - Seleccionar artículos y mostrar los precios respectivos. - Ingresar la cantidad solicitado por cada artículo. - Visualizar el precio total a pagar. - Registrar el pedido. 	x		
	004	Módulo de consulta de pedidos registrados en el día.	<ul style="list-style-type: none"> - Seleccionar cliente. - Seleccionar el motivo del no pedido. - En caso el motivo en la lista, digitarlo. - Registro del No pedido. 	x		
	003	Módulo de registro de no pedido.	<ul style="list-style-type: none"> - Buscar los pedidos por fechas. - Visualizar el monto total a pagar de cada pedido, cuantos pedidos se registraron y una sumatoria del monto total a pagar de los pedidos. 		x	Extender la búsqueda a fecha y número de pedido
	005	Anulación de pedidos mal registrados.	<ul style="list-style-type: none"> - Listar los pedidos registrados. - Opción de anular por cada pedido. 	x		

Fuente: Desarrollo Propio.

Proyecto 002: Proyecto – Venta De Clínica De Nutrición

Descripción del proyecto:

SISNUP – Nutrición: Es un software administrativo, desarrollado por Upson System y que se compone de distintos módulos de acuerdo a las necesidades de cada cliente y empresa.

Los módulos son: Administración General, Mantenimiento, Citas Médicas Operaciones, Utilitarios y Reportes, se necesita este sistema para un mejor control de los trabajadores a cargo y de nuestros clientes.

Requerimientos iniciales: En una reunión con el Product Owner, el cliente del sistema, le explicó las características básicas que debe tener el sistema para los siguientes requerimientos:

- Necesito tener un sistema que controle el ingreso al sistema por los usuarios.
- Tener una lista de todos mis clientes.
- Tener una lista de todas las citas que programen mis clientes.
- Tener una historia clínica de mis pacientes.
- Tener un reporte de la historia clínica de los pacientes.
- Imprimir historias clínicas.
- Generar reservas para que a los pacientes puedan hacer sus consultas.
- Tener una lista de egresos y dinero en caja.
- Tener una lista de los especialistas en nutrición.
- Tener un reporte de egresos, ingresos y caja.

Product backlog

Una vez que el Product Owner tiene una lista de los requerimientos iniciales, se hizo una reunión con el Scrum Master y el equipo de desarrollo, mostró la lista, se realizó un dialogo y se diseñó una lista mejorada de estos requerimientos, que pasaron a ser Historias de Usuario, del mismo modo se enumeraron, y se delimitaron los criterios de aceptación de cada historia. Para este caso vamos a tomar las historias de usuario establecidas en cuadro para un mejor entendimiento la estructura física.

Tabla 27: Lista inicial de HU y Criterios de aceptación - Caso 02

Código	Título	Historia de Usuario	Criterio de Aceptación
001	Registro de cuenta administrador	- Como administrador necesito poder registrarme en el sistema para poder controlar las actividades en la clínica.	<ul style="list-style-type: none"> - Registrar datos como nombre, apellidos, fecha de ingreso, DNI, tipo de usuario. - Debe tener un usuario de administrador y una contraseña. - Tener todos los permisos habilitados en el sistema. - Debe permitir visualizar y modificar permisos en todos los módulos.
002	Registro de especialistas	- Como administrador necesito registrar a los especialistas que trabajan en la empresa.	<ul style="list-style-type: none"> - Registrar nombres, apellidos, numero de celular y correo electrónico. - Listar los especialistas.
003	Registro de cierre de caja	Como administrador necesito registrar el cierre de caja para tener un mejor control de las ventas diarias.	<ul style="list-style-type: none"> - Crear registros de hora y fecha de cierre de caja.
004	Registro de pacientes	- Como usuario necesito poder registrar a los pacientes para tener una lista de pacientes.	<ul style="list-style-type: none"> - Registrar nombres, apellidos, numero de celular y correo electrónico - Poder seleccionar los clientes para asignarles una historia clínica. - Debe permitir búsquedas de clientes por nombres y apellidos. - Listar a los clientes.

005	Modificar y eliminar pacientes	<ul style="list-style-type: none"> - Como usuario necesito modificar y eliminar los datos de mis pacientes para tener una lista con datos actuales. 	<ul style="list-style-type: none"> - Debe cargar los datos del cliente. - Debe permitir realizar la acción de modificar los datos del cliente. - Actualizar la lista de los clientes.
006	Gestionar permiso para usuarios	<ul style="list-style-type: none"> - Como administrador necesito que los permisos de los Usuarios estén restringidos para controlar sus accesos al sistema. 	<ul style="list-style-type: none"> - El sistema debe permitir el registro de usuarios encargados, en el módulo usuarios, sólo para registrar sus datos. - El usuario sólo puede acceder a los módulos: clientes, consultas y reservas.
007	Modificar y eliminar usuarios	<ul style="list-style-type: none"> - Como administrador necesito modificar y eliminar usuarios para tener un mejor control del personal que maneja el sistema. 	<ul style="list-style-type: none"> - Permitir editar y eliminar usuarios creados. - Actualizar la lista de usuarios con los nuevos datos ingresados.
008	Modificar cierre de caja	<ul style="list-style-type: none"> - Como administrador necesito modificar el cierre de caja para tener un mejor control de los ingresos y egresos diarias. 	<ul style="list-style-type: none"> - Cargar los registros de hora y fecha de cierre de caja. - Debe permitir realizar la acción de modificar los rangos de hora y fecha de cierre de caja. - Actualizar cambios.
009	Registro de cuenta de usuario	<ul style="list-style-type: none"> - Como usuario necesito una cuenta para poder modificar mi perfil. 	<ul style="list-style-type: none"> - Poder tener acceso al sistema para registrar. - Debe registrar los datos del usuario como nombres, apellidos, fecha de ingreso, DNI, tipo de usuario. - Debe tener un nombre de usuario y una contraseña.

			Sólo debe tener acceso a usuario, reservas y consultas.
010	Registro de historias clínicas	- Como usuario necesito tener una lista de todas las historias clínicas para que el especialista pueda atender al cliente.	<ul style="list-style-type: none"> - Visualizar una lista de las historias clínicas al día. - Realizar búsqueda de los clientes. - Imprimir historias de los clientes.
011	Registro de módulo de reservas	- Como usuario necesito registrar reservas para poder generar las consultas.	<ul style="list-style-type: none"> - Crear reserva. - Seleccionar cliente. - Seleccionar especialista. - Seleccionar día disponible. - Visualizar el precio a pagar. - Concretar reserva.
012	Anular reservas	- Como usuario necesito poder anular las reservas para cancelar en caso ya no sean útiles.	<ul style="list-style-type: none"> - Seleccionar la reserva y cambiar el estado ha anulado.
013	Generar historias clínicas	- Como usuario necesito generar historias clínicas a los clientes, para poder tener datos más precisos.	<ul style="list-style-type: none"> - Debe aparecer una búsqueda de clientes. - Debe permitir agregar historia clínica. - Debe permitir ingresar datos como, peso y talla.
014	Editar y eliminar historias clínicas	- Como usuario necesito editar y eliminar historias clínicas de los clientes para un mejor control de clientes.	<ul style="list-style-type: none"> - Se debe mostrar la lista de clientes. - Debe permitir la acción de editar o eliminar cliente y junto con él la historia clínica. - Actualizar lista de clientes y sus historias clínicas.

015	Generar cierre de caja	- Como vendedor necesito realizar el cierre de caja diario para poder general los reportes de ventas diarias.	- Crear anuncio de hora próxima al cierre de caja. Bloquear el acceso al sistema.
016	Generar reportes	- Como administrador necesito tener reportes para poder controlar mejor mi negocio.	- Listar las ventas del mes. - Imprimir mis consultas. Tamaño de la hoja debe ser A4 para la impresión de reportes.
017	Registro de módulo de consultas	- Como usuario necesito poder registrar consultas, para tener un mejor control del proceso de citas.	- Seleccionar la reserva hecha y cambiar el estado a generar consulta. - Registrar la consulta. - Generar Boleta de pago. - Imprimir boleta.
018	Modificar y eliminar especialistas	- Como administrador necesito modificar y eliminar especialistas para tener un mejor control del personal que trabaja.	- Debe cargar los datos del especialista. - Debe permitir realizar la acción de modificar los datos del especialista. - Actualizar la lista de especialistas.
019	Visualizar ventas diarias y mensuales	- Como administrador necesito poder visualizar las ventas diarias y mensuales.	- Visualizar las ventas diarias. - Visualizar las ventas por mes.

Fuente: Desarrollo Propio.

Tabla 28: Priorización de HU por prioridad – Caso 02

Código	Título	Historia de Usuario	Criterio de Aceptación	Prioridad
001	Registro de cuenta administrador	<ul style="list-style-type: none"> - Como administrador necesito poder registrarme en el sistema para poder controlar las actividades en la clínica. 	<ul style="list-style-type: none"> - Registrar datos como nombre, apellidos, fecha de ingreso, DNI, tipo de usuario. - Debe tener un usuario de administrador y una contraseña. - Tener todos los permisos habilitados en el sistema. - Debe permitir visualizar y modificar permisos en todos los módulos. 	ALTA
002	Registro de especialistas	<ul style="list-style-type: none"> - Como administrador necesito registrar a los especialistas que trabajan en la empresa. 	<ul style="list-style-type: none"> - Registrar nombres, apellidos, numero de celular y correo electrónico. - Listar los especialistas. 	ALTA
003	Registro de cierre de caja	Como administrador necesito registrar el cierre de caja para tener un mejor control de las ventas diarias.	<ul style="list-style-type: none"> - Crear registros de hora y fecha de apertura y cierre de caja. 	ALTA
004	Registro de pacientes	<ul style="list-style-type: none"> - Como usuario necesito poder registrar a los pacientes para tener una lista de pacientes. 	<ul style="list-style-type: none"> - Registrar nombres, apellidos, numero de celular y correo electrónico - Poder seleccionar los clientes para asignarles una historia clínica. - Debe permitir búsquedas de clientes por nombres y apellidos. - Listar a los clientes. 	MEDIA

005	Modificar y eliminar pacientes	<ul style="list-style-type: none"> - Como usuario necesito modificar y eliminar los datos de mis pacientes para tener una lista con datos actuales. 	<ul style="list-style-type: none"> - Debe cargar los datos del cliente. - Debe permitir realizar la acción de modificar los datos del cliente. - Actualizar la lista de los clientes. 	MEDIA
006	Gestionar permiso para usuarios	<ul style="list-style-type: none"> - Como administrador necesito que los permisos de los Usuarios estén restringidos para controlar sus accesos al sistema. 	<ul style="list-style-type: none"> - El sistema debe permitir el registro de usuarios encargados, en el módulo usuarios, sólo para registrar sus datos. - El usuario sólo puede acceder a los módulos: clientes, consultas y reservas. 	ALTA
007	Modificar y eliminar usuarios	<ul style="list-style-type: none"> - Como administrador necesito modificar y eliminar usuarios para tener un mejor control del personal que maneja el sistema. 	<ul style="list-style-type: none"> - Permitir editar y eliminar usuarios creados. <p>Actualizar la lista de usuarios con los nuevos datos ingresados.</p>	ALTA
008	Modificar cierre de caja	<ul style="list-style-type: none"> - Como administrador necesito modificar el cierre de caja para tener un mejor control de los ingresos y egresos diarias. 	<ul style="list-style-type: none"> - Cargar los registros de hora y fecha de cierre de caja. - Debe permitir realizar la acción de modificar los rangos de hora y fecha de cierre de caja. - Actualizar cambios. 	ALTA
009	Registro de cuenta de usuario	<ul style="list-style-type: none"> - Como usuario necesito una cuenta para poder modificar mi perfil. 	<ul style="list-style-type: none"> - Poder tener acceso al sistema para registrar. - Debe registrar los datos del usuario como nombres, apellidos, fecha de ingreso, DNI, tipo de usuario. - Debe tener un nombre de usuario y una contraseña. - Sólo debe tener acceso a usuario, reservas y consultas. 	ALTA

010	Registro de historias clínicas	<ul style="list-style-type: none"> - Como usuario necesito tener una lista de todas las historias clínicas para que el especialista pueda atender al cliente. 	<ul style="list-style-type: none"> - Visualizar una lista de las historias clínicas al día. - Realizar búsqueda de los clientes. - Imprimir historias de los clientes. 	MEDIA
011	Registro de módulo de reservas	<ul style="list-style-type: none"> - Como usuario necesito registrar reservas para poder generar las consultas. 	<ul style="list-style-type: none"> - Crear reserva. - Seleccionar cliente. - Seleccionar especialista. - Seleccionar día disponible. - Visualizar el precio a pagar. - Concretar reserva. 	MEDIA
012	Anular reservas	<ul style="list-style-type: none"> - Como usuario necesito poder anular las reservas para cancelar en caso ya no sean útiles. 	<ul style="list-style-type: none"> - Seleccionar la reserva y cambiar el estado ha anulado. 	MEDIA
013	Generar historias clínicas	<ul style="list-style-type: none"> - Como usuario necesito generar historias clínicas a los clientes, para poder tener datos más precisos. 	<ul style="list-style-type: none"> - Debe aparecer una búsqueda de clientes. - Debe permitir agregar historia clínica. - Debe permitir ingresar datos como, peso y talla. 	MEDIA
014	Editar y eliminar historias clínicas	<ul style="list-style-type: none"> - Como usuario necesito editar y eliminar historias clínicas de los clientes para un mejor control de clientes. 	<ul style="list-style-type: none"> - Se debe mostrar la lista de clientes. - Debe permitir la acción de editar o eliminar cliente y junto con él la historia clínica. - Actualizar lista de clientes y sus historias clínicas. 	MEDIA

015	Generar cierre de caja	- Como vendedor necesito realizar el cierre de caja diario para poder general los reportes de ventas diarias.	- Crear anuncio de hora próxima al cierre de caja. Bloquear el acceso al sistema.	BAJA
016	Generar reportes	- Como administrador necesito tener reportes para poder controlar mejor mi negocio.	- Listar las ventas del mes. - Imprimir mis consultas. Tamaño de la hoja debe ser A4 para la impresión de reportes.	BAJA
017	Registro de módulo de consultas	- Como usuario necesito poder registrar consultas, para tener un mejor control del proceso de citas.	- Seleccionar la reserva hecha y cambiar el estado a generar consulta. - Registrar la consulta. - Generar Boleta de pago. - Imprimir boleta.	BAJA
018	Modificar y eliminar especialistas	- Como administrador necesito modificar y eliminar especialistas para tener un mejor control del personal que trabaja.	- Debe cargar los datos del especialista. - Debe permitir realizar la acción de modificar o eliminar los datos del especialista. - Actualizar la lista de especialistas.	ALTA
019	Visualizar ventas diarias y mensuales	- Como administrador necesito poder visualizar las ventas diarias y mensuales.	- Visualizar las ventas diarias. - Visualizar las ventas por mes.	BAJA

Nota. Fuente: Elaboración Propia.

Tabla 29: Orden de HU por prioridad - Caso 02

Código	Título	Historia de Usuario	Prioridad
001	Registro de cuenta administrador	Como administrador necesito poder registrarme en el sistema para poder controlar las actividades en la clínica.	ALTA
006	Gestionar permiso para usuarios	Como administrador necesito que los permisos de los Usuarios estén restringidos para controlar sus accesos al sistema.	ALTA
009	Registro de cuenta de usuarios	Como usuario necesito una cuenta para poder modificar mi perfil.	ALTA
007	Modificar y eliminar usuarios	Como administrador necesito modificar y eliminar usuarios para tener un mejor control del personal que maneja el sistema.	ALTA
002	Registro de especialistas	Como administrador necesito registrar a los especialistas que trabajan en la empresa.	ALTA
018	Modificar y eliminar especialistas	Como administrador necesito modificar y eliminar mis productos para tener una cartera de productos actualizada.	ALTA
003	Registro de cierre de caja	Como administrador necesito registrar el cierre de caja para tener un mejor control de las ventas diarias.	ALTA
008	Modificar y eliminar cierre de caja	Como administrador necesito modificar el cierre de caja para tener un mejor control de los ingresos y egresos diarias.	ALTA
004	Registro de módulo pacientes	Como usuario necesito poder registrar a los pacientes para tener una lista de pacientes.	MEDIA

005	Modificar y eliminar pacientes	Como usuario necesito modificar y eliminar los datos de mis pacientes para tener una lista con datos actuales.	MEDIA
011	Registro de módulo de reservas	Como usuario necesito registrar reservas para poder generar las consultas.	MEDIA
012	Anular reservas	Como usuario necesito poder anular las reservas para cancelar en caso ya no sean útiles.	MEDIA
010	Registro de historias clínicas	Como usuario necesito tener una lista de todas las historias clínicas para que el especialista p	MEDIA
013	Generar historias clínicas	Como usuario n los clientes, para poder tener datos más	MEDIA
014	Editar y eliminar historias clínicas	Como usuario necesito editar y eliminar historias clínicas de los clientes para un mejor control de clientes.	MEDIA
017	Registro de módulo de consultas	Como usuario necesito poder registrar consultas, para tener un mejor control del proceso de citas.	BAJA
019	Visualizar ventas diarias y mensuales	Como administrador necesito poder visualizar las ventas diarias y mensuales.	BAJA
016	Generar reportes	Como administrador necesito tener reportes para poder controlar mejor mi negocio.	BAJA
015	Generar cierre de caja	Como vendedor necesito realizar el cierre de caja diario para poder general los reportes de ventas diarias.	BAJA

Fuente: Desarrollo Propio.

Separación por Sprints

Teniendo en cuenta las historias de usuario presentes en el backlog y siendo priorizadas, tomamos las de mayor prioridad para ir creando los Sprints. Luego de tomar en cuenta la cantidad de Historias de Usuario a tomarse en los Sprints, mostramos la siguiente lista:

SPRINT 01

Se empezó con el diseño de la base de datos para poder iniciar con el desarrollo, para lo cual en este diseño se contará con el apoyo de los clientes, quienes informarán sobre los datos que son importantes para la empresa, trabajaremos con generar una cuenta de administrador, usuarios y el registro de especialistas.

Tabla 30: Primer Sprint - Caso 02

Código	Título	Prioridad
1	Registro de cuenta administrador	ALTA
6	Gestionar permiso para usuarios	ALTA
9	Registro de cuenta de usuarios	ALTA
7	Modificar y eliminar usuarios	ALTA
2	Registro de especialistas	ALTA
18	Modificar y eliminar especialistas	ALTA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 2 semanas.

