

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA



PROYECTO DE GRADO

**SISTEMA WEB DE SEGUIMIENTO Y LOCALIZACIÓN DE
PERSONAL DE SERVICIO DE DELIVERY**

Proyecto de Grado para obtener el Título de Licenciatura en Informática
Mención Ingeniería de Sistemas Informáticos

POR: JUAN CARLOS CHOQUE ALAVI
TUTOR METODOLÓGICO: PH.D. JAVIER REYES PACHECO
ASESOR: P.PH.D. YOHONI CUENCA SARZURI

LA PAZ – BOLIVIA

2021



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria

A mis padres Carlota Alavi Condori, quien con su fe me enseñó a luchar en la adversidad, Carlos Choque Ticona, quien me cuida en la ausencia. A mis hermanas Leonor, Norah Rossmery, y Sofía por su apoyo, comprensión y consejos en todo momento. A mis sobrinos William y Andrea por su alegría que emanan y quienes son mis hermanitos pequeños y a mi tío Antonio por sus consejos.

AGRADECIMIENTO

Agradecer a mi tutor Ph.D. Javier Reyes Pacheco por dar los consejos adecuados para culminar con esta meta.

Agradecer a mi asesor P.Ph.D. Yohoni Cuenca Sarzuri por guiarme y brindarme su colaboración para terminar este proyecto.

Agradecer a los docentes de la carrera de Informática por los conocimientos transmitidos durante el ciclo de formación universitaria.

Agradecer al Bolivia Tech Hub, por la acogida y apoyo al permitirme llevar a cabo el presente proyecto.

Agradecer a mi familia quienes siempre estarán presentes con su guía, fortaleza, y su amor.

alavjc@gmail.com

RESUMEN

El presente trabajo tiene el propósito de localizar a las personas que se encuentren en el área urbana, utilizando como medio una herramienta que en la actualidad es muy popular e indispensable como es el caso de los dispositivos móviles comúnmente llamados smartphome, por medio el GPS e internet se capturan y envían la latitud y longitud del dispositivo, estos son desplegados en un mapa que forma parte del Sistema Web de Seguimiento y Localización de Personal de Servicio de Delivery

El proyecto presenta una variedad de funcionalidades, entre ellos se destaca la captura de la georreferenciación, es decir la latitud y longitud del smarphone por medio de una aplicación móvil operada por el personal que recolecta y entrega paquetes; el despliegue y actualización constante de la georreferenciación capturada de cada dispositivo, será representada por un punto o marcador en un mapa, según los usuarios activos que usen la aplicación. Se cuenta también con avisos y/o notificaciones al personal de recolección y entrega de paquetes, desde la aplicación web a la aplicación móvil; asimismo el sistema web cuenta con notificaciones para informar tanto al administrador y cliente sobre los estados de cada proceso atendido.

En el capítulo 1, identificamos el problema general y los problemas específicos para posteriormente formular los objetivos a cumplir en la implementación del proyecto. El capítulo 2 sienta las bases teóricas de la investigación, metodologías UWE y Scrum. Pasando al capítulo 3, marco aplicativo, se tiene el desarrollo e implementación del proyecto, el análisis de requerimientos, modelado de la base de datos, historias de usuario, definición de tareas a llevar a cabo según Scrum y aplicación de UWE. Ya en el capítulo 4 tenemos el análisis de calidad del software según ISO 9126 y el análisis de costo beneficio. Para finalizar se tienen conclusiones en el capítulo 5 donde describimos el cumplimiento de objetivos y recomendando ideas que podrían a aplicarse posteriormente en el presente proyecto.

Palabras claves: Georreferenciación, Sistema de Posicionamiento Global, Smartphome

ABSTRACT

The present work has the purpose of locating people who are in the urban area, using as a means a tool that is currently very popular and indispensable as is the case of mobile devices commonly called smartphones, by means of GPS and Internet is captured and sent the latitude and longitude of the device, these are displayed on a map that is part of the Web System for Tracking and Location of Delivery Service Personnel

The project presents a variety of functionalities, among them the capture of georeferencing stands out, that is, the latitude and longitude of the smartphone through a mobile application operated by the personnel that collects and delivers packages; the deployment and constant updating of the georeferencing captured from each device will be represented by a point or marker on a map, according to the active users who use the application. There are also notices and / or notifications to the personnel of collection and delivery of packages, from the web application to the mobile application; Likewise, the web system has notifications to inform both the administrator and the client about the status of each process attended.

In Chapter 1, we identify the general problem and the specific problems to later formulate the objectives to be met in the implementation of the project. Chapter 2 lays the theoretical foundations for research, UWE and Scrum methodologies. Moving on to chapter 3, application framework, there is the development and implementation of the project, requirements analysis, database modeling, user stories, definition of tasks to be carried out according to Scrum and UWE application. Already in chapter 4 we have the software quality analysis according to ISO 9126 and the cost benefit analysis. Finally, there are conclusions in chapter 5 where we describe the fulfillment of objectives and recommend ideas that could be applied later in this project.

Keywords: Georeferencing, Global Positioning System, Smartphone

CONTENIDO

| | |
|--|----|
| CAPÍTULO 1 | 2 |
| MARCO INTRODUCTORIO..... | 2 |
| 1. Introducción..... | 2 |
| 1.2. Antecedentes..... | 3 |
| 1.2.1. Antecedentes Institucionales | 3 |
| 1.2.2. Antecedentes de Proyectos Similares | 4 |
| 1.3. Planteamiento del Problema | 5 |
| 1.3.1. Problema Central | 6 |
| 1.3.2. Problemas Secundarios..... | 6 |
| 1.4. Definición de Objetivos..... | 6 |
| 1.4.1. Objetivo General | 6 |
| 1.4.2. Objetivos Específicos | 7 |
| 1.5. Justificación..... | 7 |
| 1.5.1. Justificación Económica | 7 |
| 1.5.2. Justificación Social..... | 7 |
| 1.5.3. Justificación Tecnológica | 7 |
| 1.6. Alcances y Limites | 8 |
| 1.6.1. Alcances | 8 |
| 1.6.2. Limites | 8 |
| 1.7. Aportes | 9 |
| 1.8. Metodología..... | 9 |
| CAPÍTULO 2 | 10 |
| MARCO TEÓRICO | 10 |
| 2.1. Administración de un Delivery | 10 |
| 2.2. Geolocalización | 10 |
| 2.2.1. Sistema de posicionamiento global | 11 |
| 2.3. Diseño de Software..... | 12 |
| 2.3.1. Ingeniería de Software..... | 12 |
| 2.3.2. Ingeniería de Requerimientos..... | 12 |
| 2.3.3. Análisis de requerimientos | 13 |

| | |
|---|----|
| 2.3.4. UML Web Engineering - UWE..... | 13 |
| 2.3.4.1. Modelado de UWE..... | 14 |
| 2.3.4.2. Fases de la UWE. | 15 |
| 2.4. Desarrollo de Software | 16 |
| 2.4.1. Metodología ágil..... | 16 |
| 2.4.2. Metodología Ágil Scrum | 17 |
| 2.4.2.1. Principios de agilidad | 17 |
| 2.4.2.2. Elementos de Scrum | 18 |
| 2.4.2.3. Proceso trabajo de Scrum | 20 |
| 2.5. Herramientas de Desarrollo | 22 |
| 2.5.1. Framework Codeigniter..... | 22 |
| 2.5.2. MySql | 23 |
| 2.5.3. Android..... | 23 |
| 2.5.4. Google Maps Platform | 26 |
| 2.6. Métricas de Calidad..... | 27 |
| 2.6.1 ISO/IEC 9126 | 27 |
| 2.6. Costo y/o Beneficio | 31 |
| 2.6.1. Método de COCOMO | 31 |
| 2.7. Seguridad..... | 32 |
| 2.7.1 Autenticación de Usuarios..... | 33 |
| 2.7.1.1 Autenticación..... | 33 |
| 2.7.1.2 Autorización | 33 |
| CAPÍTULO III | 35 |
| MARCO APLICATIVO | 35 |
| 3.1. Elaboración del Product-Backlog..... | 36 |
| 3.1.1. Roles de usuarios | 36 |
| 3.1.2. Requerimientos..... | 36 |
| 3.1.3. Historias de usuario | 39 |
| 3.2. Primera Iteración (Sprint1)..... | 49 |
| 3.3. Segunda Iteración (Sprint 2)..... | 60 |
| 3.4. Tercera Iteración (Sprint 3) | 70 |

| | |
|---|-----|
| CAPÍTULO IV | 78 |
| CALIDAD DE SOFTWARE Y COSTO BENEFICIO | 78 |
| 4.1. Calidad de Software | 78 |
| 4.1.1. Norma de Calidad..... | 78 |
| 4.1.2. Métricas de Calidad | 78 |
| 4.1.2.1. Funcionalidad | 79 |
| 4.1.2.3. Usabilidad..... | 84 |
| 4.1.2.7. Análisis de Resultados..... | 90 |
| 4.2. Seguridad..... | 91 |
| 4.2.1. Seguridad de Software..... | 91 |
| 4.2.2. Seguridad de Hardware | 91 |
| 4.2.3. Seguridad de Red..... | 92 |
| 4.3. Análisis de Costo y Beneficio | 92 |
| 4.3.1. Análisis de Costos | 93 |
| 4.3.1.1. Costo del Software Desarrollado..... | 93 |
| 4.3.1.2. Costo de Despliegue del Proyecto..... | 97 |
| 4.3.1.3. Costo de Elaboración del Proyecto..... | 97 |
| 4.3.1.4. Costo Total | 98 |
| 4.3.2. Análisis de Beneficios | 98 |
| 4.3.2.1. Valor Neto Actual..... | 98 |
| 4.3.2.2. Tasa Interna de Retorno..... | 100 |
| 4.3.3. Costo / Beneficio | 102 |
| CAPITULO V | 103 |
| CONCLUSIONES Y RECOMENDACIONES..... | 103 |
| 5.1 Conclusiones..... | 103 |
| 5.2. Recomendaciones | 104 |
| BIBLIOGRAFÍA | 106 |

ÍNDICE DE FIGURAS

| | |
|---|---|
| Figura 1.1 Organigrama Institucional..... | 4 |
|---|---|

| | |
|---|----|
| Figura 2.1 Descripción del proceso de Scrum..... | 20 |
| Figura 3.1 Diagrama entidad relación | 48 |
| Figura 3.2 Diagrama de caso de uso – Administración de usuarios..... | 50 |
| Figura 3.3 Diagrama de caso de uso – Registro de cliente..... | 50 |
| Figura 3.4 Diagrama de caso de uso – Autenticación de usuario..... | 51 |
| Figura 3.5 Diagrama de navegación – Administración de usuarios..... | 51 |
| Figura 3.6 Diagrama de presentación – Administración de usuarios..... | 52 |
| Figura 3.7 Autenticación de usuarios | 52 |
| Figura 3.8 Registro para clientes y/o empresas | 53 |
| Figura 3.9 Opciones para la administración de usuarios..... | 53 |
| Figura 3.10 Gestión de administradores..... | 54 |
| Figura 3.11 Gestión de administradores – Registrar nuevo administrador | 55 |
| Figura 3.12 Gestión de administradores – Modificar usuario administrador..... | 55 |
| Figura 3.13 Gestión de administradores – Inhabilitar usuario administrador | 56 |
| Figura 3.14 Gestión de usuarios repartidores..... | 56 |
| Figura 3.15 Gestión de usuarios repartidores – Detalles de personal repartidor..... | 57 |
| Figura 3.16 Gestión de empresas Fuente: Elaboración propia..... | 57 |
| Figura 3.17 Gestión de clientes | 58 |
| Figura 3.18 Diagrama de caso de uso – Solicitud de servicio..... | 62 |
| Figura 3.19 Diagrama de caso de uso – Localización de personal..... | 62 |
| Figura 3.20 Diagrama de caso de uso – Cambios de estado por el repartidor..... | 63 |
| Figura 3.21 Diagrama de navegación – Solicitud de petición de servicio | 63 |
| Figura 3.22 Diagrama de navegación – Interacción con la aplicación móvil..... | 64 |
| Figura 3.23 Diagrama de presentación – Tablero de seguimiento | 64 |
| Figura 3.24 Solicitud de servicio..... | 65 |
| Figura 3.25 Despliegue de datos del paquete | 65 |
| Figura 3.26 Asignación de paquete al repartidor..... | 66 |
| Figura 3.27 Aplicación móvil y notificación de asignación de paquete..... | 66 |
| Figura 3.28 Aplicación móvil listado y detalle del paquete | 67 |
| Figura 3.29 Aplicación móvil punto de recolección y entrega del paquete | 67 |
| Figura 3.30 Diagrama de caso de uso – Notificación de cambio de estado del paquete..... | 71 |

| | |
|---|-----|
| Figura 3.31 Diagrama de caso de uso – Generación de reporte | 71 |
| Figura 3.32 Diagrama de navegación – Cambio de estado del paquete | 72 |
| Figura 3.33 Historial de pedidos Fuente: Elaboración propia | 73 |
| Figura 3.34 Cambio de estado del historial de pedidos | 73 |
| Figura 3.35 Generación de código QR | 74 |
| Figura 3.36 Conclusión de envío de paquete..... | 74 |
| Figura 3.37 Personal de la empresa que recolecta el paquete | 75 |
| Figura 3.38 Punto de recolección de paquete | 75 |
| Figura 3.39 Punto de entrega de paquete..... | 76 |
| Figura 3.40 Cambios de estado desde el dispositivo móvil..... | 76 |
| Figura 3.41 Reporte de paquetes procesados | 77 |
| Figura 3.42 Reporte de paquetes procesados | 77 |
| Figura 4.1 Diagrama de bloque de confiabilidad | 83 |
| Figura 4.2 Calculo por tanteo TIR..... | 101 |

ÍNDICE DE TABLAS

| | |
|---|----|
| Tabla 3.1 Historia de usuario, acceso al sistema | 40 |
| Tabla 3.2 Historia de usuario, registro de usuario | 40 |
| Tabla 3.3 Historia de usuario, modificaciones y eliminación de usuario..... | 41 |
| Tabla 3.4 Historia de usuario, solicitud de servicio de mensajería de paquetes..... | 41 |
| Tabla 3.5 Historia de usuario, confirmación de solicitud de servicio al cliente..... | 42 |
| Tabla 3.6 Historia de usuario, solicitud aceptada..... | 42 |
| Tabla 3.7 Historia de usuario, localización y estados del repetidor en el mapa..... | 43 |
| Tabla 3.8 Historia de usuario, aceptación de solicitud de transporte de paquete | 44 |
| Tabla 3.9 Historia de usuario, seguimiento del paquete..... | 44 |
| Tabla 3.10 Historia de usuario, acceso al sistema por el repartidor | 45 |
| Tabla 3.11 Historia de usuario, envío de locación del repartidor..... | 45 |
| Tabla 3.12 Historia de usuario, notificación de envío de paquete al repartidor | 46 |
| Tabla 3.13 Historia de usuario, cambios de estado del paquete | 46 |
| Tabla 3.14 Historia de usuario, generación de reportes | 47 |
| Tabla 3.15 Detalle de tareas a realizar en el Sprint 1 | 49 |