SPRINT 02

Esta iteración trabajamos las siguientes historias de usuario, relacionadas con los pacientes, reservas y cierre de caja.

Tabla 31: Segundo Sprint - Caso 02

Código	Título	Prioridad
3	Registro de cierre de caja	ALTA
8	Modificar y eliminar cierre de caja	ALTA
4	Registro de módulo pacientes	MEDIA
5	Modificar y eliminar pacientes	MEDIA
11	Registro de módulo de reservas	MEDIA
12	Anular reservas	MEDIA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 2 semanas.

SPRINT 03

Esta iteración trabajamos las siguientes historias de usuario, relacionadas con las historias clínicas y el módulo de consulta.

Tabla 32: Tercer Sprint - Caso 02

Código	Título	Prioridad
10	Registro de historias clínicas	MEDIA
13	Generar historias clínicas	MEDIA
14	Editar y eliminar historias clínicas	MEDIA
17	Registro de módulo de consultas	BAJA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 2 semanas.

SPRINT 04

Esta iteración trabajamos las siguientes historias de usuario, generación de reportes.

Tabla 33: Cuarto Sprint - Caso 02

Código	Título	Prioridad
19	Visualizar ventas diarias y mensuales	BAJA
16	Generar reportes	BAJA
15	Generar cierre de caja	BAJA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 2 semanas.

Desglose en tareas a cargo del equipo de desarrollo: El equipo de desarrollo realizó el desglose de las UH en la siguiente lista de tareas:

Tabla 34: Desglose de HU en tareas - Caso 02

N° SPRINT	CÓDIGO	NOMBRE HU	TAREAS
1	001	Registro de cuenta administrador	Creación e implementación de la base de datos
			Creación de cuenta para registro de administrador.
			Restringir permisos para los usuarios.
	006	Gestionar permiso para usuarios	Creación y restricción de permisos para los usuarios administrador y usuario – recepcionista.
	009	Registro de cuenta de usuarios	Creación del módulo recepcionista.
			Registro e inicio de sesión para recepcionista.
			Creación de acceso a los módulos: clientes, consultas y reservas.
	007	Modificar y eliminar usuarios	Modificar y eliminar datos del usuario - recepcionistas. Actualización de registro de recepcionistas.
2	002	Registro de especialistas	Creación del módulo especialistas
			Listar especialistas
	018	Modificar y eliminar especialistas	Modificar y eliminar datos de los especialistas.
			Actualización de registro de especialistas.
	003	Registro de cierre de caja	Diseño e implementación del mantenimiento caja.
			Registro de fechas y horas de inicio y cierre de caja.
	008	Modificar y eliminar cierre de caja	Modificar y eliminar datos como las fechas y horas de inicio y cierre de caja.

3	004	Registro de módulo pacientes	Creación del módulo pacientes.
			Creación de búsquedas avanzadas.
			Creación de listas para pacientes.
			Crear opción para visualizar las reservas de los pacientes, para ofrecer fechas disponibles.
			Creación de opción para aplicar descuentos para los pacientes con bajos recursos.
	005	Modificar y eliminar pacientes	Modificar y eliminar datos de los pacientes
			Actualización de listas de todos los pacientes
	011	Registro de módulo de reservas	Creación del módulo reservas.
			Mostrar el reporte de reservas por días.
			Listar las reservas.
4	012	Anular reservas	Mostrar lista de especialistas disponibles y descuentos respectivos a los pacientes.
			Crear estado de anular reservas y cambiar estados.
			Actualización de listas de las reservas.
	010	Registro de historias clínicas	Creación del módulo de historias clínicas .
			Relacionar tablas de pacientes, especialistas e historias clínicas.
	013	Generar historias clínicas	Crear historias clínicas
			Realizar búsqueda de los pacientes.
			Generar Impresión de historias de los pacientes.
	014	Editar y eliminar historias clínicas	Modificar y eliminar historias clínicas
			Actualización de listas de todos los pacientes.
	017	Registro de módulo de consultas	Crear y diseñar el cambio de estado de consultas en espera a aceptado.
			Creación del módulo consultas.
			Mostrar el reporte de consultas programadas.
	019	Visualizar ventas diarias y mensuales	Crear backup de la información.
	016	Generar reportes	General listas de ventas por mes.

		Generar e imprimir pedidos y ventas.
015	Generar cierre de caja	Crear anuncio de cierre de caja 1 hora antes.
		Restringir y bloquear todos los permisos a la hora señalada del cierre de caja.

Fuente: Desarrollo Propio.

Tabla 35: Porcentaje de avance de tareas - Caso 02

N° SPRINT	CÓDIGO	NOMBRE HU	TAREAS	ESTADO	% DE AVANCE
1	1	Registro de cuenta administrador	Creación e implementación de la base de datos	T	100%
			Creación de cuenta para registro de administrador.	T	
			Restringir permisos para los usuarios.	T	
	6	Gestionar permiso para usuarios	Creación y restricción de permisos para los usuarios administrador y usuario – recepcionista.	T	100%
	9	Registro de cuenta de usuarios	Creación del módulo recepcionista.	T	100%
			Registro e inicio de sesión para recepcionista.	T	
			Creación de acceso a los módulos: clientes, consultas y reservas.	T	
	7	Modificar y eliminar usuarios	Modificar y eliminar datos del usuario - recepcionistas.	T	100%
			Actualización de registro de recepcionistas.	T	
	2	Registro de especialistas	Creación del módulo especialistas	T	100%
			Listar especialistas	T	

2	18	Modificar y eliminar especialistas	Modificar y eliminar datos de los especialistas.	T	100%
			Actualización de registro de especialistas.	T	
	3	Registro de cierre de caja	Diseño e implementación del mantenimiento caja.	T	100%
			Registro de fechas y horas de inicio y cierre de caja.	T	
	8	Modificar y eliminar cierre de caja	Modificar y eliminar datos como las fechas y horas de inicio y cierre de caja.	T	100%
	4	Registro de módulo pacientes	Creación del módulo pacientes.	T	100%
			Creación de búsquedas avanzadas.	T	
			Creación de listas para pacientes.	T	
			Crear opción para visualizar las reservas de los pacientes, para ofrecer fechas disponibles.	T	
			Creación de opción para aplicar descuentos para los pacientes con bajos recursos.	T	
	5	Modificar y eliminar pacientes	Modificar y eliminar datos de los pacientes	T	100%
			Actualización de listas de todos los pacientes	T	
	11	Registro de módulo de reservas	Creación del módulo reservas.	T	100%
			Mostrar el reporte de reservas por días.	T	

3			Listar las reservas.	T	
			Mostrar lista de especialistas disponibles y descuentos respectivos a los pacientes.	T	
	12	Anular reservas	Crear estado de anular reservas y cambiar estados.	T	100%
			Actualización de listas de las reservas.	T	
	10	Registro de historias clínicas	Creación del módulo de historias clínicas.	T	100%
			Relacionar tablas de pacientes, especialistas e historias clínicas.	T	
	13	Generar historias clínicas	Crear historias clínicas	T	100%
			Realizar búsqueda de los pacientes.	T	
			Generar Impresión de historias de los pacientes.	T	
	14	Editar y eliminar historias clínicas	Modificar y eliminar historias clínicas	T	100%
			Actualización de listas de	T	
			todos los pacientes.	T	
4	17	Registro de módulo de consultas	Crear y diseñar el cambio de estado de consultas en espera a aceptado.	T	100%
			Creación del módulo de consultas.	T	
			Mostrar el reporte de consultas programadas.	T	
	19	Visualizar ventas diarias y mensuales	Crear backup de la información.	T	100%

16	Generar reportes	General listas de ventas por mes.	T	100%
		Generar e imprimir pedidos y ventas.	T	
15	Generar cierre de caja	Crear anuncio de cierre de caja 1 hora antes.	T	100%
		Restringir y bloquear todos los permisos a la hora señalada del cierre de caja.	T	

Fuente: Desarrollo Propio.

Checklist: Es una lista detallada de los criterios de aceptación fijados por el PO en un inicio junto con las historias de usuario, el PO deberá dar la conformidad y aprobación si algo no resulta como lo solicitó debe agregar observaciones detalladas del cambio, para que el equipo de desarrollo sepa si eso fue lo que se solicitó o se deben hacer cambios. Para este caso los criterios de aceptación son los siguientes:

Los criterios de aceptación son los siguientes:

Tabla 36: Verificación de criterios de aceptación - Caso 02

Sprint	Código de HU	Nombre de HU	Criterio de Aceptación	Cumplimiento		Observaciones
				Si	No	
1	001	Registro de cuenta administrador	<ul style="list-style-type: none"> - Registrar datos como nombre, - Debe tener un usuario de administrador y una contraseña. - Tener todos los permisos habilitados en el sistema. - Debe permitir visualizar y modificar permisos en todos los módulos. 	X		
	006	Gestionar permiso para usuarios	<ul style="list-style-type: none"> - Permitir editar y eliminar usuarios creados. - Actualizar la lista de usuarios con los nuevos datos ingresados. 	X		
	009	Registro de cuenta de usuarios	<ul style="list-style-type: none"> - Poder tener acceso al sistema para registrar. - Debe registrar los datos del usuario como nombres, apellidos, etc. - Debe tener un nombre de usuario y una contraseña. - Sólo debe tener acceso a usuario, reservas y consultas. 	X		
	007	Modificar y eliminar usuarios	<ul style="list-style-type: none"> - Crear registros de hora y fecha de cierre de caja. 	X		

002	Registro de especialistas	<ul style="list-style-type: none"> - Registrar nombres, apellidos, numero de celular y correo electrónico. - Listar los especialistas. 	x	
	Modificar y eliminar especialistas	<ul style="list-style-type: none"> - Debe cargar los datos del especialista. - Debe permitir realizar la acción de modificar y eliminar los datos del especialista. - Actualizar la lista de especialistas. 	x	
003	Registro de cierre de caja	<ul style="list-style-type: none"> - Registrar nombres, apellidos, numero de celular y correo electrónico - Poder seleccionar los clientes para asignarles una historia clínica. - Debe permitir búsquedas de clientes por nombres y apellidos. - Listar a los clientes. 	x	La lista se debe mostrar por orden alfabético
	Modificar y eliminar cierre de caja	<ul style="list-style-type: none"> - Cargar los registros de hora y fecha de cierre de caja. - Debe permitir realizar la acción de modificar los rangos de hora y fecha de cierre de caja. - Actualizar cambios. 	x	Debe permitir hacer otros tipos de descuentos.

3	004	Registro de módulo pacientes	<ul style="list-style-type: none"> - Debe cargar los datos del cliente. - Debe permitir realizar la acción de modificar los datos del cliente. - Actualizar la lista de los clientes. 	x	No se cumplió no se puede realizar el registro, se cargan datos incompletos.
	005	Modificar y eliminar pacientes	<ul style="list-style-type: none"> - El sistema debe permitir el registro de usuarios encargados, en el módulo usuarios, sólo para registrar sus datos. - El usuario sólo puede acceder a los módulos: clientes, consultas y reservas. 	x	El registro sólo debe ser para fechas posteriores al mes de noviembre.
	011	Registro de módulo de reservas	<ul style="list-style-type: none"> - Crear reserva. - Seleccionar cliente. - Seleccionar especialista. - Seleccionar día disponible. - Visualizar el precio a pagar. - Concretar reserva. 	x	
	012	Anular reservas	<ul style="list-style-type: none"> - Seleccionar la reserva y cambiar el estado ha anulado. 	x	
	010	Registro de historias clínicas	<ul style="list-style-type: none"> - Visualizar una lista de las historias clínicas al día. - Realizar búsqueda de los clientes. - Imprimir historias de los clientes. 	x	

4	013	Generar historias clínicas	<ul style="list-style-type: none"> - Debe aparecer una búsqueda de clientes. - Debe permitir agregar historia clínica. - Debe permitir ingresar datos como, peso y talla. 	x	Cuando esté en estado Anulado debe ponerse en rojo.
	014	Editar y eliminar historias clínicas	<ul style="list-style-type: none"> - Se debe mostrar la lista de clientes. - Debe permitir la acción de editar o eliminar cliente y junto con él la historia clínica. - Actualizar lista de clientes y sus historias clínicas. 	x	
	017	Registro de módulo de consultas	<ul style="list-style-type: none"> - Seleccionar la reserva hecha y cambiar el estado a generar consulta. - Registrar la consulta. - Generar Boleta de pago. - Imprimir boleta. 	x	Mostrar el anuncio en color azul con letras más grandes.
	019	Visualizar ventas diarias y mensuales	<ul style="list-style-type: none"> - Visualizar las ventas diarias. - Visualizar las ventas por mes. 		
4	016	Generar reportes	<ul style="list-style-type: none"> - Listar las ventas del mes. - Imprimir mis consultas. - Tamaño de la hoja debe ser A4 para la impresión de reportes. 	x	
	015	Generar cierre de caja	<ul style="list-style-type: none"> - Crear anuncio de hora próxima al cierre de caja. - Bloquear el acceso al sistema. 	x	

Fuente: Desarrollo Propio.

En este caso vemos que los criterios de aceptación de una Historias de Usuario no se llegaron a cumplir de acuerdo a lo que pidió el PO y los cambios que desea no son simples, así que esta historia de usuario irá en el siguiente Sprint, Es decir la HU con código 004 está ubicada en el segundo sprint, al no poder ser modificada inmediatamente pasa al tercer Sprint como una nueva Historia, en este caso no borraremos la HU 004 para que quede como registro para vulnerabilidades posteriores.

SPRINT 02

Esta iteración trabajamos las siguientes historias de usuario, relacionadas con los pacientes, reservas y cierre de caja.

Tabla 37: Sprint a evaluar - Caso 02

Código	Título	Prioridad
003	Registro de cierre de caja	ALTA
008	Modificar y eliminar cierre de caja	ALTA
004	Registro de módulo pacientes	MEDIA
005	Modificar y eliminar pacientes	MEDIA
011	Registro de módulo de reservas	MEDIA
012	Anular reservas	MEDIA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 2 semanas.

SPRINT 03

Esta iteración trabajamos las siguientes historias de usuario, relacionadas con las historias clínicas y el módulo de consulta.

Tabla 38: Sprint 03 modificado - Caso 02

Código	Título	Prioridad
020	Registro de módulo pacientes	ALTA
010	Registro de historias clínicas	MEDIA
013	Generar historias clínicas	MEDIA
014	Editar y eliminar historias clínicas	MEDIA
017	Registro de módulo de consultas	BAJA

Fuente: Desarrollo Propio.

Tiempo de Estimación: 3 semanas.

Como vemos la Historia de Usuario 004 no se llegó a cumplir ya que el Product Owner escribe que “*no se cumplió*”, entonces esta historia tiene que pasar al tercer Sprint con prioridad alta para ser desarrollada antes que las anteriores.

Para esto lo ingresamos como una nueva historia de usuario al backlog con un código distinto para poder diferenciarla del resto, en este caso registraremos la historia con el código 020 y le agregamos los criterios de aceptación y asignarle una prioridad alta para que ingrese primero en la lista del siguiente Sprint.

Una vez agregada pasará por los procesos antes descritos en el caso 01 y así sucesivamente hasta ser aprobada por el Product Owner.

Cabe detallar además que el Product Owner debe detallar con precisión los criterios que no se cumplieron para un mejor entendimiento con el equipo de desarrollo para facilitar el desenvolvimiento de las siguientes actividades en los Sprints

Los criterios de aceptación son los siguientes:

Tabla 39: Verificación de criterios de aceptación - Caso 02

Sprint	Código de HU	Nombre de HU	Criterio de Aceptación	Cumplimiento		Observaciones
				Si	No	
1	001	Registro de cuenta administrador	<ul style="list-style-type: none"> - Registrar datos como nombre, - Debe tener un usuario de administrador y una contraseña. - Tener todos los permisos habilitados en el sistema. - Debe permitir visualizar y modificar permisos en todos los módulos. 	x		
	006	Gestionar permiso para usuarios	<ul style="list-style-type: none"> - Permitir editar y eliminar usuarios creados. - Actualizar la lista de usuarios con los nuevos datos ingresados. 	x		
	009	Registro de cuenta de usuarios	<ul style="list-style-type: none"> - Poder tener acceso al sistema para registrar. - Debe registrar los datos del usuario como nombres, apellidos, etc. - Debe tener un nombre de usuario y una contraseña. - Sólo debe tener acceso a usuario, reservas y consultas. 	x		
	007	Modificar y eliminar usuarios	<ul style="list-style-type: none"> - Crear registros de hora y fecha de cierre de caja. 	x		
	002	Registro de especialistas	<ul style="list-style-type: none"> - Registrar nombres, apellidos, numero de celular y correo electrónico. - Listar los especialistas. 	x		

018	Modificar y eliminar especialistas	- Debe cargar los datos del especialista.	X
		- Debe permitir realizar la acción de modificar y eliminar los datos del especialista.	
		- Actualizar la lista de especialistas.	
003	Registro de cierre de caja	- Registrar nombres, apellidos, numero de celular y correo electrónico	X
		- Poder seleccionar los clientes para asignarles una historia clínica.	
		- Debe permitir búsquedas de clientes por nombres y apellidos. Listar a los clientes.	
2 008	Modificar y eliminar cierre de caja	- Cargar los registros de hora y fecha de cierre de caja.	X
		- Debe permitir realizar la acción de modificar los rangos de hora y fecha de cierre de caja.	
		- Actualizar cambios.	
004	Registro de módulo pacientes	- Debe cargar los datos del cliente.	X
		- Debe permitir realizar la acción de modificar los datos del cliente.	
		- Actualizar la lista de los clientes.	

005	Modificar y eliminar pacientes	- El sistema debe permitir el registro de usuarios encargados, en el módulo usuarios, sólo para registrar sus datos.	X	-
		- El usuario sólo puede acceder a los módulos: clientes, consultas y reservas.		
		- Crear reserva.		
011	Registro de módulo de reservas	- Seleccionar cliente.		
		- Seleccionar especialista.		
		- Seleccionar día disponible.	X	
		- Visualizar el precio a pagar.		
		- Concretar reserva.		
012	Anular reservas	Seleccionar la reserva y cambiar el estado ha anulado.	X	
020	Registro de módulo pacientes	- Debe cargar los datos del cliente incluyendo DNI y número de celular.		
		- Debe permitir realizar la acción de modificar y eliminar los datos del cliente.	X	
		- Actualizar la lista de los clientes cada día.		
3	010	- Visualizar una lista de las historias clínicas al día.		
		- Realizar búsqueda de los clientes.	X	
		- Imprimir historias de los clientes.		