| | |
|--|-----|
| Tabla 3.16 Prueba de registro de usuarios..... | 58 |
| Tabla 3.17 Prueba de modificación de usuarios..... | 59 |
| Tabla 3.18 Prueba de autenticación de usuarios..... | 59 |
| Tabla 3.19 Detalle de tareas a realizar en el Sprint 2 | 60 |
| Tabla 3.20 Prueba de solicitud de servicio del cliente | 68 |
| Tabla 3.21 Prueba de asignación de transporte de paquete | 68 |
| Tabla 3.22 Notificación de aviso..... | 69 |
| Tabla 3.23 Lista y detalle del paquete | 69 |
| Tabla 3.24 Detalle de tareas a realizar en el Sprint 3 | 70 |
| Tabla 4.1 Conteo de elementos de representación del sistema..... | 79 |
| Tabla 4.2 Métricas de punto función | 80 |
| Tabla 4.3 Componentes del factor de complejidad técnica | 81 |
| Tabla 4.4 Calculo de confiabilidad..... | 83 |
| Tabla 4.5 Evaluación de usabilidad..... | 85 |
| Tabla 4.6 Escala de evaluación..... | 85 |
| Tabla 4.7 Conteo de errores antes y después de la entrega | 87 |
| Tabla 4.8 Resultados de métricas de calidad..... | 90 |
| Tabla 4.9 Calculo punto función no ajustado | 93 |
| Tabla 4.10 Calculo de punto función ajustado | 94 |
| Tabla 4.11 Conversión de Puntos Función a KLDC | 95 |
| Tabla 4.12 Coeficiente ab y bb | 96 |
| Tabla 4.13 Costo de elaboración del proyecto | 97 |
| Tabla 4.14 Costo total de proyecto..... | 98 |
| Tabla 4.15 Criterio de interpretación del VAN | 99 |
| Tabla 4.16 Datos para el cálculo del VAN..... | 99 |
| Tabla 4.17 Estimación de ganancias anuales..... | 100 |
| Tabla 4.18 Calculo del VAN | 100 |
| Tabla 4.19 Calculo por tanteo TIR..... | 101 |

CAPÍTULO 1

MARCO INTRODUCTORIO

1. Introducción

La información es uno de los activos más importante en las empresas, debido a la información que cuenten de su empresa pueden llegar a prosperar como decaer si no es bien administrada, en la actualidad se cuenta con sistemas informáticos desarrollados para ayudar clasificar esta información de forma automática presentando resultados de los datos obtenidos de los movimientos que realiza la empresa inversiones, pagos a empleados, pagos de servicios y otros, de esta manera una empresa puede tomar decisiones correctas a futuro.

En la actualidad una empresa puede iniciar como una startup, un negocio con ideas innovadoras, con capacidad de cambio, su operatividad en un inicio es con costos mínimos obteniendo ganancias que crecen exponencialmente y tienen una fuerte relación laboral con la tecnología.

Él BOLIVIA TECH HUB empezó como es un espacio de desarrollo colaborativo de proyectos tecnológicos relacionados a TIC. Generando oportunidades de aprendizaje, desarrollo, ofertas laborales en nuevas áreas de estudio, competencias nacionales e internacionales denominados hackáthones.

Al pasar el tiempo BOLIVIA TECH HUB se fue adquiriendo conocimiento en el desarrollo de startup (emprendimientos) relacionado a otras áreas laborales como en el caso de turismo y comercialización de artículos artesanales.

Ahora se plantea un nuevo emprendimiento en el cual se pretende de contar con una plataforma de logística aplicado a la actividad de Delivery, el presente proyecto propuesto tiene el enfoque de resolver y disminuir la susceptibilidad en el transporte de paquete por medio de delivery planteado en el área urbana, sistematizando el proceso que con lleva realizar tal trabajo, proponiendo de esta manera realizar una administración en la recolección y reparto de paquetes por medio del personal que será gestionado por GPS de dispositivos celulares Smartphone, realizando una oportuna de designación de personal disponible, como también se controla la recepción del paquete por código QR.

1.2. Antecedentes

En cuanto a los antecedentes tenemos Antecedentes Institucionales y Antecedentes de Proyectos Similares.

1.2.1. Antecedentes Institucionales

“BOLIVIA TECH HUB” inició actividades a mediados de noviembre del 2014, inicialmente contaba con un administrador, un administrador de proyectos y tres secretarías. Actualmente se tiene un incremento en el equipo de trabajo.

Su oficina es ubicada en la Zona Sopocachi final Sánchez Lima, Pasaje Fanini Nro. 2687.

MISIÓN INSTITUCIONAL

Ser un espacio de desarrollo colaborativo de proyectos tecnológicos relacionados a TIC.

VISIÓN INSTITUCIONAL

Convertirse en un centro de innovación tecnológicas apoyando al desarrollo de los jóvenes, creando relaciones y oportunidades con otros entes interesados o con necesidades en TIC.

ORGANIGRAMA

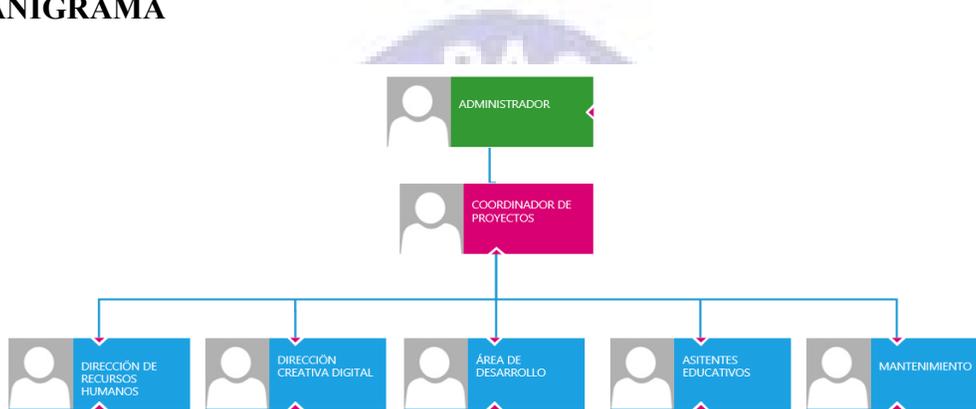


Figura 1.1 Organigrama Institucional
Fuente: (Pressman, 2010)

PROYECTOS INSTITUCIONALES:

- Desarrollan aplicaciones móviles para plataformas Android e iOS.
- Desarrollan aplicaciones web usando distintos lenguajes de programación.
- Desarrollan videojuegos en realidad aumentada, realidad virtual, 2D y 3D.

EMPRENDIMIENTOS:

- Jukumari Advice
- Jukumari Souvenirs

1.2.2. Antecedentes de Proyectos Similares

“Plataforma de Geolocalización de Personas Mediante Dispositivo Móvil” el mencionado trabajo a tiende una problemática de ya varios años, tener que ubicar a una persona en un determinado territorio, plantea el uso de un sistema web que brinda servicios REST el cual será consumido por un dispositivo móvil usando el GPS del dispositivo, de esta manera pretende ayudar a la sociedad en el caso de no poder localizar a un familiar. Guachalla, F. (2015).

“Software móvil de geolocalización para la banca en la ciudad de La Paz”, El objetivo que lleva este proyecto es ubicar por georreferenciación a las entidades bancarias, sucursales y cajeros automáticos según la entidad requerida y cercana según un radio de proximidad desplegando un mapa que detalle la cercanía, se desarrolla un sistema web para la administración de la georreferenciación de las entidades bancarias como también el desarrollo de una app móvil que mostrara al usuario la ubicación de la entidad bancaria cercana, Pérez, A. (2014).

“Sistema web de seguimiento y control de supervisores y obras basado en la geolocalización en dispositivos móviles caos: Agencia Estatal de Vivienda”, en este proyecto se plantea el control y seguimiento de personal de la Agencia Estatal de Vivienda, donde por un sistema Web se podrá desplegar los lugares en construcción, como también realizar un control al personal por medio de dispositivos móviles y GPS, también la captura de datos, fotografías integrando la información necesaria sobre los proyectos de construcción, Mamani. D. (2016).

“Desarrollo de aplicación de geolocalización con teléfonos celulares con sistema operativo Android utilizando el simulador SDK 1.6 y Eclipse 3.7.”, este trabajo plantea el uso de dispositivo móviles con la finalidad de mapear el campus universitario que cuenta con 110 manzanas equivalente a 768.680 metros cuadrados y disponer a los estudiantes como docentes de información precisa de las áreas de estudio y la posición que se encuentra uno, debe tomarse en cuenta que esta trabajo se realizó con celulares de básicos del año 2012 Castellón, G. y Velásquez, R. (2012).

1.3. Planteamiento del Problema

En Bolivia Tech Hub cuenta con emprendimientos propios como es el caso de “Jukumari Advice” proyecto orientado con el fin de promover los lugares turísticos de Latino América especialmente de Bolivia, como también el proyecto “Jukumari Souvenirs“, donde realiza la difusión de artesanías, artículos de decoración, vestimentas artesanales, con la colaboración de socios que comparten su conocimiento sobre el tema.

En la actualidad se tiene empresas que realizan trabajos de transporte de paquetes y/o encomiendas, ya sean departamentales como solo de áreas urbanas o en la ciudad. En el caso de un área urbana es desconocido saber si el personal transporte o repartidor partió con el paquete, se encuentra en medio camino para su entrega o ya concreto la entrega, se tiene la incertidumbre del lugar actual en el que se encuentra el personal.

El Bolivia Tech Hub observando el caso anteriormente presentado tiene como objetivo mejorar estos aspectos de logística presentados en el servicio de delivery de esta manera iniciar un nuevo emprendimiento en beneficio de la empresa.

1.3.1. Problema Central

¿Es posible mejorar el proceso de seguimiento y/o localización de personal en la recepción y entrega de paquetes en un delivery en el área urbana?

1.3.2. Problemas Secundarios

- Desconocimiento preciso del lugar y la persona que se debe contactar para entregar el paquete.
- Se desconoce si el paquete y/o encomienda ya fue entregada o llevada a su destino.
- El cliente no tiene un medio digital para realizar la descripción de su pedido y georreferenciarla.
- No se cuenta con una coordinación sobre qué persona ira a recolectar y entregar el paquete.
- No se cuenta con registros inmediatos para saber la hora en el cual fue recolectado y entregado un paquete

1.4. Definición de Objetivos

1.4.1. Objetivo General

Desarrollar un sistema web de seguimiento y localización de personal para el servicio de delivery.

1.4.2. Objetivos Específicos

- Obtener la georreferenciación de personal repartidor mediante una aplicación móvil.
- Capturar los puntos próximos de recolección y entrega de paquetes, que serán desplegados en un mapa al que tendrá acceso el personal que procesará el paquete.
- Contar con una cuenta de usuario en el cual podrá realizar solicitudes de envío de paquetes.
- Notificar al cliente sobre el personal asignado que procesara su solicitud de envío de paquete.
- Generar códigos QR para cada solicitud de envío de paquete, de esta manera controlar la recolección y entrega del paquete en el sistema.

1.5. Justificación

Las justificaciones son desarrolladas de acuerdo a tres aspectos, económica, social y tecnológica.

1.5.1. Justificación Económica

Un sistema de seguimiento y localización de personal, permitirá controlar las tareas referentes al transporte de paquetes, optimizar tiempo de recepción y entrega de paquetes, y a la vez contar con una plataforma propia y de bajo costo usando software OpenSource.

1.5.2. Justificación Social

Se contará con la posibilidad de notificar a los usuarios del sistema, dando a conocer la información referente al detalle de solicitudes de envíos de paquete al usuario administrador, disponibilidad de servicio para el usuario cliente, y cambios de estado de un paquete en proceso para el administrador como para el cliente. De esta manera se contará con comunicación sencilla entre los usuarios y se facilita información temprana al usuario final que son los clientes.

1.5.3. Justificación Tecnológica

En la actualidad contamos con herramientas tecnológicas a la mano como también acceso a internet, gracias a dispositivos Smartphone podemos acceder a todo tipo de datos sistemas y otros servicios. Se tendrá una comunicación entre el Smartphone y el sistema web intercambiando información mediante servicios REST que dará a conocer su ubicación por GPS y notificando cambios de estados o solicitudes de envíos de paquetes, de esta manera aprovechamos la tecnología que tenemos a nuestro alcance.

1.6. Alcances y Limites

El presente proyecto se desarrolla de acuerdo a los siguientes alcances y límites que presentamos a continuación.

1.6.1. Alcances

Para cumplir nuestro objetivo desarrollaremos los siguientes módulos:

- Se implementará el módulo de registro y procesos para el usuario
- Se implementará el módulo de registro y control de personal
- Se implementará el módulo de despliegue de mapa
 - Capturar el punto de recolección y entrega de paquete
 - Selección del personal cercano el cual atenderá una nueva petición de recolección/envió
- Se implementará una app para dispositivos con sistema operativo Android
 - Para recepción de datos de un nuevo paquete a entregar
 - Para el envío de coordenadas (latitud y longitud) actuales

1.6.2. Limites

- Solo se implementará apps para dispositivos con sistema operativo Android

- El sistema será usado en el área urbana

1.7. Aportes

Se tiene el objetivo de realizar la integración de datos desde un sistema web con una aplicación instalado en un dispositivo móvil, este último es usado como herramienta de localización GPS, el diseño e implementación del sistema web proporcionara a la empresa una fuente temprana de información respecto a conocer la ubicación de sus empleados que están realizando las tareas de recolección y/o entrega de paquetes.

Despliegue de datos de localización capturados desde el dispositivo móvil en el sistema web

Notificaciones en tiempo real, según tareas realizadas: petición de servicio, recolección de paquete, entrega del paquete.

Podrá realizar sus reportes sobre según servicio prestado con rapidez y confiabilidad.

1.8. Metodología

Para alcanzar los objetivos propuestos del presente proyecto se aplicará las siguientes métodos y técnicas.

Método de investigación descriptiva, basada en la recolección de información.

- Entrevistas. Revisión de documentación, recolección de requisitos que debe cumplir el sistema.

La metodología para el diseño y desarrollo de software

- UML Web Engineering - UWE
- Scrum

CAPÍTULO 2

MARCO TEÓRICO

En este capítulo se contemplará los conceptos básicos del proyecto a llevarse a cabo y herramientas que se usaran para su desarrollo y diseño, de esta manera tratamos que el lector asimile y comprenda las bases que conlleva la construcción del software obtenido una mejor óptica del mismo.

2.1. Administración de un Delivery

Un delivery se rige a partir de una función logística de trabajo que tiene por finalidad colocar bienes, servicios, fondos o información directo en el lugar de consumo o cliente final. En el ámbito empresarial, la gestión del delivery se preocupa del diseño, planificación, implementación y mejoramiento de los flujos asociados a la entrega, generalmente sujeta a restricciones de tiempos y costos (Palenque, 2018).