4	013	Generar historias clínicas	<ul style="list-style-type: none"> - Debe aparecer una búsqueda de clientes. - Debe permitir agregar historia clínica. - Debe permitir ingresar datos como, peso y talla. 	x	
	014	Editar y eliminar historias clínicas	<ul style="list-style-type: none"> - Se debe mostrar la lista de clientes. - Debe permitir la acción de editar o eliminar cliente y junto con él la historia clínica. - Actualizar lista de clientes y sus historias clínicas. 	x	-
	017	Registro de módulo de consultas	<ul style="list-style-type: none"> - Seleccionar la reserva hecha y cambiar el estado a generar consulta. - Registrar la consulta. - Generar Boleta de pago. - Imprimir boleta. 	x	
	019	Visualizar ventas diarias y mensuales	<ul style="list-style-type: none"> - Visualizar las ventas diarias. - Visualizar las ventas por mes. - Imprimir boleta. 	x	
	016	Generar reportes	<ul style="list-style-type: none"> - Listar las ventas del mes. - Imprimir mis consultas. - Tamaño de la hoja debe ser A4 para la impresión de reportes. 	x	-
	015	Generar cierre de caja	<ul style="list-style-type: none"> - Crear anuncio de hora próxima al cierre de caja. - Bloquear el acceso al sistema. 	x	-

Fuente: Desarrollo Propio.

3.5. IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL

Para realizar la implementación de la aplicación móvil en las empresas se realizaron varias reuniones con distintos trabajadores los cuales desarrollaron los roles definidos en Scrum. Con los datos anteriores tomando los dos casos como base desarrollamos los siguientes proyectos en la aplicación.

3.5.1. Desarrollo de proyectos

Tomamos en base los proyectos anteriores y los ingresamos para ser trabajados en las sus respectivas interfaces.

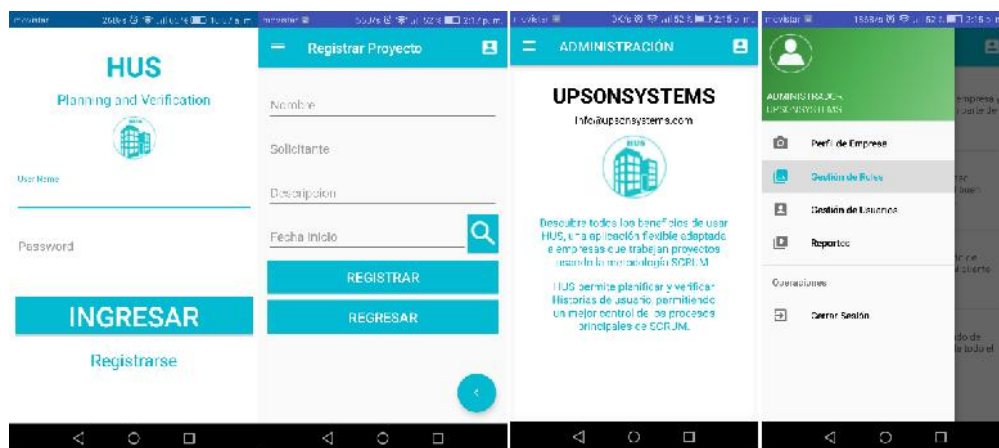
Código: 001

Nombre: Inicio de sesión - Registro

Descripción: El administrador al ingresar por primera vez tendrá que registrar los datos de la empresa, y crear una cuenta de administrador.

Requisitos: Tener conexión a internet.

Interfaces:



Entradas: Datos de la empresa.

Proceso: La persona que desarrollará el papel de administrador deberá ir a la Opción “Registrarse”, y registrará los datos de la empresa en general y una cuenta de usuario y contraseña para los usuarios.

Resultados esperados: Mensaje de confirmación de cuenta y acceso a las opciones de Perfil de Empresa, Gestión de roles y Gestión de Usuarios.

Código: 002

Nombre: Gestión de Cuenta de Administrador – Perfil de empresa y gestión de roles

Descripción: El administrador podrá tener acceso a las siguientes interfaces: Perfil de Empresa, Gestión de roles.

Requisitos: Haber registrado una cuenta de usuario como administrado, y haberse registrado como tal.

Interfaces:

Entradas: Usuario y contraseña de administrador

Proceso: Se habilitará 2 pantallas: El perfil de empresa y gestión de roles, las cuales sólo son informativas y sólo podrán ser visualizadas sin poder realizar cambios.

Resultados esperados: Visualización del perfil de empresa y gestión de roles.

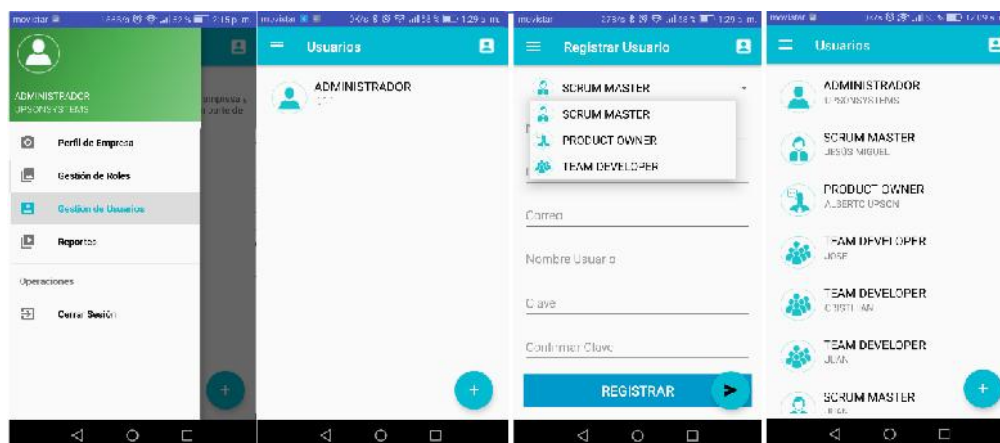
Código: 003

Nombre: Gestión de Cuenta de Administrador – Gestión de usuarios.

Descripción: El administrador podrá registrar el nuevo ingreso de usuarios asignándoles un rol específico y sus respectivas cuentas de usuarios.

Requisitos: Haber registrado una cuenta de usuario como administrado, y haberse registrado como tal.

Interfaces:



Entradas: Datos de los encargados que van a trabajar en los proyectos, Puede crear solo un Product Owner, 01 Scrum Master y un máximo de 5 personas en el equipo de desarrollo.

Proceso: El administrador agregará los datos de los usuarios uno a uno tomando en cuenta, el Rol, sus nombres, apellidos, dirección, correo, nombre de usuario y contraseña.

Resultados esperados: Lista de usuarios que participarán en los proyectos con sus respectivos nombres de usuario y contraseña.

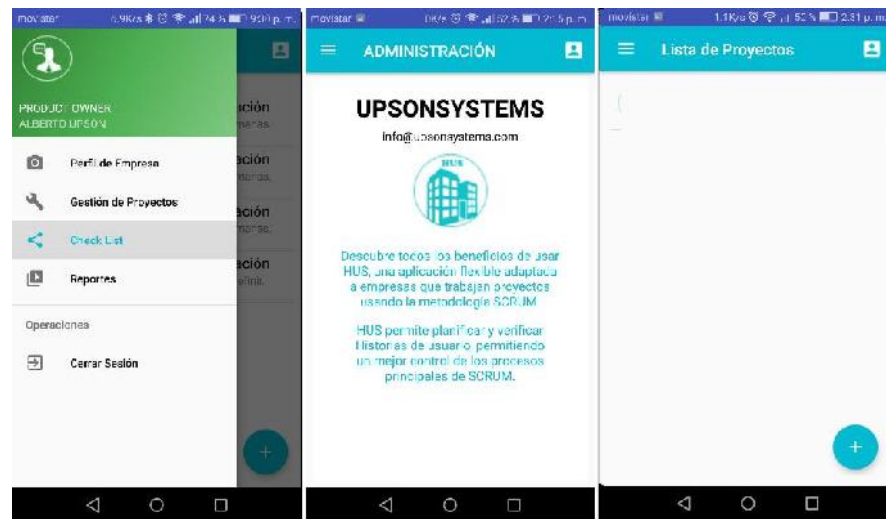
Código: 004

Nombre: Gestión de cuenta de Product Owner

Descripción: En este proceso el Product owner podrá ingresar su clave y contraseña para tener acceso a Perfil de empresa, crear proyectos y asignar el personal a tales proyectos.

Requisitos: Tener una cuenta y un usuario como PO generada por el Administrador.

Interfaces:



Entradas: Nombre de usuario y contraseña.

Proceso: El Product Owner determinado como tal, podrá tener acceso a Perfil de empresa (modo Visualización), Gestión de proyectos y checklist.

Resultados esperados: Tener visualización del perfil de empresa, poder gestionar proyecto y gestionar checklist.

3.5.2. Planificación de proyectos

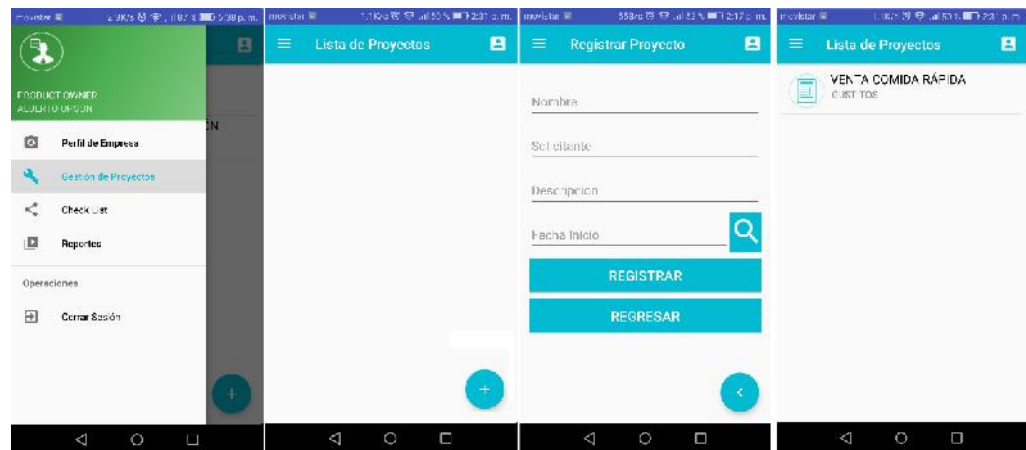
Código: 005

Nombre: Gestión de cuenta de Product Owner - Gestión de proyecto.

Descripción: En este proceso el Product owner podrá crear proyectos, para ser desarrollados por todo el personal.

Requisitos: Tener una cuenta y un usuario como PO generada por el Administrador.

Interfaces:



Entradas: Debemos tener los datos del proyecto como nombre, Solicitante, descripción y la fecha de inicio.

Proceso: El Product Owner determinado como tal deberá iniciar sesión, una vez validados sus datos podrá tener acceso a *Gestión de Proyectos*, donde deberá registrar los datos del proyecto, seleccionar una fecha de inicio y finalmente crear el proyecto.

Resultados esperados: El Product owner crea proyectos.

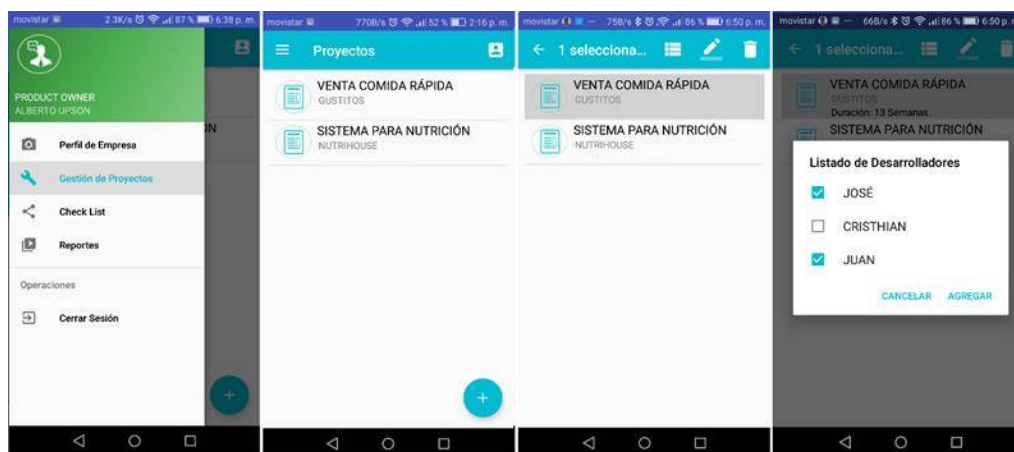
Código: 006

Nombre: Gestión de cuenta de Product Owner – Asignar personal

Descripción: En este proceso el Product Owner podrá asignar al personal del equipo de desarrollo a los proyectos creados.

Requisitos: Tener una cuenta y un usuario como PO generada por el Administrador y tener una lista del personal que trabajará en el equipo de desarrollo.

Interfaces:



Entradas: Tener proyectos creados y seleccionados.

Proceso: En este proceso el PO deberá seleccionar un proyecto y se desplegará la lista de desarrolladores, seleccionará al personal que trabaje en ese proyecto. Cabe mencionar que al seleccionar el proyecto también se habilitará las opciones de “editar” y “eliminar”.

Resultados esperados: El PO podrá asignar personal del Equipo de desarrollo a un proyecto seleccionado, también podrá editar y eliminar proyectos.

3.5.3. Generación de Historias de Usuario

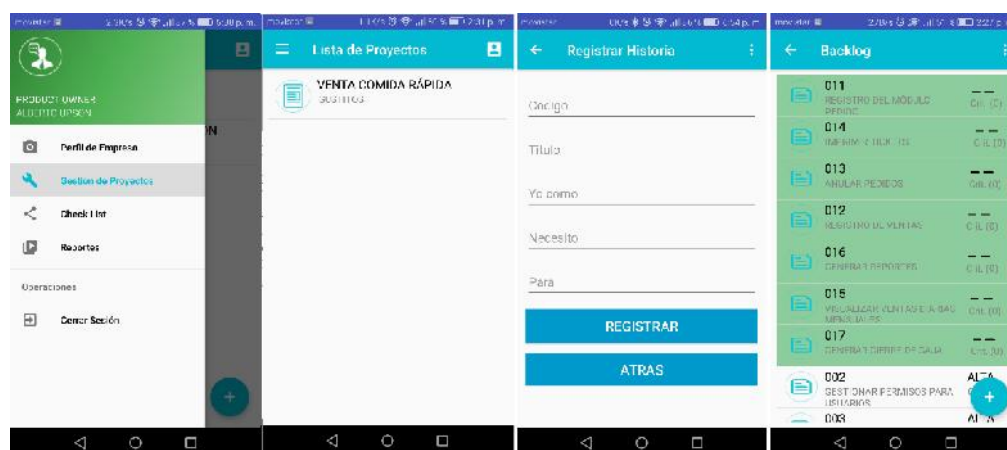
Código: 007

Nombre: Gestión de cuenta de Product Owner – Creación de HU en el backlog

Descripción: En este proceso el Product Owner podrá registrar, editar y eliminar historias de usuario en el backlog.

Requisitos: Tener una cuenta y un usuario como PO generada por el Administrador y tener una lista de requerimientos para ser ingresados como historias de usuario.

Interfaces:



Entradas: Para crear las Historias de Usuario debemos seguir la siguiente secuencia, un Código, título, seguido de las premisas *Yo Como, necesito, para*. Según lo establecido anteriormente.

Proceso: En este proceso el PO deberá seleccionar el ícono de agregar, para poder agregar historias de usuario, seguirá ingresando la secuencia antes señalada e irá agregando las historias de usuario una a una.

Resultados esperados: El PO podrá agregar una lista de historias de usuario al backlog.

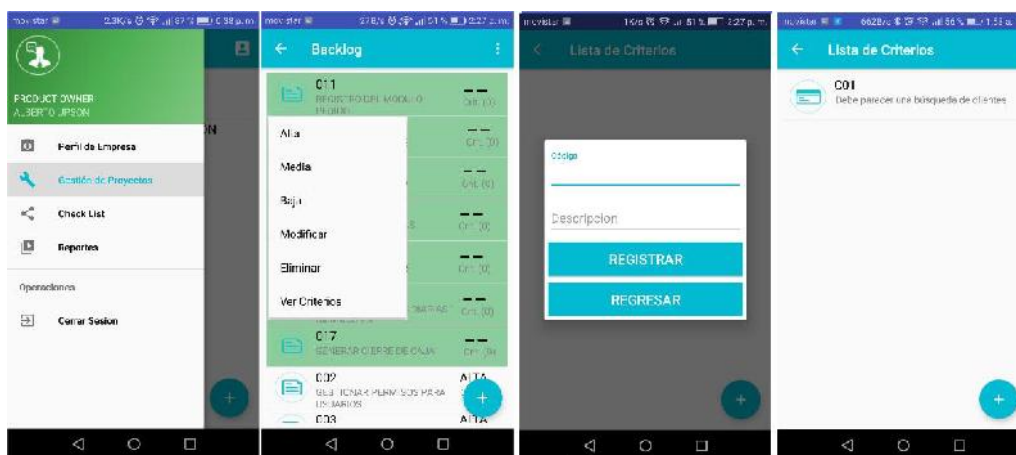
Código: 008

Nombre: Gestión de cuenta de Product Owner – Gestión de Historias de Usuario – Criterios de Aceptación.

Descripción: En este proceso el Product Owner podrá agregar los criterios de aceptación a las historias de usuario en el backlog.

Requisitos: Tener una cuenta y un usuario como PO generada por el Administrador y tener una lista de historias de usuarios con sus respectivos criterios de aceptación.

Interfaces:



Entradas: Código y descripción

Proceso: En este proceso el PO deberá seleccionar una historia de usuario, seleccionar en “Ver criterios”, se desplegará una pantalla que permitirá agregar criterios de aceptación siguiendo la siguiente estructura: Código y Descripción. Los criterios de aceptación se irán guardando internamente y aparecerán como números en las Historias de usuario.