2.2. Geolocalización

Desde épocas antiguas el hombre enfrento con la necesidad de saber dónde se encuentra exactamente, ya sea para volver a su hogar, ir de un punto A al punto B, o saber dónde se encuentra otra persona. Orientarse en la superficie del planeta no fue fácil y es por ello que tuvo que buscar la manera de guiarse buscar puntos de referencias visuales, pero ello no fue tan útil ya que a medida que se alejaban de estas referencias volvían a desorientarse. Ya en los años 150 A.C., en la antigua Grecia se tiene la invención del “astrolabio” este aparato fue creado por Hiparco con la finalidad de calcular y predecir las posiciones de los cuerpos celestes, determinando la posición y altura de las estrellas sobre el cielo pero este también

podía medir la latitud (medida que obtiene la posición entre nuestro norte y sur) a partir de la hora o viceversa y permitiendo la medición de distancias por triangulación, posteriormente fue usado en el siglo XIII para viajes en el mar teniendo que realizarse algunas modificaciones de esa manera se tiene el astrolabio marino. También se tiene otro instrumento la brújula que fue inventado en China 200 años A.C., este instrumento funciona por medio de una aguja imantada señalando el norte magnético terrestre. Con este instrumento solo tenemos la latitud y la altitud medidas que fueron precisas para tal época especialmente para los marineros (Omicrono-EE, 2020), pero ¿dónde queda la longitud?

La longitud (medida que se obtiene la posición desde este a oeste), se desarrollaron distintos métodos complejos para calcular esta medida hechos por Galileo, Vespucci, usando trayectorias de la luna u otros planetas. Para obtener la longitud se necesita saber la hora exacta entre dos lugares, la ubicación en la se encuentre uno y un lugar de referencia esto ya lo sabía el inventor del astrolabio Hiparco que también propuso un sistema de medid similar a la actual que fue perfeccionado por los sabios del mundo árabe en el siglo XII. Fue de esta manera que se desarrolló la obtención de las medidas de latitud y longitud, pero ello aún no se quedó allí ya que en se desarrollaron otros aparatos que permiten darnos a conocer nuestra posición (Omicrono-EE, 2020).

2.2.1. Sistema de posicionamiento global

GPS acrónimo que referencia al sistema de posicionamiento global, permite obtener la posición de un objeto en la tierra, fue desarrollado por el Departamento de Defensa de los Estados Unidos, este sistema dependiendo de satélites que orbitan el planeta tierra. Para el desarrollo de este nuevo aparato se introdujo la teoría de la relatividad general de Einstein y la física de Newton usando una forma de triangulación, se podría decir que el posicionamiento es resultado de la intersección de un único punto de tres circunferencias (GPS, 2019).

Actualmente podemos encontrar este tipo de sistema de localización en los dispositivos móviles Smartphone, podemos saber el posicionamiento desde nuestros celulares.

El presente proyecto propuesto tiene el enfoque de resolver y disminuir la susceptibilidad en el transporte de paquetes sistematizando el proceso que con lleva realizar tal trabajo, proponiendo realizar una administración en la recolección y entrega de paquetes realizando un seguimiento por medio del GPS que cuentan los celulares Smartphone, y oportuna de designación de personal disponible, y anunciar los estados del envío al usuario final.

2.3. Diseño de Software

2.3.1. Ingeniería de Software

La ingeniería de software es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software; es decir, la aplicación de la ingeniería al software. Relacionado al estudio de enfoques sistemático, disciplinado y cuantificable aplicado por un equipo de software podría ser algo burdo para otro. Se necesita disciplina, pero también adaptabilidad y agilidad (Pressman, 2010).

Según Somerville (2011) la ingeniería de software es una disciplina de ingeniería que se interesa por todos los aspectos de la producción de software.

La Ingeniería de Software se puede considerar como la ingeniería aplicada al software, esto es, por medios sistematizados y con herramientas preestablecidas, la aplicación de ellos de la forma más eficiente para la obtención de resultados óptimos, objetivos que siempre busca la ingeniería.

2.3.2. Ingeniería de Requerimientos

La ingeniería de requerimientos del software es un proceso de descubrimiento, refinamiento, modelado y especificación. Tanto el desarrollador como el cliente tienen un papel activo en la ingeniería de requerimientos (Pressman, 2010).

Para generar un sistema completamente funcional y acorde a las exigencias del cliente es indispensable identificar los requerimientos que el sistema debe cumplir y así satisfacer las necesidades de los usuarios como de los clientes. Para dar una solución de excelente calidad

y funcionalidad el proceso de especificación de requerimientos debe ser claro y adecuado a las exigencias del ambiente donde se aplicará la solución, de lo contrario el producto obtenido contará con falencias y provocará retardos que afectarán organizacional y financieramente tanto al equipo de desarrollo como al cliente.

2.3.3. Análisis de requerimientos

El análisis de requerimientos es una tarea de ingeniería del software que cubre el hueco entre la definición del software a nivel sistema y el diseño de software. El análisis de requerimientos permite al ingeniero de sistemas especificar las características operacionales del software función, datos y rendimientos, indica la interfaz del software con otros elementos del sistema y establece las restricciones que debe cumplir el software. El ingeniero de requisitos debe poseer habilidades particulares para facilitar la comunicación con el cliente y ganar su confianza (Piattini, 2020).

Con el análisis obtenemos como resultado la especificación operativa del software, indica la interfaz de éste y otros elementos del sistema, y establece las restricciones que limitan al software. El análisis de los requerimientos permite construir sobre los requerimientos básicos las tareas de concepción, indagación y negociación, que son parte de la ingeniería de los requerimientos.

2.3.4. UML Web Engineering - UWE

En UWE (2020), define que UML Web Engineering es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

En el marco de UWE es necesario la definición de un perfil UML (extensión) basado en estereotipos con este perfil se logra la asociación de una semántica distinta a los diagramas del UML puro, con el propósito de acoplar el UML a un dominio específico, en este caso, las

aplicaciones Web. Entre los principales modelos de UWE podemos citar: el modelo lógico-conceptual, modelo navegacional, modelo de presentación, visualización de Escenarios Web y la interacción temporal, entre los diagramas: diagramas de estado, secuencia, colaboración y actividad.

2.3.4.1. Modelado de UWE.

El enfoque UWE proporciona una notación específica de dominio, un proceso de desarrollo basado en modelos y soporte de herramientas para la ingeniería de aplicaciones web. Las actividades base de modelado de UWE son el análisis de requerimientos, el modelo conceptual, el modelo navegacional y el modelo de presentación, se pueden sumar otros modelos como lo son el modelo de interacción y la visualización de escenarios Web (UWE, 2020).

El modelo que propone UWE está compuesto por etapas o sub-modelos:

a) Modelo Lógico-Conceptual.

En UWE (2020), apunta a construir un modelo conceptual de una aplicación Web, procura no hacer caso en la medida de lo posible de cuestiones relacionadas con la navegación, y de los aspectos de interacción de la aplicación Web. La construcción de este modelo lógico-conceptual se debe llevar a cabo de acuerdo con los casos de uso que se definen en la especificación de requerimientos. El modelo conceptual incluye los objetos implicados en las actividades típicas que los usuarios realizarán en la aplicación Web.

b) Modelo de Navegación

Consta de la construcción de dos modelos de navegación, el modelo del espacio de navegación y el modelo de la estructura de navegación. El primero especifica que objetos serán visitados por el navegador a través de la aplicación. El segundo define como se relacionan (Galiano, 2020).

c) Modelo de presentación

Describe dónde y cómo los objetos de navegación y accesos primitivos serán presentados al usuario, es decir, una representación esquemática de los objetos visibles al usuario (Galiano, 2020).

d) Escenarios Web

Permiten detallar la parte dinámica del modelo de navegación, especificando los eventos que disparan las situaciones, definen condiciones y explícitamente incluyen las acciones que son realizadas. Junto con el modelo de interacción temporal, los escenarios Web proveen la representación funcional dinámica del modelo de navegación (Galiano, 2020).

2.3.4.2. Fases de la UWE.

UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrando además su atención en aplicaciones personalizadas o adaptativas.