Resultados esperados: Product Owner agregará los criterios de aceptación de las historias de usuario en el backlog además de poder editar, eliminar y posteriormente priorizar Historias de Usuario.

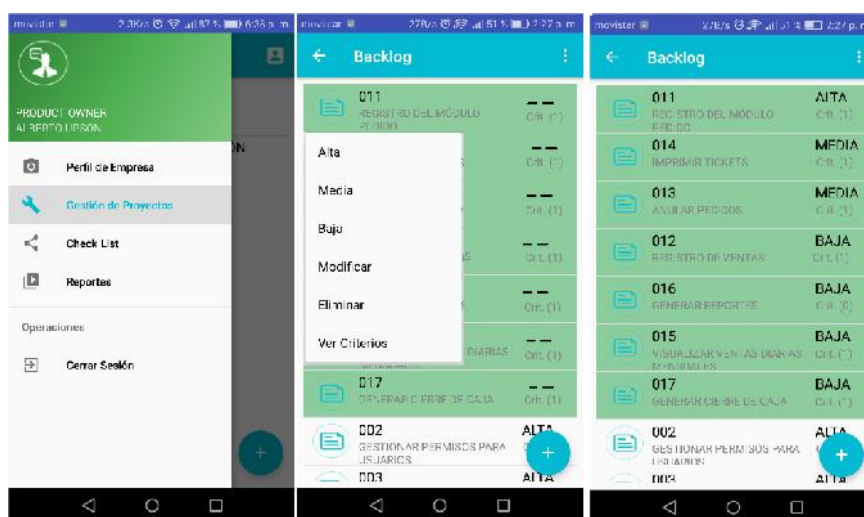
Código: 009

Nombre: Gestión de cuenta de Product Owner – Gestión de Historias de Usuario - Priorización.

Descripción: En este proceso el Product Owner podrá priorizar en 3 rangos: Alta, media y baja, además de ver los criterios de aceptación de las historias de usuario en el backlog.

Requisitos: Tener una cuenta y un usuario como PO generada por el Administrador y tener una lista de historias de usuario con sus criterios de aceptación.

Interfaces:



Entradas: Para priorizar las Historias de Usuario debemos seleccionar la historia de usuario.

Proceso: En este proceso el PO deberá seleccionar una historia de usuario, para poder asignarle una prioridad, que vaya de acorde al rango establecido: “Alta”, “Media” o “Baja”. Además de poder modificar, eliminar y poder visualizar los Criterios de Aceptación en cada HU.

Resultados esperados: El PO podrá priorizar una lista de historias de usuario en el backlog de acuerdo con el nivel de priorización que el PO considere. Además de poder modificar, eliminar y poder visualizar los Criterios de Aceptación.

3.5.4. Planificación de Sprints

Código: 010

Nombre: Gestión de cuenta de Product Owner – Gestión de Sprints – Creación de Sprints.

Descripción: En este proceso el Product Owner podrá crear Sprints de acuerdo a la prioridad de las historias de usuario, las HU de más alta prioridad irán en el Primer Sprint y así sucesivamente.

Requisitos: Como PO tener registrada una lista de historias de usuario ya priorizadas con sus respectivos criterios.

Interfaces:



Entradas: Historias de Usuario priorizadas con sus criterios de aceptación.

Proceso: En este proceso el PO deberá seleccionar “Nuevo Sprint”, se cargará las historias de usuario en orden desde la más alta a baja, podrá seleccionar las HU que irán en el primer Sprint y así irán cargando las demás HU hasta que todas estén asignadas a un sprint.

Resultados esperados: El PO creará y visualizará los Sprints con las historias de usuario priorizadas.

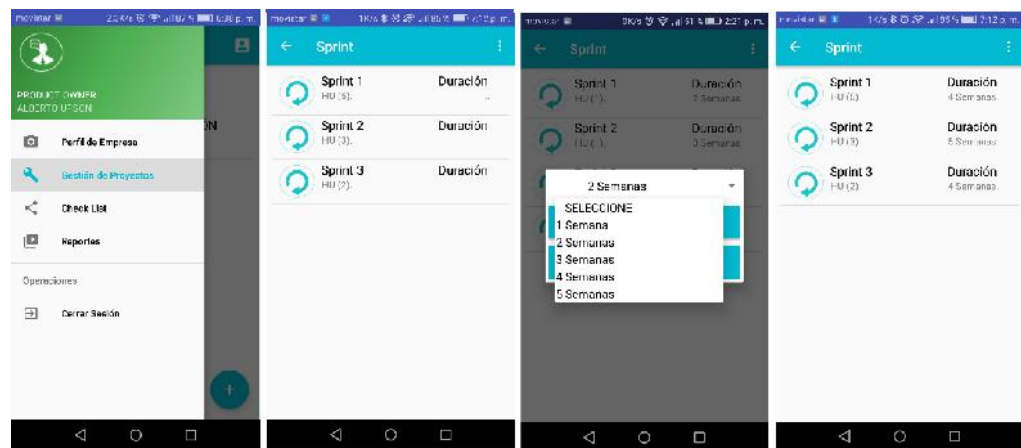
Código: 011

Nombre: Gestión de cuenta de Product Owner – Gestión de Sprints – Duración de Sprints.

Descripción: En este proceso el Product Owner podrá asignar un tiempo establecido en semanas para la realización de Sprints.

Requisitos: Como PO tener registrada una lista de Sprints.

Interfaces:



Entradas: Sprints asignados a cada proyecto.

Proceso: En este proceso el PO deberá seleccionar entre el siguiente rango: 1 – 5 semanas se cargará los Sprints en con la duración asignada.

Resultados esperados: El PO asignará una duración a los Sprints.

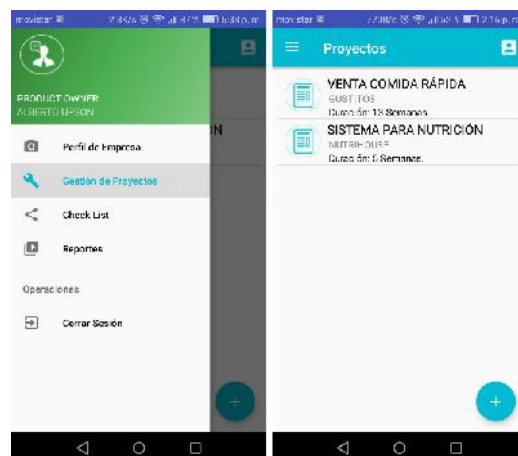
Código: 012

Nombre: Gestión de cuenta de Product Owner – Gestión de Proyecto – Duración por proyecto.

Descripción: En este proceso el Product Owner podrá visualizar el tiempo estimado por proyecto.

Requisitos: Como PO tener registrada una lista de proyectos a realizar.

Interfaces:



Entradas: Sprints con una duración asignada en semanas.

Proceso: En este proceso el PO al regresar a la pantalla del proyecto podrá ver su estimación en semanas.

Resultados esperados: El PO visualizará la duración del proyecto.

3.5.5. Desglose de Historias de Usuarios en tareas

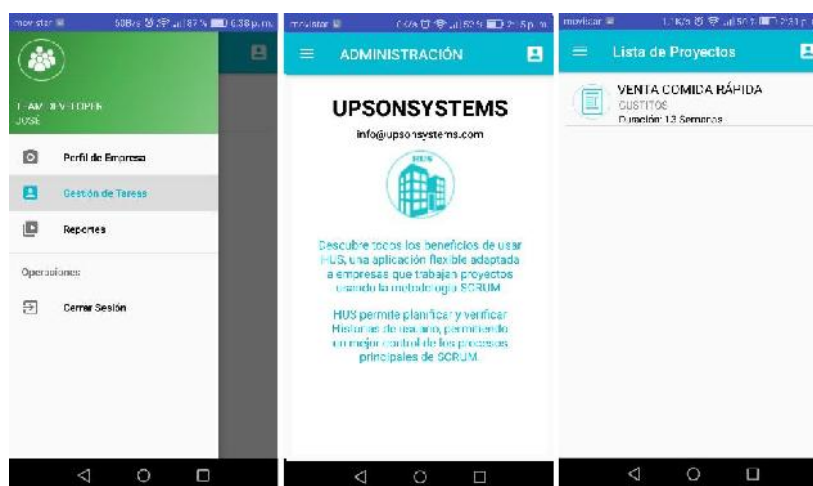
Código: 013

Nombre: Gestión de cuenta de Equipo de desarrollo

Descripción: En este proceso el Usuario asignado como parte del equipo de desarrollo podrá visualizar el Perfil de empresa, gestión de proyecto al que está asignado.

Requisitos: Tener una cuenta y un usuario como Equipo de desarrollo generada por el Administrador.

Interfaces:



Entradas: Nombre de usuario y contraseña.

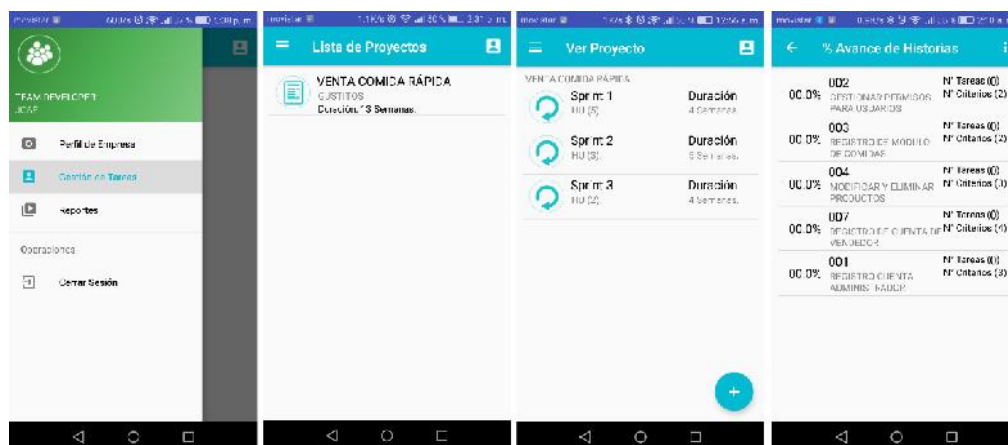
Proceso: En este proceso el usuario asignado como parte del equipo de desarrollo deberá iniciar sesión, una vez validados sus datos podrá tener acceso a Perfil de empresa (modo Visualización), y Gestión de tareas.

Resultados esperados: Podrá visualizar Perfil de empresa, gestión de proyecto al que está asignado.

Código: 014**Nombre: Gestión de cuenta de Equipo de desarrollo - Gestión de Sprints**

Descripción: En este proceso el Usuario asignado como parte del equipo de desarrollo podrá gestionar el proyecto al que está asignado y los Sprint que el PO creó anteriormente.

Requisitos: Tener Sprints creados por el PO.

Interfaces:

Entradas: Nombre de usuario y contraseña.

Proceso: En este proceso el usuario asignado como parte del equipo de desarrollo deberá seleccionar el Sprint a trabajar, el cual muestra la cantidad de HU. Posteriormente se desplegará el contenido del Sprint, se visualizará el Porcentaje de avance, la historia de Usuario con su código y nombre, la cantidad de criterios de aceptación por HU y finalmente la cantidad de tareas por HU.

Resultados esperados: Podrá visualizar los Sprints, las historias de usuario, el Porcentaje de avance, cantidad de criterios y podrá agregar tareas.

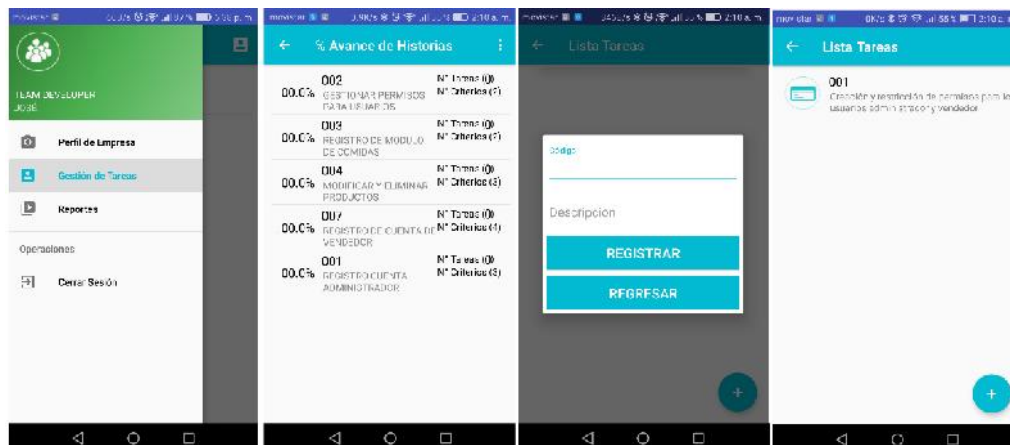
Código: 015

Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de Sprints – Creación de tareas.

Descripción: En este proceso el Usuario asignado como parte del equipo de desarrollo podrá definir y agregar tareas asignadas a las historias de usuario.

Requisitos: Tener Sprints creados por el PO y la lista de HU.

Interfaces:



Entradas: Código y descripción de las tareas.

Proceso: En este proceso el usuario asignado como parte del equipo de desarrollo deberá seleccionar la historia de usuario a trabajar, se desplegará una interfaz para agregar tareas, seleccionamos el ícono de agregar y llenaremos los datos para registrar las tareas.

Resultados esperados: Podrá crear y visualizar la lista de tareas asignadas a cada HU.

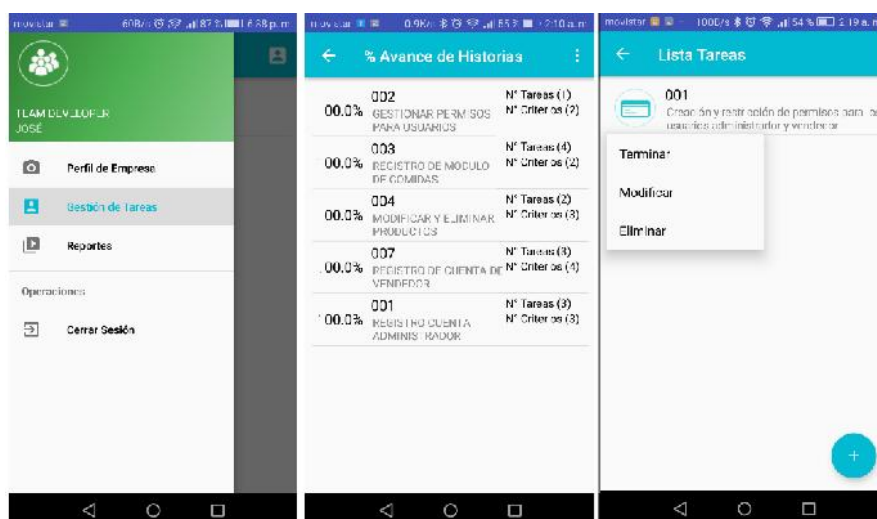
Código: 016

Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de tareas

Descripción: En este proceso el Usuario asignado como parte del equipo de desarrollo podrá editar, eliminar y terminar una tarea cuando esté realizada de acuerdo a las HU.

Requisitos: Tener una lista de tareas creadas.

Interfaces:



Entradas: Lista de Tareas.

Proceso: En este proceso el usuario asignado como parte del equipo de desarrollo podrá seleccionar una tarea para poder “terminar” si el desarrollo de la misma se aprobó y se da por finalizado por el equipo de desarrollo; del mismo modo la tarea puede ser modificada y eliminada siempre y cuando no haya sido terminada.

Resultados esperados: Podrá terminar, modificar y eliminar las tareas asignadas a cada HU.

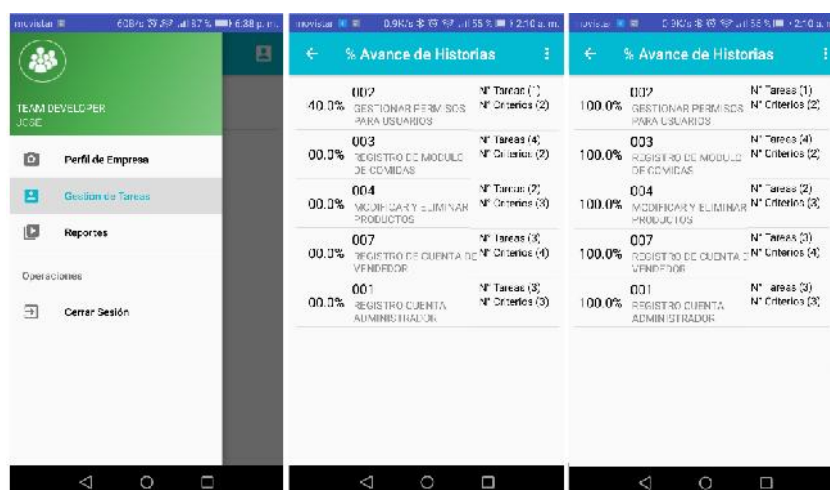
Código: 017

Nombre: Gestión de cuenta de Equipo de desarrollo – Gestión de tareas - Avance

Descripción: En este proceso el Usuario asignado como parte del equipo de desarrollo podrá ver el avance de las tareas y visualizarlo de acuerdo a porcentajes para saber si una historia de usuario ya está completa.

Requisitos: Tener una lista de las tareas por historias de usuario.

Interfaces:



Entradas: Lista de HU con sus respectivas tareas.

Proceso: En este proceso el usuario asignado como parte del equipo de desarrollo podrá seleccionar una tarea para poder “terminar”, de acuerdo al avance de las tareas el porcentaje irá cambiando hasta que las tareas de las historias de usuario hayan sido totalmente completadas.

Resultados esperados: Visualización general de la historia de usuario, número de criterios, número de tareas y el porcentaje de avance de las tareas con sus respectivas historias de usuario.

3.5.6. Verificación del cumplimiento de historias de usuario

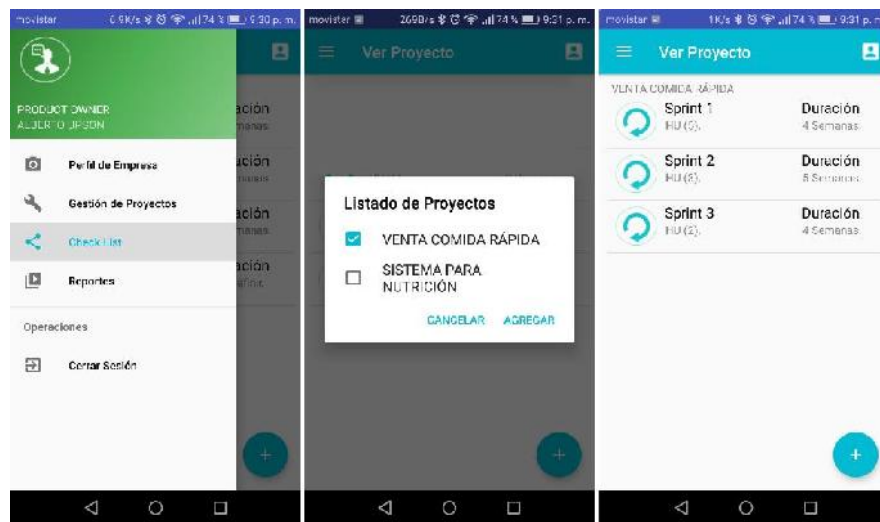
Código: 018

Nombre: Gestión de cuenta de PO – checklist

Descripción: En este proceso el PO, accederá al checklist seleccionará el proyecto que desea analizar y visualizará los Sprints que contiene.

Requisitos: Tener una lista de las tareas terminadas, con esto obtendremos una lista de Sprints terminados para poder visualizar.

Interfaces:



Entradas: Proyecto, historias de usuario y Sprints.

Proceso: En este proceso el PO accediendo al checklist se desplegará un menú con la lista de proyectos registrados, se podrá seleccionar el proyecto que desea verificar, una vez seleccionado se mostrarán los Sprints del proyecto.

Resultados esperados: Visualización de los Sprints en el proyecto con la cantidad de historias de usuario que contiene.

Código: 019

Nombre: Gestión de cuenta de Product Owner – checklist – criterios de aceptación

Descripción: En este proceso el Product Owner, accederá al checklist para dar conformidad a los criterios de aceptación que asignó en un inicio, esto se trabajará por cada proyecto.

Requisitos: Tener una lista de las tareas terminadas y el sprint a analizar seleccionado.

Interfaces:



Entradas: Sprint, historias de usuario y criterios de aceptación.

Proceso: En este proceso el PO accediendo al checklist se desplegará el sprint seleccionado, se mostrarán los criterios con la opción de validar con check los criterios que para el PO llegaron a cumplirse.

Resultados esperados: Visualización de los Sprints en el proyecto con la cantidad de historias de usuario y aprobación del cumplimiento de los criterios de aceptación establecidos.

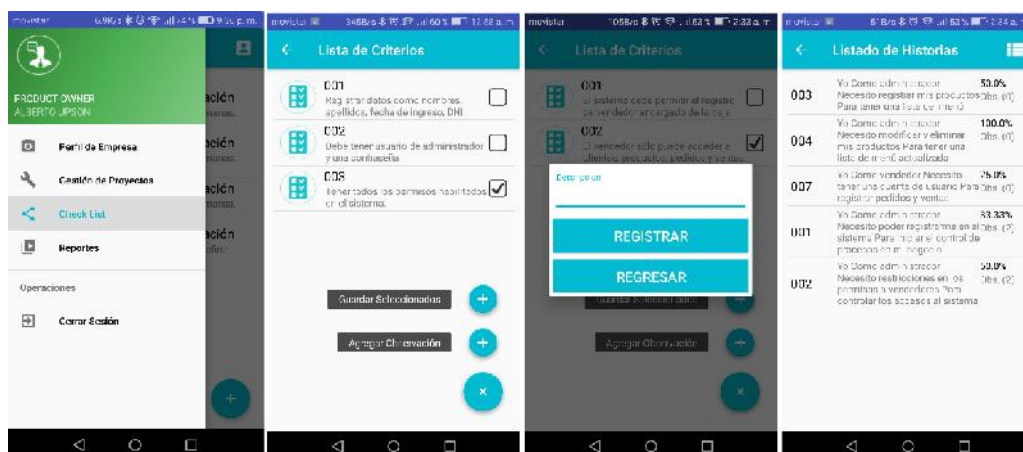
Código: 020

Nombre: Gestión de cuenta de PO – checklist –Observaciones en los criterios de aceptación

Descripción: En este proceso el PO, accederá al checklist para dejar una observación en caso los criterios de aceptación no se hayan cumplido tal cual lo solicitó.

Requisitos: Tener una lista de los criterios con un porcentaje del 100%, es decir que los criterios estén terminados, listos para ser visualizados por el Product Owner.

Interfaces:



Entradas: Historias de usuario y criterios de aceptación terminados.

Proceso: En este proceso el PO dará la conformidad del desarrollo de los criterios de aceptación y tendrá la posibilidad de agregar una observación de ser necesario si no cumplió del todo lo que inicialmente pidió.

Resultados esperados: El PO podrá dar por aprobado los criterios de aceptación y además podrá agregar observaciones adicionales.

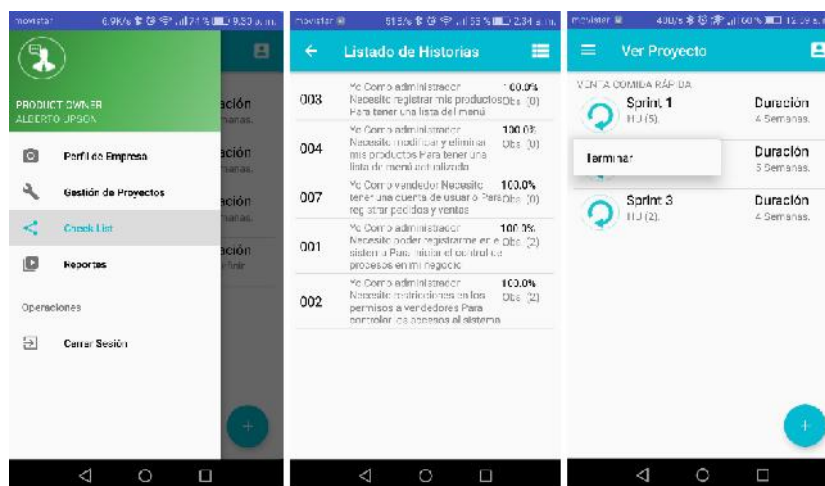
Código: 021

Nombre: Gestión de cuenta de PO – checklist – Sprints terminados

Descripción: En este proceso el PO, accederá a la lista de Sprints y podrá decidir si está terminado.

Requisitos: Tener una lista de las historias de usuario con un porcentaje del 100%, es decir que el equipo de desarrollo al terminar las tareas y cumplir con los criterios, las historias de usuarios se muestren como terminadas.

Interfaces:



Entradas: Historias de usuario y criterios de aceptación terminados.

Proceso: En este proceso el PO dará la conformidad del desarrollo de los Sprints pudiendo dar por terminados los que crea convenientes.

Resultados esperados: El PO podrá dar por terminado Sprints.

3.5.7. Monitoreo de planificación y verificación

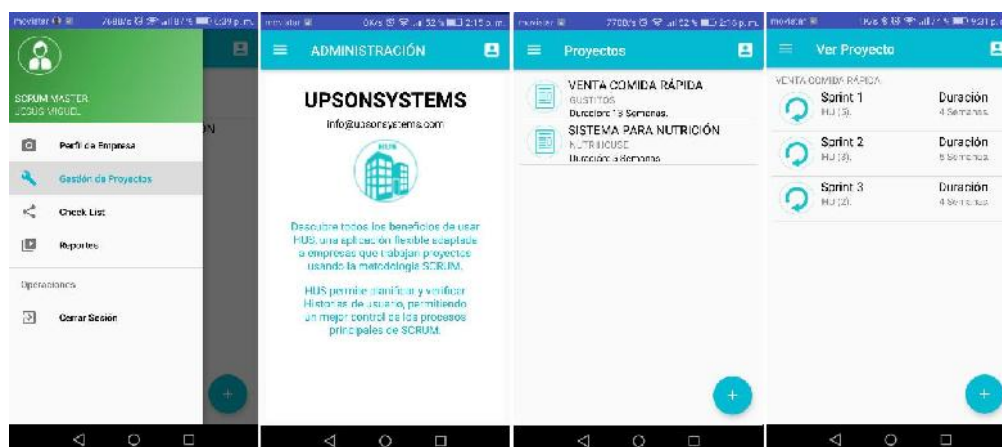
Código: 022

Nombre: Gestión de cuenta de SM.

Descripción: En este proceso el SM, podrá acceder a todas las pantallas del proyecto excepto las que maneja el administrador.

Requisitos: Tener un usuario y contraseña como SM.

Interfaces:



Entradas: Usuario y contraseña.

Proceso: En este proceso el SM podrá visualizar, editar y eliminar historias de usuario, Sprints, criterios que considere alejados de los conceptos manejados por SCRUM.

Resultados esperados: el SM visualizara, editara y eliminara campos que considere no se alinean a Scrum, con previo consentimiento del equipo SCRUM.

3.5.8. Reportes y resultados

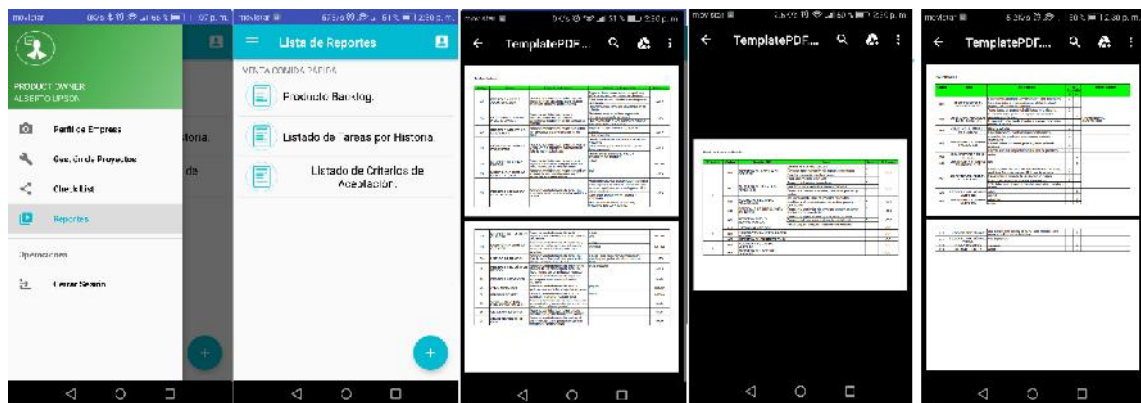
Código: 023

Nombre: Reportes para todos los usuarios

Descripción: En este proceso el SM, PO y el TD podrán acceder a todos los reportes.

Requisitos: Tener un usuario y contraseña por cada rol asignado.

Interfaces:



Entradas: Usuario y contraseña.

Proceso: En este proceso el SM, PO y TD podrán visualizar, un listado de reportes para poder conocer todo el proceso de desarrollo del proyecto.

Resultados esperados: el SM, PO y TD Visualizarán los reportes en formato PDF.

3.6. FICHA TÉCNICA

A continuación, se mostrará características de la aplicación de manera más detallada y resumida que sirvió como base para el desarrollo y servirán a posteriores desarrolladores a comprender de manera sencilla.

I. DATOS GENERALES			
Nombre de la Aplicación	HUS		
Descripción	Aplicación móvil para equipos móviles con sistema operativo Android, apoya en la planificación y verificación de Historias de Usuario, basado en la metodología SCRUM.		
Servidor		Sistema Operativo	Windows
URL de Acceso (Intranet/Internet)	camjpsystems.ddns.net/SCRUM/download/		
Fecha Creación Documento	30/12/2017		
Fecha última Actualización	30/02/2018		

II. DATOS BÁSICOS APLICACIÓN					
Tipo de puesta:	<input checked="" type="checkbox"/> Producción <input type="checkbox"/> Pruebas				
Tipo de aplicación:	<input type="checkbox"/> WEB <input type="checkbox"/> Escritorio <input type="checkbox"/> Stand Alone <input checked="" type="checkbox"/> Movil				
Servidor Web:	<input checked="" type="checkbox"/> Tomcat <input type="checkbox"/> Apache <input type="checkbox"/> IIS	Versión: 8.0.18			
Código ejecutado en el cliente:	<input checked="" type="checkbox"/> JAVA <input type="checkbox"/> JAVASCRIPT <input type="checkbox"/> ACTION SCRIPT <input type="checkbox"/> HTML5 <input type="checkbox"/> ACTIVE X <input type="checkbox"/> Otro ¿Cuál? _____				
Código ejecutado en el servidor:	<input type="checkbox"/> VB <input type="checkbox"/> PL / SQL <input type="checkbox"/> PHP Cuál _____ <input type="checkbox"/> .Net <input checked="" type="checkbox"/> JAVA <input type="checkbox"/> XML _____ <input type="checkbox"/> C# <input type="checkbox"/> Otro _____ <input type="checkbox"/> .Net _____ <input type="checkbox"/> ASP				

¿Requiere base de datos?	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO	Cuál: <input type="checkbox"/> SQL Server <input type="checkbox"/> PostgreSQL <input type="checkbox"/> Oracle <input type="checkbox"/> MySQL <input checked="" type="checkbox"/> Postgres SQLite <input type="checkbox"/> Otro Cuál _____
¿Interactúa con otra aplicación?	<input checked="" type="checkbox"/> SI <input type="checkbox"/> NO Con una web service REST.	

III. REQUERIMIENTOS DE SOFTWARE PARA EL SERVIDOR		
DE SISTEMA OPERATIVO		
Tipo	Versión/Distribución	Arquitectura
<input checked="" type="checkbox"/> Windows <input type="checkbox"/> Linux <input type="checkbox"/> Otro	2008	<input type="checkbox"/> 32 bits <input type="checkbox"/> 64 bits <input checked="" type="checkbox"/> x86
DE APLICACIÓN		
Lenguaje/Servicio	Versión	Bibliotecas
JAVA	9.3	JDK
DE BASE DE DATOS		
Manejador	Versión	Puerto
PostGres	9.3	5432
Observaciones: Configurar solo en el puerto 5432.		

IV. REQUERIMIENTOS DE HARDWARE PARA EL SERVIDOR	
Tipo	Características
Procesador	2.4 GHz
Memoria	2 GB RAM
Espacio en disco	500GB

V. ARCHIVOS DE CONFIGURACIÓN O PARAMETRIZACIÓN				
Directorio	Nombre del archivo	Variabl e/Tipo de variabl e	Nombre de la variable	Descripción
Tomcat	Server.xml	Host	name	Para establecer la URL de conexión.

VI. INFORMACION DE BASES DE DATOS ASOCIADA/S A LA APLICACION			
Manejador	<input type="checkbox"/> SQL Server <input checked="" type="checkbox"/> Postgres <input type="checkbox"/> Otro <input type="checkbox"/> Oracle <input type="checkbox"/> MySQL Cuál _____		
Nombre Servidor de BD	ERICK		
Nombre BD	db_hus	Usuario	erick
Espacio en disco (actual)	500GB	Proyección a 1 año	500GB
Directorio de Backup	D://BACKUP/BASE/		
Periodicidad	<input type="checkbox"/> Diaria <input checked="" type="checkbox"/> Semanal <input type="checkbox"/> Mensual		

CAPÍTULO IV. CONTRASTACIÓN DE HIPÓTESIS

En este capítulo detallamos nuestra hipótesis y comprobar si se acepta o se rechaza de acuerdo a los indicadores planteados, además nos permitirá analizar la influencia y la relación que hay entre las variables dependientes e independientes.

4.1. HIPÓTESIS

Una aplicación móvil en plataforma Android, basado en SCRUM, permite desarrollar las actividades de planificación y verificación de historias de usuario en proyectos de desarrollo de software

4.2. IDENTIFICACIÓN DE VARIABLES

4.2.1. Variables

Variable independiente: Aplicación móvil en plataforma Android, basado en SCRUM.

Variable dependiente: Planificación y verificación de historias de usuario en proyectos de desarrollo de software

4.2.2. Cuadro de operacionalización de variables

El modelo conceptual de la investigación que se grafica a continuación muestra las dimensiones que se evaluarán para contrastar la hipótesis:

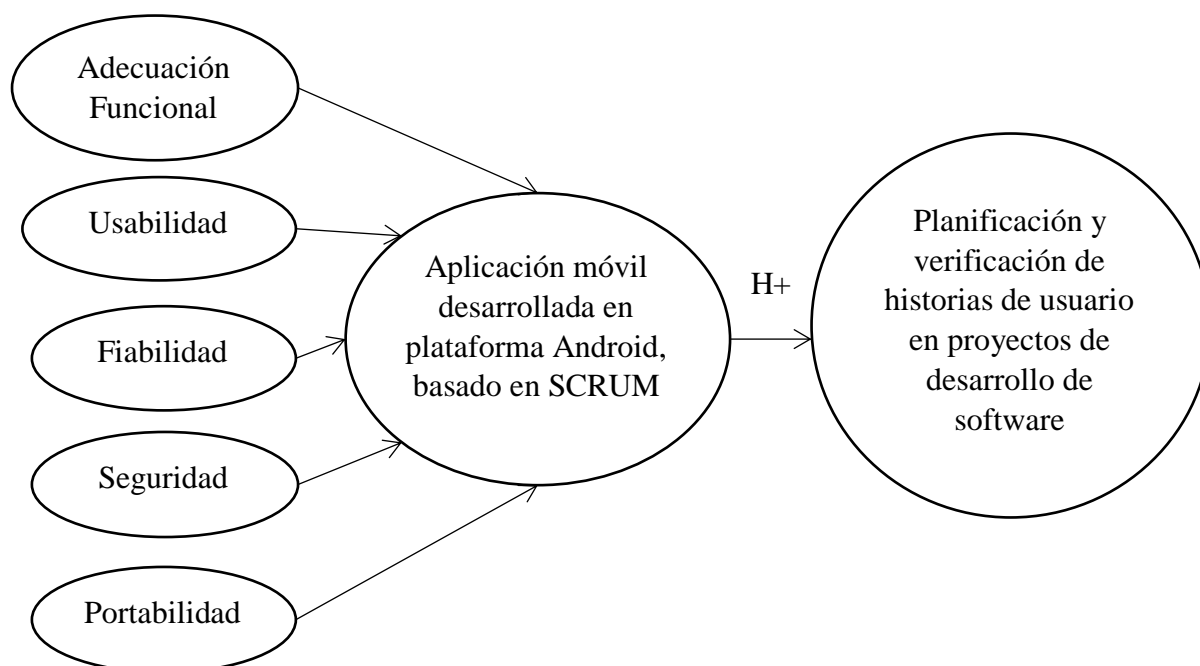


Ilustración 27: Modelo conceptual de la investigación

Nota. Fuente: Creación propia.