Las fases o etapas a utilizar son:

a) Captura, análisis y especificación de requisitos

En simple palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web. Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipado de la interfaz de usuario (Galiano, 2020).

b) Diseño del sistema

Se basa en la especificación de requisitos producido por el análisis de los requerimientos, el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web (Galiano, 2020).

c) Codificación del software

Durante esta etapa se realizan las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior (Galiano, 2020).

d) Pruebas

Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código (Galiano, 2020).

e) La Instalación o Fase de Implementación

Es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados, y, eventualmente, configurados, todo ello con el propósito de ser ya utilizados por el usuario final (Galiano, 2020).

f) El Mantenimiento

Es el proceso de control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado de la fase de pruebas de control (Galiano, 2020).

2.4. Desarrollo de Software

La metodología de desarrollo de software tiene como principal objetivo aumentar la calidad del software que se produce en todas y cada una de sus fases de desarrollo. No existe una metodología de software universal, ya que toda metodología debe ser adaptada a las características de cada proyecto, exigiéndose así que el proceso sea configurable. Las metodologías de desarrollo se pueden dividir en dos grupos de acuerdo con sus características y los objetivos que persiguen: ágiles y robustas (Ecured, 2020).

2.4.1. Metodología ágil

Se caracterizan por hacer énfasis en la comunicación cara a cara, es decir, se basan en una fuerte y constante interacción, donde clientes desarrolladores y desarrolladores trabajan constantemente juntos, estableciéndose así una estrecha comunicación. Estas metodologías

están orientadas al resultado del producto y no a la documentación; exige que el proceso sea adaptable, permitiendo realizar cambios de último momento. Se puede hacer mención dentro de las metodologías ágiles a: XP¹, Scrum y Crystal Methodologies (Pressman, 2010).

2.4.2. Metodología Ágil Scrum

Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos. En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales (Pressman, 2010).

2.4.2.1. Principios de agilidad

En Pressman (2010), menciona los siguientes principios de agilidad:

- La prioridad más alta es satisfacer al cliente a través de la entrega pronta y continua de software valioso.
- Son bienvenidos los requerimientos cambiantes, aun en una etapa avanzada del desarrollo. Los procesos ágiles dominan el cambio para provecho de la ventaja competitiva del cliente.
- Entregar con frecuencia software que funcione, de dos semanas a un par de meses, de preferencia lo más pronto que se pueda.

¹ siglas en inglés Extreme Programming

- Las personas de negocios y los desarrolladores deben trabajar juntos, a diario y durante todo el proyecto.
- Hay que desarrollar los proyectos con individuos motivados. Debe darse a éstos el ambiente y el apoyo que necesiten, y confiar en que harán el trabajo.
- El método más eficiente y eficaz para transmitir información a los integrantes de un equipo de desarrollo, y entre éstos, es la conversación cara a cara.
- La medida principal de avance es el software que funciona.
- Los procesos ágiles promueven el desarrollo sostenible. Los patrocinadores, desarrolladores y usuarios deben poder mantener un ritmo constante en forma indefinida.
- La atención continua a la excelencia técnica y el buen diseño mejora la agilidad.
- Es esencial la simplicidad: el arte de maximizar la cantidad de trabajo no realizado.
- Las mejores arquitecturas, requerimientos y diseños surgen de los equipos con organización propia.
- El equipo reflexiona a intervalos regulares sobre cómo ser más eficaz, para después afinar y ajustar su comportamiento en consecuencia.

2.4.2.2. Elementos de Scrum

a) Herramientas de Scrum

En Kniberg (2007) menciona como herramienta de Scrum a:

- **Product Backlog:** Conjunto de requisitos denominados historias de usuario descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.

- **Sprint Planning:** Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.
- **Sprint Backlog:** Es el punto de entrada de cada Sprint. Es una lista que tiene los ítems de la Product Backlog List que van a ser implementados en el siguiente Sprint.
- **Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo.
- **Daily sprint meeting:** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.
- **Retrospectiva:** Reunión que se celebra al final del sprint y en la que el equipo presenta las historias conseguidas mediante una demostración del producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos.

b) Responsabilidades

En Scrum, se tiene una definido responsabilidades en el equipo de desarrollo como lo indica Kniberg (2007):

- **Product owner:** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio y él es responsable del proyecto (entregar un valor superior al dinero invertido). Traslada la visión del proyecto al equipo, formaliza las prestaciones en historias a incorporar en el Product Backlog y las reprioriza de forma regular.

- Scrum master o facilitador: Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner.
- Team: Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.

2.4.2.3. Proceso trabajo de Scrum

En la figura la 2.1 describe el proceso de trabajo que conlleva la metodología Scrum. En Scrum un proyecto se ejecuta en ciclos temporales cortos y de duración fija iteraciones que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback de producto real y reflexión. Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite (Pressman, 2010).

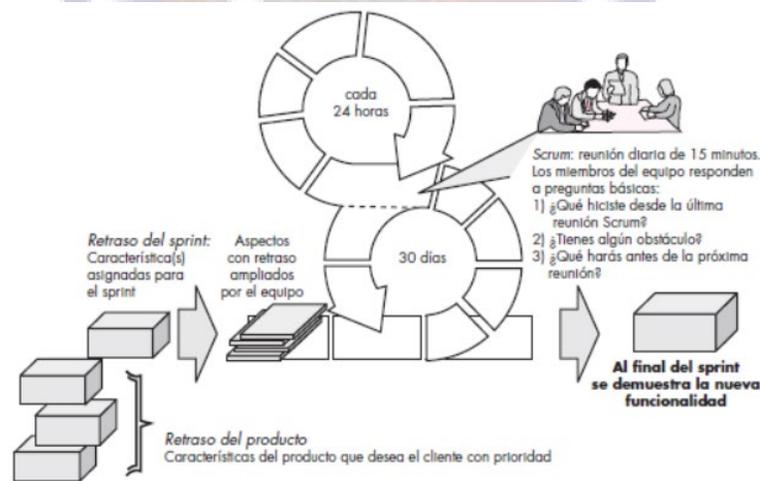


Figura 2.1 Descripción del proceso de Scrum
Fuente: (Pressman, 2010)

El proceso parte de la lista de objetivos y requisitos priorizada del producto, que actúa como plan del proyecto. En esta lista el cliente Product Owner prioriza los objetivos balanceando

el valor que le aportan respecto a su coste que el equipo estima considerando su avance y quedan repartidos en iteraciones y entregas.

Las actividades que se llevan a cabo en Scrum son las siguientes (Kniberg, 2007):

a) Planificación de la iteración

El primer día de la iteración se realiza la reunión de planificación de la iteración. Tiene dos partes:

- Selección de requisitos. El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que prevé que podrá completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.
- Planificación de la iteración. El equipo elabora la lista de tareas de la iteración necesarias para desarrollar los requisitos seleccionados. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se auto asigna las tareas, se auto organizan para trabajar incluso en parejas o grupos mayores con el fin de compartir conocimiento o para resolver juntos objetivos especialmente complejos.

b) Ejecución de la iteración

Cada día el equipo realiza una reunión para compartir avances o dificultades que se tuvo el día anterior, normalmente delante de un tablero físico o pizarra. El equipo inspecciona el trabajo que el resto está realizando dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo para poder hacer las adaptaciones necesarias que permitan cumplir con la previsión de objetivos a mostrar al final de la iteración. En la reunión cada miembro del equipo responde a tres preguntas:

- ¿Qué he hecho desde la última reunión para ayudar al equipo a cumplir su objetivo?
- ¿Qué voy a hacer para ayudar al equipo a cumplir su objetivo?

- ¿Qué impedimentos tengo o voy a tener que nos impidan conseguir nuestro objetivo?

Durante la iteración el Facilitador Scrum Master se encarga de que el equipo pueda mantener el foco para cumplir con sus objetivos. Elimina los obstáculos que el equipo no puede resolver por sí mismo, como también protege al equipo de interrupciones externas que puedan afectar el objetivo de la iteración o su productividad.