La tabla siguiente muestra los indicadores que se obtendrán para cada una de las dimensiones consideradas en la evaluación de la variable independiente, que es la variable que se va a manipular y en qué escala de medición vamos a tomar el valor de cada indicador.

Tabla 40: Tabla de indicadores.

VARIABLE	DIMENSION	INDICADOR	ESCALA
INDEPENDIENTE Aplicación móvil desarrollada en plataforma Android, basado en SCRUM	Adecuación funcional	Funciones que cubren los objetivos de los usuarios.	Likert de 5 niveles
		Genera resultados correctos	Likert de 5 niveles
	Usabilidad	Usuarios que entienden el software.	Likert de 5 niveles
		Operatividad y control el software	Likert de 5 niveles
		Interacción amigable y agradable.	Likert de 5 niveles
	Fiabilidad	Accesibilidad al Software.	Likert de 5 niveles
		Recuperación de información ante fallas o pérdidas.	Likert de 5 niveles
	Seguridad	Identificación de usuarios según tipo (Autenticidad).	Likert de 5 niveles
		Acceso solo a usuarios autorizados	Likert de 5 niveles
	Portabilidad	Compatibilidad con varias versiones del sistema operativo y equipos.	Likert de 5 niveles
		Facilidad para instalar o desinstalar.	Likert de 5 niveles
DEPENDIENTE Planificación y verificación de historias de usuario en proyectos de desarrollo de software	Satisfacción del usuario	Grado de cumplimiento de las actividades de Planificación y verificación.	Likert de 5 niveles

Fuente: Creación propia.

4.3. DISEÑO DE CONTRASTACIÓN DE HIPÓTESIS

De acuerdo al modelo conceptual de la investigación para la contratación de la hipótesis se utilizó un método relacional, porque se tienen como propósito medir el grado de relación que exista entre las dos variables definidas: Aplicación móvil en plataforma Android, basado en SCRUM, permite desarrollar las actividades de planificación y verificación de historias de usuario en proyectos de desarrollo de software.

El modelo lógico de contrastación es del tipo cuasi experimental porque no se tendrá un control efectivo de las variables seleccionadas. Es decir, no se puede seleccionar aleatoriamente los proyectos de desarrollo de software para las tareas experimentales y la información que se obtenga será de los proyectos de desarrollo de software que las empresas encuestadas nos brinden.

El modelo a utilizar es el siguiente:

GE: X r Y

Dónde:

X: Aplicación móvil desarrollada en plataforma Android.

Y: Planificación y verificación de historias de usuario basado en la metodología SCRUM.

El diseño para la contrastación de la hipótesis buscará evaluar las siguientes consecuencias lógicas:

- a. Nivel de experiencia en el desarrollo de software usando SCRUM.
- b. Tamaño de la empresa.
- c. Tipos de proyectos.
- d. Formas de planificación de los proyectos que manejan.
- e. Formas de verificación de historias de usuario.
- f. Funcionalidad requerida para proyectos.

4.4. POBLACIÓN Y MUESTRA DE ESTUDIO

Unidad de Análisis: responsables de la planificación de proyectos en las empresas de desarrollo y áreas de sistemas donde se gestionan productos de software en la ciudad de Chiclayo Perú, siendo un total de 16 personas consideradas por conveniencia para realizar nuestro estudio.

Población: Se tomará como población muestral empresas dedicadas al desarrollo y empresas que tengan un área de sistemas donde se desarrollen productos de software basados en la metodología SCRUM, en la ciudad de Chiclayo - Perú.

Debido a que no existe un registro oficial de empresas de desarrollo de software que declaren la aplicación de la metodología SCRUM como framework de trabajo en la ciudad de Chiclayo, las empresas tomadas como caso de estudio han sido seleccionadas intencionalmente y por conveniencia, en base a recomendación de docentes, por búsqueda por internet y tomando como referencia base de datos de la Superintendencia de Registros Públicos y la Cámara de Comercio de Chiclayo. Las dos empresas seleccionadas como unidades de análisis han sido las que nos han permitido acceder a la información de sus proyectos de desarrollo de software con los cuales se ha trabajado en esta investigación.

- **Técnicas de muestreo:** La técnica de muestreo es No Probabilística por conveniencia, debido a que se seleccionó sólo aquellos proyectos donde se aplica la Metodología SCRUM.

Se aplicó una encuesta a los 16 usuarios de la muestra planteada, obteniendo las mismas 16 encuesta válidas y consistente (contestaron sin errores y todas las preguntas). Ésta se aplicó durante el mes de abril.

Tabla 41 Trabajadores de Upson Systems – Chiclayo

Tipo de usuario/cliente	Nº Usuarios
Scrum Master	1
Product Owner	1
Equipo de desarrollo	4
TOTAL	6

Nota. Fuente: Creación propia.

Tabla 42 Trabajadores de Despensa peruana S.A. – Chiclayo

Tipo de usuario/cliente	Nº Usuarios
Scrum Master	2
Product Owner	2
Equipo de desarrollo	6
TOTAL	10

Nota. Fuente: Creación propia.

Tabla 43 Total de trabajadores encuestados

Tipo de usuario/cliente	Nº Usuarios
Scrum Master	3
Product Owner	3
Equipo de desarrollo	10
TOTAL	16

Nota. Fuente: Creación propia.

4.5. TÉCNICAS DE RECOPIACIÓN DE DATOS

En este capítulo, luego de presentar el aplicativo final a la empresa, se procedió a realizar la recolección de los datos para poder analizarlos y medir las variables planteadas inicialmente, para luego llegar a la conclusión que permita saber si con la consecución del sistema se cumplen los objetivos planteados.

Las técnicas para la recopilación de datos se muestran a continuación:

4.5.1. Técnicas e instrumentos de recolección de datos

Las técnicas para la recolección de datos fue la siguiente:

Encuesta:

La encuesta fue diseñada de tal forma que sea compatible con los indicadores que se desean evaluar en esta investigación. Para ello se elaboró la siguiente tabla que muestra la relación de las preguntas diseñadas en la encuesta con los correspondientes indicadores que permiten medirlo con la información recopilada.

Tabla 44: Matriz de consistencia entre los indicadores y las preguntas de la encuesta

DIMENSION	INDICADOR	PREGUNTAS
Adecuación funcional	Funciones que cubren los objetivos de los usuarios.	a) ¿Cómo califica la capacidad de la aplicación para cumplir con los objetivos de planificación de Historias de usuario según Scrum?
		b) ¿Cómo califica la capacidad de la aplicación para cumplir con los objetivos en cuanto a verificación de historias de Usuario según Scrum?
	Genera resultados correctos.	c) ¿Cómo califica la capacidad de la aplicación para cumplir con los objetivos en cuanto a verificación de historias de Usuario según Scrum?
Usabilidad	Usuarios que entienden el software.	a) ¿Cómo califica la capacidad de entender el funcionamiento la aplicación?
	Operatividad y control el software.	b) ¿Cómo califica las características de la aplicación para ser manipulada y controlada?
	Interacción amigable y agradable.	c) ¿Cómo califica la capacidad de aprender a usar el sistema?
Fiabilidad	Accesibilidad al Software.	a) ¿Cómo califica el acceso a la aplicación cuando requiere registrar u operar información?
	Recuperación de información ante fallas o pérdidas.	b) ¿Cómo califica el comportamiento de la aplicación para recuperar datos en caso de interrupción y fallos?

Seguridad	Identificación de usuarios según tipo (Autenticidad).	a) ¿Cómo califica la forma de acceder a los recursos de la aplicación?
	Acceso solo a usuarios autorizados.	b) ¿Cómo califica la privacidad de la información ingresada por parte los usuarios en la aplicación?
Portabilidad	Compatibilidad con varias versiones del sistema operativo y equipos.	a) ¿Cómo califica la capacidad de adaptabilidad de la aplicación a la versión de Android de su dispositivo?
	Facilidad para instalar o desinstalar.	b) ¿Cómo califica la forma y el tiempo que se invirtió para la instalación del sistema en su dispositivo móvil?
		c) ¿Cómo califica la forma y el tiempo que se invirtió para la desinstalación del sistema en su dispositivo móvil?
Satisfacción de usuarios	Grado de cumplimiento de las actividades de Planificación y verificación.	a) ¿Cómo fue la calidad de servicio que brinda la aplicación con respecto al rol que Ud. Desempeña?
		b) ¿Qué tan satisfecho está usted con esta aplicación?
		c) ¿Cree que la aplicación es una buena opción para mejorar los procesos en SCRUM?
		d) ¿Cree que la aplicación de alinea al proceso de planificación y verificación de Historias de Usuario?

Fuente: Creación propia.

4.6. TRATAMIENTO DE LOS DATOS Y DISCUSIÓN DE RESULTADOS

Para el tratamiento de los datos, se utilizó el aplicativo SPSS v 23, obteniéndose los siguientes resultados:

4.6.1. Fiabilidad del instrumento (Encuesta)

Usamos el estadístico Alfa de Cronbach, como método estadístico para determinar el nivel de fiabilidad del instrumento (la encuesta) entre los indicadores de las variables, dependiente e independiente medidos bajo la escala Likert (5 niveles). Como se cita en Almanza Jiménez & Vargas Hernández (2015), con este método se mide el mismo constructo y la correlación entre variables. Cuanto más cerca se encuentre el valor del alfa al 1 mayor es la consistencia interna de los ítems analizados.

Como criterio general George y Mallery (2003) sugieren las evaluaciones siguientes para evaluar los coeficientes de Alfa de Cronbach:

Coefficiente alfa $> .9$ es excelente.

Coefficiente alfa $> .8$ es bueno.

Coefficiente alfa $> .7$ es aceptable.

Coefficiente alfa $> .6$ es cuestionable.

Coefficiente alfa $> .5$ es pobre.

Procesados los datos se obtuvo lo siguiente:

Resumen de procesamiento de casos

	N	%
Casos Válido	16	100,0
Excluido ^a	0	,0
Total	16	100,0

a. La eliminación por lista se basa en todas las variables del procedimiento.

Estadísticas de fiabilidad

Alfa de Cronbach	N de elementos
,746	12

En este caso se ha alcanzado un coeficiente de Alfa de Cronbach de 0.746, y según las recomendaciones para evaluar a éste, nos indica que la encuesta aplicada es aceptable. Además, se observa que los datos procesados corresponden a 16 encuestas, el total de la muestra encuestada, no habiendo sido ninguna encuesta excluida del procesamiento de datos.

4.6.2. Análisis de la regresión múltiple

Utilizamos regresión múltiple porque nuestra hipótesis pretende estudiar la posible relación entre las variables independientes (predictoras o explicativas) y la variable dependiente (criterio, explicada, respuesta). En este caso, nuestras variables son:

- **Variable Independiente (Xi):** Aplicación móvil desarrollada en plataforma Android, descrita a través de las dimensiones, Adecuación funcional (X_1), Usabilidad (X_2), Fiabilidad (X_3), Seguridad (X_4) y Portabilidad (X_5).
- **Variable dependiente (Y):** Planificación y verificación de historias de usuario en proyectos de desarrollo de software

Por tanto, el modelo a evaluar es un modelo de regresión múltiple de la forma:

$$Y = C_0 + C_1X_1 + C_2X_2 + C_3X_3 + C_4X_4 + C_5X_5 + E$$

Esto significa que se pretende evaluar la relación existente entre la variable dependiente “Planificación y verificación de historias de usuario en proyectos de desarrollo de software” y la variable independiente “Aplicación móvil desarrollada en plataforma Android, basado en Scrum”, esta última explicada por cinco dimensiones: Adecuación funcional (X_1), Usabilidad (X_2), Fiabilidad (X_3), Seguridad (X_4) y Portabilidad (X_5).

Para lograr este objetivo, se desarrolló el siguiente procedimiento:

a. Reducción de ítems de cada dimensión evaluada

Dado que cada una de las dimensiones tiene más de un ítem a evaluar (ver Tabla N° 37) se tuvo que reducir a un solo ítem, de la siguiente manera:

Tabla 45 Matriz de reducción de ítems evaluados

DIMENSION	INDICADOR	PREGUNTAS	ÍTEM REDUCIDO
Adecuación funcional (X_1)	Funciones que cubren los objetivos de los usuarios.	P_1	$(X_1) = \frac{P_1 + P_2 + P_3}{3}$
		P_2	
	Genera resultados correctos.	P_3	
Usabilidad (X_2)	Usuarios que entienden el software.	P_4	$(X_2) = \frac{P_4 + P_5 + P_6}{3}$
	Operatividad y control el software.	P_5	
	Interacción amigable y agradable.	P_6	
Fiabilidad (X_3)	Accesibilidad al Software.	P_7	$(X_3) = \frac{P_7 + P_8}{2}$
	Recuperación de información ante fallas o pérdidas.	P_8	
Seguridad (X_4)	Identificación de usuarios según tipo (Autenticidad).	P_9	$(X_4) = \frac{P_9 + P_{10}}{2}$
	Acceso solo a usuarios autorizados.	P_{10}	
Portabilidad (X_5)	Compatibilidad con varias versiones del sistema operativo y equipos.	P_{11}	$(X_5) = \frac{P_{11} + P_{12} + P_{13}}{3}$
		P_{12}	
	Facilidad para instalar o desinstalar.	P_{13}	
Satisfacción de usuarios	Grado de cumplimiento de las actividades de Planificación y verificación.	P_{14}	$(Y) = \frac{P_{14} + P_{15} + P_{16} + P_{17}}{4}$
		P_{15}	
		P_{16}	
		P_{17}	

Nota. Fuente: Creación propia.

b. Aplicación de la metodología de regresión múltiple

Para nuestro análisis se aplicará la metodología de regresión múltiple jerárquica con cinco bloques, donde se fueron tomando variable por variable independiente con las que estamos trabajando, con la finalidad de generar diferentes modelos. Los modelos que esperamos generar son los siguientes:

- **Modelo 1:** sólo con la variable Adecuación funcional (X_1)
- **Modelo 2:** sólo con las variables Adecuación funcional (X_1), Usabilidad (X_2),
- **Modelo 3:** con las tres variables Adecuación funcional (X_1), Usabilidad (X_2), Fiabilidad (X_3),
- **Modelo 4:** sólo con las variables Adecuación funcional (X_1), Usabilidad (X_2), Fiabilidad (X_3), Seguridad (X_4)
- **Modelo 5:** con las tres variables Adecuación funcional (X_1), Usabilidad (X_2), Fiabilidad (X_3), Seguridad (X_4) y Portabilidad (X_5).

Esto nos permitirá identificar más información de las variables independientes con las que estamos trabajando; así como también nos permite identificar si alguna de esas variables independientes no aporta al modelo, por tanto, puede ser excluida del modelo. Los resultados obtenidos se muestran a continuación.

Resumen del modelo					
Modelo	R	R cuadrado	R cuadrado ajustado	Error estándar de la estimación	Durbin-Watson
1	,157 ^a	,025	-,045	,528	
2	,515 ^b	,265	,152	,476	
3	,541 ^c	,293	,116	,485	
4	,606 ^d	,367	,137	,480	
5	,635 ^e	,403	,104	,489	2,861

a. Predictores: (Constante), ADEC_FINAL

b. Predictores: (Constante), ADEC_FINAL, USA_FINAL

c. Predictores: (Constante), ADEC_FINAL, USA_FINAL, FIABI_FINAL

d. Predictores: (Constante), ADEC_FINAL, USA_FINAL, FIABI_FINAL, SEGUR

e. Predictores: (Constante), ADEC_FINAL, USA_FINAL, FIABI_FINAL, SEGUR, PORTA

f. Variable dependiente: SATIS

Del cuadro se deduce que:

- El Modelo 1 (sólo con la variable Adecuación Funcional (X_1)) explica el 2.5% de la varianza de la variable dependiente.
- El Modelo 2 (sólo con las variables Adecuación funcional (X_1) y Usabilidad (X_2), explica el 26.5% de la varianza de la variable dependiente.
- El Modelo 3 con las tres variables Adecuación funcional (X_1), Usabilidad (X_2) y Fiabilidad (X_3), explica el 29.3% de la varianza de la variable dependiente.
- En el Modelo 4: sólo con las variables Adecuación funcional (X_1), Usabilidad (X_2), Fiabilidad (X_3) y Seguridad (X_4), explica el 36.7% de la varianza de la variable dependiente.
- Modelo 5: con las variables Adecuación funcional (X_1), Usabilidad (X_2), Fiabilidad (X_3), Seguridad (X_4) y Portabilidad (X_5), explica el 40.3% de la varianza de la variable dependiente.

Para efectos de la demostración de la hipótesis seleccionamos el Modelo 5 donde se incluyen las cinco variables independientes.

Por otro lado, en el mismo cuadro observamos el resultado de la prueba de Durbin-Watson que nos da un valor para determinar la independencia de errores, pero no una significancia; por lo que tenemos que tener algunos criterios de identificación de cuando este valor es bueno o no bueno.

El valor esperado de la prueba Durbin-Watson es que sea lo más cercano a 2, en este caso tenemos un valor de 2.861 que es bueno. El rango que se debe tener en cuenta para aceptar el resultado de la prueba de Durbin-Watson es 1 ± 2 , es decir entre 1 y 3.