Durante la iteración, el o los clientes junto con el equipo refinan la lista de requisitos para prepararlos para las siguientes iteraciones y, si es necesario, cambian o re planifican los objetivos del proyecto con el objetivo de maximizar la utilidad de lo que se desarrolla y el retorno de inversión.

c) Inspección y adaptación

El último día de la iteración se realiza la reunión de revisión. Tiene dos partes:

- Revisión de 1,5 horas. El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias.
- Retrospectiva de 1,5 horas. El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar adecuadamente, mejorando de manera continua su productividad. El Facilitador se encargará de eliminar o escalar los obstáculos identificados que estén más allá del ámbito de acción del equipo.

2.5. Herramientas de Desarrollo

2.5.1. Framework Codeigniter

Un framework nos permite un trabajo más ligero ya que es un conjunto de herramientas que facilitan el desarrollo de aplicaciones web, su función es estructurar de mejor manera el código en un proyecto con el fin de organizarlo y también poder reutilizarlo. Codeigniter es un framework para aplicaciones web de código abierto creando sitios web dinámicos con

PHP². Permite a los desarrolladores realizar proyectos ofreciendo un conjunto de bibliotecas para tareas comunes, así como una interfaz simple y una estructura lógica para acceder esas bibliotecas. CodeIgniter se basa en el patrón Modelo-vista-controlador donde se divide en capas el código con lo que es la lógica del negocio en el Modelo y la parte visual del sistema FrontEnd³ con el BackEnd modelo de negocio, siendo gestionado la interacción de estas capas por lo que es el Controlador (EllisLab, 2020).

CodeIgniter nació de ExpressionEngine, esencialmente una colección de clases refaccionadas originalmente escritas para el CMS⁴ insignia de EllisLab. CodeIgniter se convirtió en un kit de herramientas simple y elegante, que permite el rápido desarrollo de sitios web y aplicaciones web, y atrae a miles de talentosos desarrolladores de PHP.

2.5.2. MySql

El servicio de base de datos Oracle MySQL es un servicio de base de datos totalmente administrado que permite a los desarrolladores desarrollar e implementar rápidamente aplicaciones nativas en la nube seguras utilizando la base de datos de código abierto más popular del mundo. El motor de análisis del servicio de base de datos MySQL es el único servicio de MySQL con un motor de análisis en tiempo real integrado y escalable de forma masiva.

2.5.3. Android

Android es un sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tabletas y también para relojes inteligentes, televisores y automóviles. Inicialmente fue desarrollado por Android Inc., empresa que Google respaldó económicamente (Gallego, 2012).

²Hypertext Preprocessor, PHP es un lenguaje de programación de uso general y de código abierto.

³ Referencian a la separación de intereses entre una capa de presentación y una capa de acceso a datos.

⁴ Un sistema de gestión de contenidos o CMS (del inglés content management system)

Según Gallego (2012), en la actualidad los sistemas operativos abstraen al programador del hardware. Un ejemplo clásico es Symbian. Pero este tipo de plataformas todavía requieren que el programador escriba código C/C++ complicado, haciendo uso de librerías propietarias. Especiales complicaciones pueden surgir cuando se trabaja con hardware específico, como GPS, trackballs, pantallas táctiles, y otros.

a) Arquitectura

Según la página oficial de DeveloperAndroid (2020), indican que Android es una pila de software de código abierto basado en Linux creada para una variedad amplia de dispositivos y factores de forma. En la siguiente figura 2.2, se muestran los componentes principales de la plataforma Android.

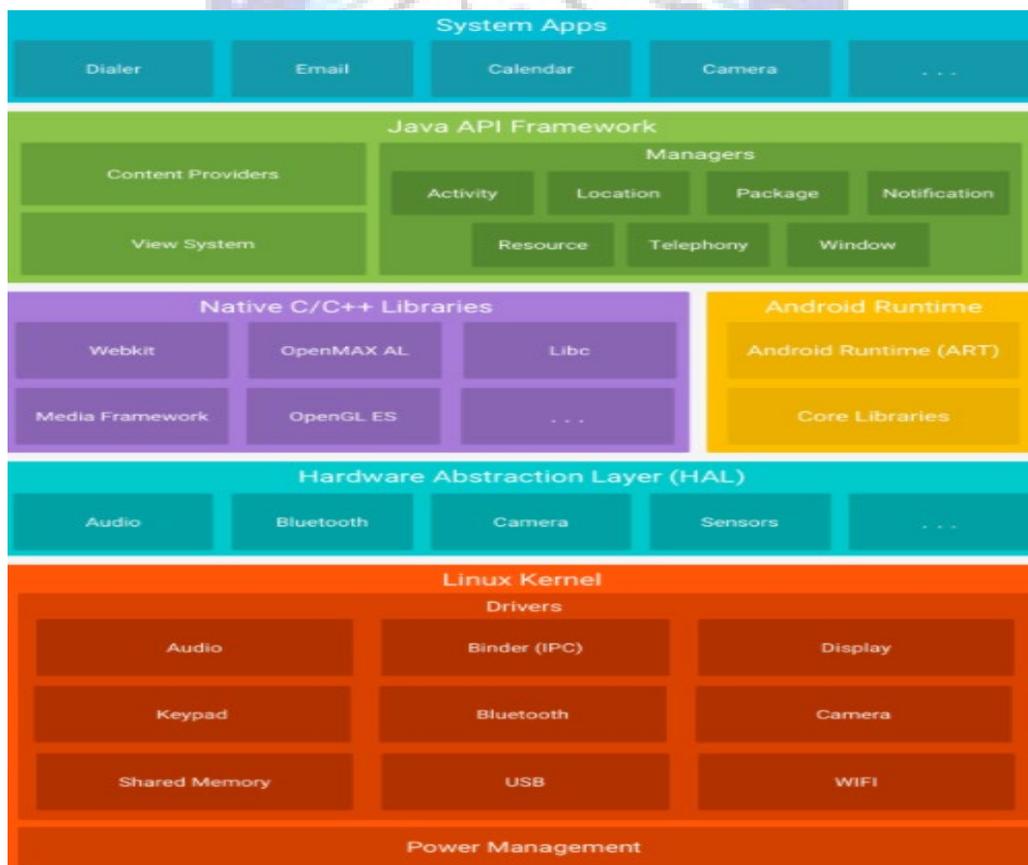


Figura 2.2 Arquitectura de Android

Fuente: (Android developers)

- Aplicaciones: las aplicaciones base incluyen un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos y otros. Todas las aplicaciones están escritas en lenguaje de programación Java.
- Marco de trabajo de aplicaciones: los desarrolladores tienen acceso completo a las mismas API del entorno de trabajo usados por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades.
- Bibliotecas: Android incluye un conjunto de bibliotecas de C/C++ usadas por varios componentes del sistema. Estas características se exponen a los desarrolladores a través del marco de trabajo de aplicaciones de Android. Algunas son: System C library (implementación biblioteca C estándar), bibliotecas de medios, bibliotecas de gráficos, 3D y SQLite, entre otras.
- Runtime de Android: Android incluye un set de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual. La Máquina Virtual está basada en registros y corre clases compiladas por el compilador de Java que han sido transformadas al formato dex por la herramienta incluida dx.
- Núcleo Linux: Android depende de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila de software.

b) Entorno de Desarrollo

En la página oficial DeveloperAndroid (2020), Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de apps para Android y está basado

en IntelliJ IDEA, este es un IDE⁵ oficial para la plataforma Android totalmente gratuito a través de la Licencia Apache 2.0 y multiplataforma⁶.

Android Studio ofrece funciones que aumentan la productividad durante la compilación de apps estas son:

- Sistema de compilación flexible basado en Gradle.
- Entorno unificado para desarrollo en todo tipo de dispositivos Android.
- Instant Run, para aplicar cambios sin la necesidad de compilar un nuevo APK¹⁴.
- Integración de plantillas de código y GitHub.
- Herramientas Lint para detectar problemas de rendimiento, uso, compatibilidad de versión, etc.
- Compatibilidad con C++ y NDK
- Soporte integrado para Google Cloud Platform.

2.5.4. Google Maps Platform

En la página oficial de DeveloperAndroid (2020), Google Maps Platform es un conjunto de API⁷ que los desarrolladores pueden usar a fin de incorporar Google Maps en apps para dispositivos móviles y páginas web o recuperar datos de Google Maps, permitiendo a los desarrolladores programar dentro de mapas basándose en un conjunto de librerías y servicios proporcionadas por la API de Google Maps.

La API de JavaScript de Maps le permite personalizar mapas con su propio contenido e imágenes para mostrarlos en páginas web y dispositivos móviles. La API de JavaScript de

⁵ por sus siglas en inglés, Entorno de Desarrollo Integrado

⁶ Soporte para Microsoft Windows, macOS y GNU/Linux

⁷ Del inglés application programming interface, es un conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software

Maps presenta cuatro tipos de mapas básicos que puede modificar utilizando capas y estilos, controles y eventos, y varios servicios y bibliotecas (DeveloperAndroid, 2020).

- **Círculos:** Google Maps incluye la clase `circle` que permite trazar círculos en los cuales se puede definir colores, grosores y niveles de opacidad personalizados para el borde del círculo, así como colores y opacidades personalizados para el área que engloba.
- **Polilíneas:** Google Maps incluye la clase `polyline` que permite trazar caminos, como la creación de rutas de ciclismo, caminata y otros. Las polilíneas se componen de varias líneas conectadas. Una línea consiste en dos puntos: un punto de partida y un punto de terminal. Estos se componen de coordenadas.
- **Servicio de rutas:** El servicio de rutas de Google Maps proporciona la clase `directions Service`, que despliega una ruta completamente modificable, a la cual se le debe especificar un origen y destino mediante coordenadas geográficas, las coordenadas geográficas se deben representar en base a pares ordenados de latitud y longitud, además esta ruta se adapta automáticamente a calles y avenidas permitiendo describir rutas dentro de Google Maps.

2.6. Métricas de Calidad

2.6.1 ISO/IEC 9126

Esta norma Internacional fue publicada en 1992, la cual es usada para la evaluación de la calidad de software, llamado “Information technology-Software product evaluation-Quality characteristics and guidelines for their use”; o también conocido como ISO 9126. Este estándar describe 6 características generales: Funcionalidad, Confiabilidad, Usabilidad, Eficiencia, Mantenibilidad, y Portabilidad (ISO, 2020), las describimos a continuación.

a) Funcionalidad

Funcionalidad es la capacidad del software de cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas cuando es utilizado en condiciones específicas. A continuación, se muestra la característica de Funcionalidad y se divide en cinco criterios:

- Adecuación: La capacidad del software para proveer un adecuado conjunto de funciones que cumplan las tareas y objetivos especificados por el usuario.
- Exactitud: La capacidad del software para hacer procesos y entregar los resultados solicitados con precisión o de forma esperada.
- Interoperabilidad: La capacidad del software de interactuar con uno o más sistemas específicos.
- Seguridad: La capacidad del software para proteger la información y los datos de manera que los usuarios o los sistemas no autorizados no puedan acceder a ellos para realizar operaciones, y la capacidad de aceptar el acceso a los datos de los usuarios o sistemas autorizados
- Conformidad de la funcionalidad: La capacidad del software de cumplir los estándares referentes a la funcionalidad.

b) Confiabilidad

La confiabilidad es la capacidad del software para asegurar un nivel de funcionamiento adecuado cuando es utilizando en condiciones específicas. En este caso la confiabilidad se amplía en sostener un nivel especificado de funcionamiento y no una función requerida, la confiabilidad se divide en cuatro criterios:

- Madurez: La capacidad que tiene el software para evitar fallas cuando encuentra errores. Ejemplo, la forma como el software advierte al usuario cuando realiza operaciones en la unidad de diskette vacía, o cuando no encuentra espacio suficiente el disco duro donde esta almacenando los datos.

- Tolerancia a errores: La capacidad que tiene el software para mantener un nivel de funcionamiento en caso de errores.
- Recuperabilidad: La capacidad que tiene el software para restablecer su funcionamiento adecuado y recuperar los datos afectados en el caso de una falla.
- Conformidad de la fiabilidad: La capacidad del software de cumplir a los estándares o normas relacionadas a la fiabilidad.

c) Usabilidad

La usabilidad es la capacidad del software de ser entendido, aprendido, y usado en forma fácil y atractiva. Algunos criterios de funcionalidad, fiabilidad y eficiencia afectan la usabilidad, pero para los propósitos de la ISO/IEC 9126 ellos no clasifican como usabilidad. La usabilidad está determinada por los usuarios finales y los usuarios indirectos del software, dirigidos a todos los ambientes, a la preparación del uso y el resultado obtenido.

La usabilidad se divide en cinco criterios:

- Entendimiento: La capacidad que tiene el software para permitir al usuario entender si es adecuado, y de una manera fácil como ser utilizado para las tareas y las condiciones particulares de la aplicación. En este criterio se debe tener en cuenta la documentación y de las ayudas que el software entrega.
- Aprendizaje: La forma como el software permite al usuario aprender su uso. También es importante considerar la documentación.
- Operabilidad: La manera como el software permite al usuario operarlo y controlarlo.
- Atracción: La presentación del software debe ser atractiva al usuario. Esto se refiere a las cualidades del software para hacer más agradable al usuario, ejemplo, el diseño gráfico.

- Conformidad de uso: La capacidad del software de cumplir los estándares o normas relacionadas a su usabilidad.

d) Eficiencia

Esta característica permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos usados. Aspectos a evaluar:

- Comportamiento con respecto al Tiempo.
- Comportamiento con respecto a Recursos.

e) Mantenibilidad

La mantenibilidad es la cualidad que tiene el software para ser modificado. Incluyendo correcciones o mejoras del software, a cambios en el entorno, y especificaciones de requerimientos funcionales, el mantenimiento se divide en cinco criterios:

- Capacidad de ser analizado: La forma como el software permite diagnósticos de deficiencias o causas de fallas, o la identificación de partes modificadas.
- Cambiabilidad: La capacidad del software para que la implementación de una modificación se pueda realizar, incluye también codificación, diseño y documentación de cambios.
- Estabilidad: La forma como el software evita efectos inesperados para modificaciones del mismo.
- Facilidad de prueba: La forma como el software permite realizar pruebas a las modificaciones sin poner el riesgo los datos.
- Conformidad de facilidad de mantenimiento: La capacidad que tiene el software para cumplir con los estándares de facilidad de mantenimiento.

f) Portabilidad

La capacidad que tiene el software para ser trasladado de un entorno a otro, se divide en cinco criterios:

- **Adaptabilidad:** Es como el software se adapta a diferentes entornos especificados ya sea en hardware o software sistemas operativos sin que implique reacciones negativas ante el cambio
- **Facilidad de instalación:** La facilidad del software para ser instalado en un entorno específico o por el usuario final.
- **Coexistencia:** La capacidad que tiene el software para coexistir con otros, compartiendo recursos comunes.
- **Reemplazabilidad:** La capacidad que tiene el software para ser reemplazado por otro software del mismo tipo, y para el mismo objetivo. Ejemplo, la reemplazabilidad de una nueva versión es importante para el usuario, la propiedad de poder migrar los datos a otro software de diferente proveedor.
- **