La interpretación de este resultado es que no existe dependencia de las observaciones recogidas, por lo tanto, se demuestra que la recogida de la información ha sido aleatoria, evitando así invalidar por completo las conclusiones del análisis estadístico (obteniendo conclusiones erróneas).

c. Anova:

Modelo		Suma de cuadrados	gl	Media cuadrática	F	Sig.
1	Regresión	,098	1	,098	,352	,062 ^b
	Residuo	3,902	14	,279		
	Total	4,000	15			
2	Regresión	1,060	2	,530	2,342	,035 ^c
	Residuo	2,940	13	,226		
	Total	4,000	15			
3	Regresión	1,173	3	,391	1,659	,028 ^d
	Residuo	2,827	12	,236		
	Total	4,000	15			
4	Regresión	1,469	4	,367	1,596	,024 ^e
	Residuo	2,531	11	,230		
	Total	4,000	15			
5	Regresión	1,611	5	,322	1,349	,0320 ^f
	Residuo	2,389	10	,239		
	Total	4,000	15			

a. Variable dependiente: SATIS

b. Predictores: (Constante), ADEC_FINAL

c. Predictores: (Constante), ADEC_FINAL, USA_FINAL

d. Predictores: (Constante), ADEC_FINAL, USA_FINAL, FIABI_FINAL

e. Predictores: (Constante), ADEC_FINAL, USA_FINAL, FIABI_FINAL, SEGUR

f. Predictores: (Constante), ADEC_FINAL, USA_FINAL, FIABI_FINAL, SEGUR, PORTA

Como el modelo de regresión que estamos trabajando es saber si las cinco variables independientes están prediciendo la variable dependiente, entonces nos quedamos con los resultados del segundo modelo (Modelo 2) que se muestra en la tabla ANOVA que tiene el mayor valor de Fisher ($F=2.342$). Aquí se observa que hay una significancia menor al 0.05 ($0.035 \leq 0.05$) y la interpretación en términos de hipótesis es que el modelo que estamos probando mejora significativamente la predicción de la variable dependiente.

d. Análisis de coeficiente de la ecuación de regresión

Coeficientes^a

Modelo	Coeficientes no estandarizados		Coeficientes estandarizados	t	Sig.	Estadísticas de colinealidad	
	B	Error estándar	Beta			Tolerancia	VIF
1 (Constante)	3,973	,898		4,422	,001		
ADEC_FINAL	,131	,220	,157	,593	,562	1,000	1,000
2 (Constante)	4,841	,912		5,306	,000		
ADEC_FINAL	,422	,244	,506	1,732	,107	,664	1,507
USA_FINAL	-,557	,270	-,602	-2,062	,060	,664	1,507
3 (Constante)	4,931	,940		5,245	,000		
ADEC_FINAL	,248	,354	,297	,700	,047	,328	3,051
USA_FINAL	-,576	,277	-,622	-2,079	,060	,657	1,522
FIABI_FINAL	,175	,252	,278	,693	,052	,366	2,735
4 (Constante)	3,338	1,684		1,982	,073		
ADEC_FINAL	,045	,393	,053	,113	,012	,260	3,851
USA_FINAL	-,773	,324	-,835	-2,384	,036	,469	2,131
FIABI_FINAL	,147	,251	,233	,584	,057	,362	2,762
SEGURI	,887	,782	,517	1,134	,028	,277	3,614
5 (Constante)	1,378	3,063		,450	,062		
ADEC_FINAL	-,219	,460	-,262	-,476	,044	,197	1,075
USA_FINAL	1,832	,339	,899	2,454	,034	,445	2,247
FIABI_FINAL	,142	,289	,067	,146	,087	,283	1,536
SEGURI	,239	,861	,664	1,323	,015	,237	1,219
PORTA	,257	,332	,272	,773	,058	,481	2,078

Variable dependiente: SATIS

En la tabla de coeficientes siguientes se observa que nuestro modelo de regresión es:

$$Y = C_0 + C_1X_1 + C_2X_2 + C_3X_3 + E$$

$$Y = 1.378 - 0.219X_1 + 1.832X_2 + 0.142X_3 + 0.239X_4 + 0.257X_5 + E$$

De los coeficientes obtenidos concluimos que las variables Adecuación Funcional, Fiabilidad, Seguridad y Portabilidad aportan para explicar la varianza de la variable independiente. Mientras que la variable Adecuación Funcional no aportan a la explicación de la variable dependiente porque su coeficiente es muy bajo (-0.219), esto quiere decir que la variable es irrelevante para los usuarios.

De la misma tabla, también podemos observar los valores t y su significancia, que son valores que nos demuestran que tanto podemos generalizar el modelo de predicción a la población, son: $t = 0.47, -2.454, 0.146, 1.323$ y 0.773 . Sin embargo, la significancia para estos mismos coeficientes respectivamente es la siguiente $= 0.044, 0.034, 0.087, 0.015$ y 0.058 ; lo que significa que las variables al no ser sig. < 0.05 se pueden excluir del modelo.

e. Estadísticos de colinealidad

Los estadísticos de colinealidad no tienen significancia, por lo que se tiene que interpretar en base a los valores que se están obteniendo. Se analiza con el valor de FIV (Factor de Varianza Inflada) el cual debe tener valores menores a 10 pero cercanos a 1, para evitar colinealidad entre las variables.

Los valores FIV obtenidos son para las variables Adecuación funcional, Usabilidad, Fiabilidad, Seguridad y Portabilidad son: 1.075, 2.247, 1.536, 1.219, 2.078 respectivamente. Por lo tanto, no hay colinealidad entre las variables independientes (no están en la misma recta).

CAPÍTULO V. RESULTADOS Y DISCUSIÓN

Se logró desarrollar e implementar con éxito una aplicación móvil en plataforma Android, como se demostró en la parte de la construcción. Esta aplicación fue puesta a prueba mediante una encuesta a un equipo de 16 personas de un grupo de empresas especializadas en desarrollar software con metodología scrum, para corroborar que ejecuta la planificación y verificación de historias de usuario alineados a SCRUM.

En este caso se ha alcanzado un coeficiente de Alfa de Cronbach de 0.746, y según las recomendaciones para evaluar a éste, nos indica que la encuesta aplicada es aceptable.

Al evaluar la funcionalidad de la aplicación Android en relación con el cumplimiento de las actividades de planificación y verificación se logró determinar que la adecuación funcional explica el 2.5% aproximadamente en la variable dependiente lo que significa que No influye mucho en el grado de importancia que el usuario le da para poder planificar y verificar historias de usuario.

Al haber realizado pruebas en distintos dispositivos, se comprobó que portabilidad de la aplicación móvil funciona en sistema operativo Android desde la versión Android 4.4 hasta versiones de Android 7.0, pudiendo escalar a versiones posteriores. Esto permite la instalación en distintos dispositivos como celulares y tablets que cumplan con las características básicas. Pero que sólo influye en un 3.6% aproximadamente en la variable dependiente lo cual significa que en la opinión del usuario no es muy relevante para planificar y verificar historias de usuario.

El uso de la aplicación durante las pruebas a distintos usuarios que trabajan proyectos usando la metodología scrum se logró comprobar que los usuarios se familiarizaron rápidamente con las interfaces de la aplicación pudiendo usarla y entenderla de manera fácil. Es decir, la usabilidad tiene más influencia 24% aproximadamente, siendo la que más influye en la variable dependiente lo cual significa que los usuarios sienten una ventaja para planificar y verificar historias de usuario con el aplicativo.

Al hacer una evaluación de la trazabilidad de las acciones realizadas como creación de los registros de proyectos, backlog, historias y usuario, criterios de aceptación, Sprint, tareas y checklist, se logró comprobar que se realizaron secuencialmente alineados a la planificación y verificación de historias de usuarios en los proyectos trabajados bajo la metodología scrum.

CONCLUSIONES Y RECOMENDACIONES

- Una aplicación móvil en plataforma Android permite desarrollar las actividades de planificación y verificación de historias de usuario en proyectos de desarrollo de software, bajo la metodología SCRUM, como se ha demostrado en la presente investigación.
- Se evaluó la funcionalidad de la aplicación Android con relación al cumplimiento de las actividades de planificación y verificación, lográndose probar con éxito en los dos casos detallados, siendo considerada no tan relevante para los usuarios.
- La portabilidad de la aplicación Android para dispositivos móviles con distintas versiones del sistema operativo Android, se pudo comprobar que funciona en con versiones mayores a Android 4.0 y es adaptable a celulares y tablets sin generar inconvenientes.
- Según los resultados obtenidos podemos aseverar que la usabilidad de la aplicación es fácil de entender por los usuarios que desarrollan proyectos de software, siendo fácil de usar debido a sus interfaces amigables alineadas a la planificación y verificación de historias de usuarios según la metodología Scrum.
- Al construir la aplicación móvil en Android se realizó una serie de procedimientos que permitieron evaluar la trazabilidad de este proyecto, lográndose desarrollar con éxito actividades como: registros de proyectos, backlog, historias de usuario, criterios de aceptación, Sprint, tareas y checklist que es la verificación de que las historias se hayan desarrollado correctamente, basado en la planificación y verificación de historias de usuario según SCRUM.
- La presente aplicación móvil muestra que la orientación a la innovación tecnológica es importante, ya que se puede usar como una herramienta para el desarrollo de proyectos de software mejorando la planificación y verificación de historias de usuario, permitiendo dirigir sus operaciones apoyándose en herramientas tecnológicas para mejorar sus procesos.
- La herramienta permite a la organización ahorrar una gran cantidad de tiempo al investigar las diferentes metodologías, centrando sus esfuerzos solo en aplicar una metodología concreta que es Scrum y aumentando la probabilidad de éxito al momento de planificar y verificar historias de usuario.

- Podemos aseverar que se ha evidenciado la importancia que tiene el Product Owner en la definición de los requerimientos iniciales para poder generar las historias de usuarios, pues de ellos depende la conformidad y satisfacción de los usuarios del sistema. Además de promover el trabajo en equipo siguiendo los Sprints y tareas como parte esencial para generar un producto de software, sin dejar de lado al Scrum master que puede visualizar y monitorear todo el proceso de desarrollo evitando una documentación excesiva, y sólo documentando lo esencial.

RECOMENDACIONES:

- Continuar con el estudio de la metodología Scrum, ya que es una metodología que va creciendo y se va extendiendo en nuestro país, por las empresas de desarrollo de software de la región ya han empezado a usarla como base para la creación de proyectos de software.
- Cabe mencionar que la aplicación sólo abarca la parte de planificación y verificación de historias de usuario, siendo esta una pequeña parte de toda la metodología expuesta, por lo que se recomienda plantear una nueva versión que abarque otros procesos de scrum que no han sido considerados.
- Se puede usar la aplicación siempre y cuando el personal esté capacitado y conozca los procesos de planificación y verificación de historias de usuario según lo definido por scrum.
- El aplicativo solo puede ser instalado en las versiones de Android señaladas anteriormente, es decir, versiones de Android mayores a 4.0 por lo cual se recomienda un dispositivo móvil con esa característica, sin importar si es un celular o una Tablet.
- Para el correcto funcionamiento de la aplicación se recomienda tener una conexión estable a internet para poder manipular y guardar la información sobre los proyectos a desarrollar.
- Continuar con el desarrollo de esta aplicación para abarcar otros procesos de Scrum que no han sido considerados en este proyecto para obtener una versión más completa para el desarrollo de proyectos de desarrollo de software usando la metodología Scrum.

REFERENCIAS BIBLIOGRÁFICAS

- Alaimo, D. M. (2013). *Proyectos ágiles con Scrum : flexibilidad, aprendizaje,*. Buenos Aires: Kleer.
- Almanza Jiménez, R., & Vargas Hernández, J. G. (2015). *Las Competencias Profesionales y su relación con la empleabilidad de los Ingenieros en Gestión Empresarial egresados del ITLAC*. México: Revista gestión de las personas y tecnología.
- Arrivi, O. (3 de Octubre de 2010). *El blog de Oscar Arrivi* . Obtenido de <http://theartoftheleftfoot.blogspot.pe/2010/10/el-patron-modelo-vista-presentador-mvp.html>
- Axosoft. (Agosto de 2016). *Axosoft.com*. Obtenido de <http://www.scrumhub.com/scrum-guide/product-backlog/>
- Cohn, M. (Febrero de 2009). *User Stories Applied for Agile Software Development*. Estados Unidos, Estados Unidos: Pearson Education, Inc.
- Corral, L., Sillitti, A., & Giancarlo, S. (2013). Agile Software Development Processes for Mobile Systems: Accomplishment, Evidence and Evolution. *International Conference on Mobile Web Information Systems*. Paphos, Chipre.
- Dapena, A., García-Naya, J. A., Castro, P. M., & Pan, C. (2010). Aplicación web para evaluación y seguimiento del rendimiento de asignaturas y titulaciones universitarias. *Formación e Innovación Educativa Universitaria*. Vol. 3, 152-165.
- Giraldo Rivera, D. A., & Martínez Pérez, N. V. (08 de Mayo de 2015). *APLICATIVO (PROTOTIPO) PARA LA ADMINISTRACIÓN Y ESTANDARIZACIÓN DE AUDITORÍAS INTEGRALES INTERNAS EN LA PRESELECCIÓN, EVALUACIÓN Y SEGUIMIENTO DE CONTRATISTAS*. Bogotá, Bogotá, Colombia.
- González Mérida, D. (Octubre de 2012). Desarrollo de una Aplicación Móvil: Caso Univerdiada 2012. Veracruz, México.
- Instituto Nacional de Estadística e Informática. (Diciembre de 2002). *Actualización del impacto de las tecnologías de información y comunicación en el Perú*. Lima, Lima.
- ISO 25010. (01 de 03 de 2011). *ISO-International Organization for Standardization*. Obtenido de <https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en>
- Ken, S., & Jeff, S. (2013). *La Guía de Scrum*.
- Kennet, R. (2012). *Essential Scrum*. Michigan: Pearson Education, Inc.
- Miguel, P. V. (Noviembre de 2011). Analisis de Plataformas Populares de Desarrollo de Aplicaciones para Dispositivos Móviles. Guatemala.
- Mike, C. (Octubre de 2005). *Agile Estimating and Planning*. Pearson Education, Inc.
- Navarro Cadavid, A., Fernández Martínez, J. D., & Morales Vélez, J. (20 de 09 de 2013). Revisión de metodologías ágiles para el desarrollo de software. Universidad Icesi.

- Ponce Mendoza, U., Madrid Monteverde, J. D., Garcia Gorrostieta, J. M., & Juárez de Haro, A. J. (Abril de 2015). *Análisis Comparativo de la Metodología de Prototipado Rápido de Aplicaciones (RAP) para Desarrollo de SW en Móviles*. México.
- Pressman, R. S. (2002). *Ingeniería del Software. Un enfoque práctico*. México: McGraw-Hill.
- Pressman, R. S. (2005). *Ingeniería de Software: Un enfoque practico 6ta edición*. México: McGRAW-HILL INTERAMERICANA EDITORES, S.A.
- Pressman, R. S. (2010). *Ingeniería de Software: Un enfoque práctico 7ma edición*. México: McGRAW-HILL INTERAMERICANA EDITORES, S.A.
- Scarone, C. A. (Abril de 2005). *La innovación en la empresa: la orientación al mercado como factor de éxito en el proceso de innovación en producto*.
- Spataru, A. C. (2010). *Agile Development Methods*. Edimburgo, Edimburgo, Escocia.
- Spataru, A. C. (2010). *Agile Development Methods* . Edimburgo, Escocia: Universidad de Edimburgo, Escuela de Informática.
- Tomás Gironés, J. (2012). *El gran Libro de Android*. México: Alfaomega Grupo Editos.
- Yáñez Moreno, V., Ponce Mendoza, U., & Soto Bernal, R. A. (01 de Setiembre de 2014). *Propuesta Metodológica para Desarrollo de Aplicaciones Móviles para. Propuesta Metodológica para Desarrollo de Aplicaciones Móviles para dispositivos Android*. Villahermosa, Tabasco, México: Academia Journals.

ANEXOS

Anexo 01: Formato de Encuestas.**Encuesta**

Se le mostrarán las siguientes preguntas con respecto al desarrollo de los proyectos realizados responda de acuerdo con su experiencia usando nuestra aplicación como herramienta de apoyo para la planificación y verificación de historias de usuario.

Nombres: _____

Empresa en la que labora: _____

Rol que desempeña: _____

Coloque el número que corresponda a su nivel de apreciación siendo
1-Muy malo, 2-Malo, 3-Regular, 4-Bueno, 5-Muy bueno

Formato de encuesta – Adecuación Funcional

Coloque un valor a cada pregunta, según la siguiente escala: **Respuesta**

- d) ¿Cómo califica la capacidad de la aplicación para cumplir con los objetivos de planificación de Historias de usuario según Scrum?
 - e) ¿Cómo califica la capacidad de la aplicación para cumplir con los objetivos en cuanto a verificación de historias de Usuario según Scrum?
 - f) ¿Cómo califica la capacidad del sistema para generar resultados alineados a Scrum?
-

Formato de encuesta – Usabilidad

Coloque un valor a cada pregunta, según la siguiente escala: **Respuesta**

- d) ¿Cómo califica la capacidad de entender el funcionamiento la aplicación?
 - e) ¿Cómo califica las características de la aplicación para ser manipulada y controlada?
 - f) ¿Cómo califica la capacidad de aprender a usar el sistema?
-

Formato de encuesta – Fiabilidad

Coloque un valor a cada pregunta, según la siguiente escala:	Respuesta
---	------------------

- c) ¿Cómo califica el acceso a la aplicación cuando requiere registrar u operar información?
 - d) ¿Cómo califica el comportamiento de la aplicación para recuperar datos en caso de interrupción y fallos?
-

Formato de encuesta – Seguridad

Coloque un valor a cada pregunta, según la siguiente escala:	Respuesta
---	------------------

- c) ¿Cómo califica la forma de acceder a los recursos de la aplicación?
 - d) ¿Cómo califica la privacidad de la información ingresada por parte los usuarios en la aplicación?
-

Formato de encuesta - Portabilidad

Coloque un valor a cada pregunta, según la siguiente escala:	Respuesta
---	------------------

- d) ¿Cómo califica la capacidad de adaptabilidad de la aplicación a la versión de Android de su dispositivo?
 - e) ¿Cómo califica la forma y el tiempo que se invirtió para la instalación del sistema en su dispositivo móvil?
 - f) ¿Cómo califica la forma y el tiempo que se invirtió para la desinstalación del sistema en su dispositivo móvil?
-

Formato de encuesta - Satisfacción

En su experiencia más reciente con SCRUM. Piense acerca de las características y beneficios de la Aplicación móvil

Coloque un valor a cada pregunta, según la siguiente escala:	Respuesta
---	------------------

- e) ¿Cómo fue la calidad de servicio que brinda la aplicación con respecto al rol que Ud. Desempeña?
 - f) ¿Qué tan satisfecho está usted con esta aplicación?
 - g) ¿Cree que la aplicación es una buena opción para mejorar los procesos en SCRUM?
 - h) ¿Cree que la aplicación de alinea al proceso de planificación y verificación de Historias de Usuario?
-

Anexo 02: Documento de recolección de datos sobre empresas de desarrollo

N° A - 0014049

 sunarp <small>Superintendencia Nacional de los Registros Públicos</small>	SOLICITUD DE ACCESO A LA INFORMACIÓN PÚBLICA <small>(Texto único ordenado de la ley 27806)</small> <small>Ley de Transparencia y Acceso a la Información Pública aprobada por Decreto Supremo N° 043-2003 PCM</small>	N° de Registro
	FORMULARIO	

I	Funcionario Responsable de Entregar la Información

II	Datos del Solicitante			
Apellidos y Nombres / Razón Social		Documento de Identidad: D.N.I. / L.M. / C.E. / Otro		
LIVAQUE DELGADO KETTY ADALI		DNI : 47505419		
Domicilio				
Av. / Calle / Jr. / Psj.	N° / Dpto / Int.	Distrito	Urbanización	
LUIS CASTILLA	164	CHICLAYO	LA PRIMAVERA	
Provincia	Departamento	Correo Electrónico	Teléfono	
CHICLAYO	LAMBAYEQUE	ketty-adali4@hotmail.com	950929772	

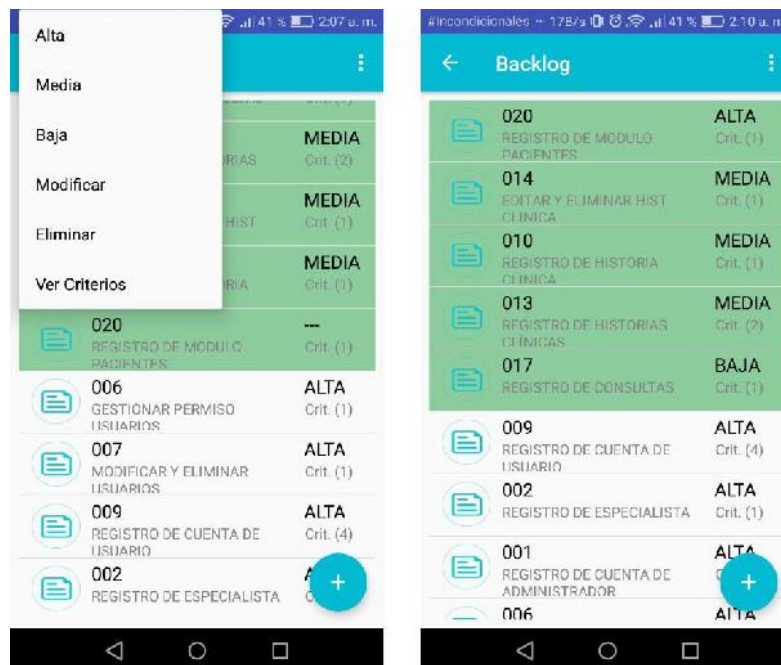
III	Información Solicitada
LISTA DE EMPRESAS QUE DESARROLLAN SOFTWARE EN LA PROVINCIA DE CHICLAYO	

IV	Dependencia de la cual se requiere la información
ARCHIVO REGISTRAL	

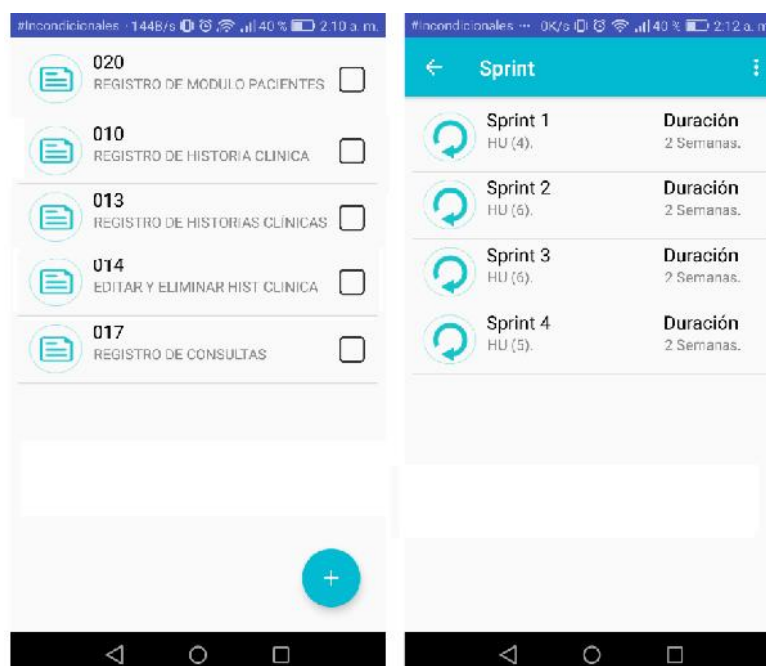
V	Forma de Entrega de la información (marcar con una x)		
Fotocopia	(x)	Diskette	Disco Compacto (CD)
Unidad de Video-Casette			
Apellidos y Nombres		Fecha y hora de recepción:	
Firma			
Observaciones: SUNARP NO CUENTA CON ESTE TIPO DE INFORMACIÓN POR LO CUAL NO PROCEDE LA SOLICITUD.			

Anexo 03: Historias de usuario no completadas de acuerdo con el Product Owner.

Se le asigna una prioridad **ALTA** para que se pueda ubicar en la parte superior y sea desarrollada primero, ya que es una historia de usuario que no cumple con lo solicitado por el Product Owner.



Una vez priorizada se le asignará al Sprint que le corresponde, en este caso la Historia de Usuario con código 020 estará en el Sprint 3 para ser desarrollada con urgencia.



Anexo 04: Reportes de la Aplicación

Product Backlog.

Código	Título	Historia de Usuario	Criterio de Aceptación	Prioridad
001	REGISTRO CUENTA ADMINISTRADOR	Yo como administrador necesito poder registrarme en el sistema para iniciar el control de procesos en mi negocio	Registrar datos como nombres, apellidos, fecha de ingreso, DNI, tipos de usuarios	ALTA
			Debe tener usuario de administrador y una contraseña	
			Tener todos los permisos habilitados en el sistema.	
002	GESTIONAR PERMISOS PARA USUARIOS	Yo como administrador necesito restricciones en los permisos a vendedores para controlar los accesos al sistema	El sistema debe permitir el registro de vendedor encargado de la caja	ALTA
			El vendedor sólo puede acceder a clientes, productos, pedidos y ventas.	
003	REGISTRO DE MODULO DE COMIDAS	Yo como administrador necesito registrar mis productos para tener una lista del menú	Registrar código, nombre y precio del producto	ALTA
			Listar productos	
004	MODIFICAR Y ELIMINAR PRODUCTOS	Yo como administrador necesito modificar y eliminar mis productos para tener una lista de menú actualizada	Permitir eliminar un producto de la lista de productos	ALTA
			Actualizar los productos con los nuevos datos ingresados	
			Permitir editar el nombre, precio y descripción del producto	
005	REGISTRO DE CIERRE DE CAJA	Yo como administrador necesito crear y registrar el cierre de caja para tener un mejor control de las ventas diarias	Crear registros de hora y fecha de apertura y cierre de caja.	ALTA
006	MODIFICAR Y ELIMINAR CIERRE DE CAJA	Yo como administrador necesito modificar el cierre de caja para tener un mejor control de las ventas diarias	Cargar los registros de hora y fecha de cierre de caja.	ALTA
007	REGISTRO DE CUENTA DE VENDEDOR	Yo como vendedor necesito tener una cuenta de usuario para registrar pedidos y ventas	Poder tener acceso al sistema para registrar	ALTA
			Debe registrar los datos del vendedor como nombres, apellidos, fecha de ingreso, DNI, tipo de usuario	
			Debe tener un nombre de usuario y contraseña	
			Sólo debe tener acceso a clientes, productos, pedidos y ventas.	

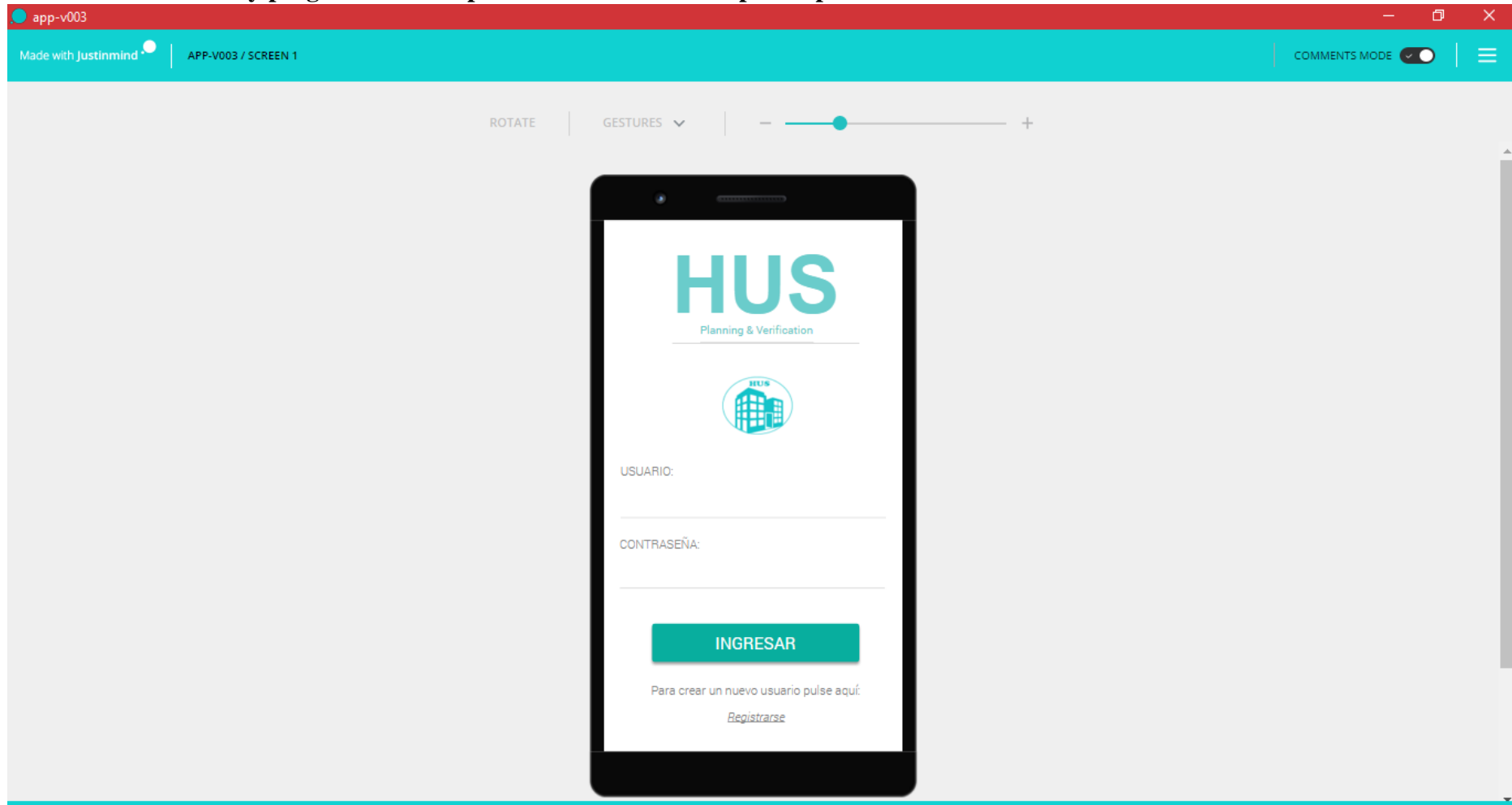
Listado de Avance de Historias.

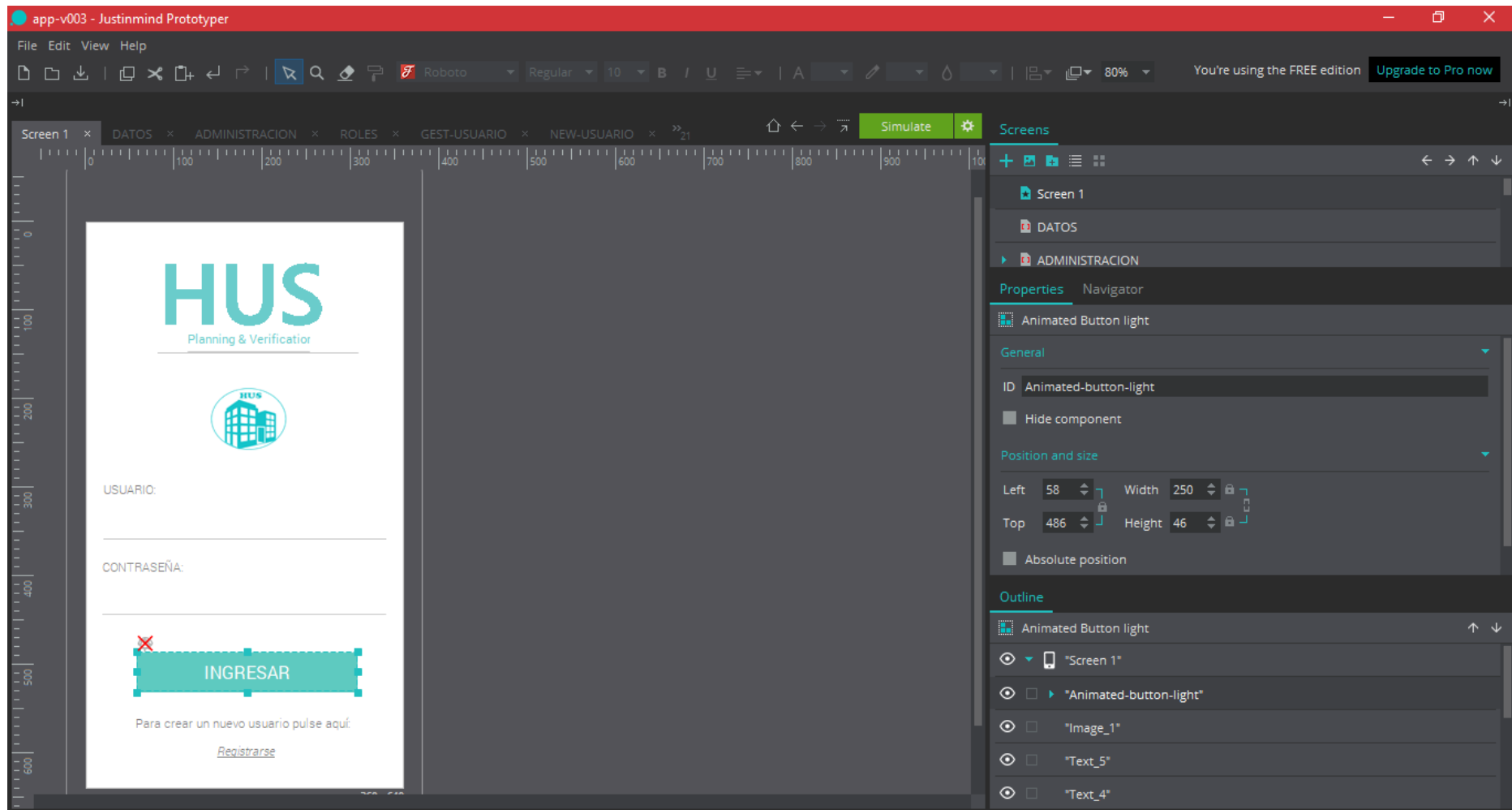
N° Sprint	Código	Nombre HU	Tarea	Estado	% Avance
1	003	REGISTRO DE MODULO DE COMIDAS	Generar lista de los productos	T	100.0
			Diseño e implementación del módulo de producto	T	
			Registro de código, nombre y precio	T	
			Crear opciones de búsqueda	T	
	007	REGISTRO DE CUENTA DE VENDEDOR	Creación del módulo vendedores	T	100.0
			Registro e inicio de sesión para vendedores	T	
			Creación de acceso a clientes, productos, pedidos y ventas	T	
	004	MODIFICAR Y ELIMINAR PRODUCTOS	Actualización de listas de todos los productos	T	100.0
			Modificar y eliminar datos como nombre, precio y descripción	T	
	002	GESTIONAR PERMISOS PARA USUARIOS	Creación y restricción de permisos para los usuarios administrador y vendedor	T	100.0
	001	REGISTRO CUENTA ADMINISTRADOR	Creación e implementación de la base de datos	T	100.0
			Creación del login para registro de administrador	T	
			Restringir permisos para usuarios - vendedores	T	

Product Backlog.

Código	Título	Descripción	Se Cumplió		Observaciones
			Si	No	
001	REGISTRO CUENTA ADMINISTRADOR	Debe tener usuario de administrador y una contraseña	X		
		Registrar datos como nombres, apellidos, fecha de ingreso, DNI, tipos de usuarios	X		
		Tener todos los permisos habilitados en el sistema.	X		
002	GESTIONAR PERMISOS PARA USUARIOS	El sistema debe permitir el registro de vendedor encargado de la caja		X	la ventana del registro debe ser un poco más grande
		El vendedor sólo puede acceder a clientes, productos, pedidos y ventas.	X		
003	REGISTRO DE MODULO DE COMIDAS	Listar productos	X		
		Registrar código, nombre y precio del producto	X		
004	MODIFICAR Y ELIMINAR PRODUCTOS	Actualizar los productos con los nuevos datos ingresados	X		
		Permitir editar el nombre, precio y descripción del producto	X		
		Permitir eliminar un producto de la lista de productos	X		
005	REGISTRO DE CIERRE DE CAJA	Crear registros de hora y fecha de apertura	X		
006	MODIFICAR Y ELIMINAR CIERRE DE CAJA	Cargar los registros de hora y fecha de cierre de caja	X		
007	REGISTRO DE CUENTA DE VENDEDOR	Debe registrar los datos del vendedor como nombres, apellidos, fecha de ingreso, DNI, tipo de usuario	X		
		Debe tener un nombre de usuario y contraseña	X		
		Poder tener acceso al sistema para registrar	X		
		Sólo debe tener acceso a clientes, productos, pedidos y ventas.	X		

Anexo 05: Simulador y programa usados para el desarrollo de los prototipos





Anexo 06: Primer diagrama de secuencias de prototipos

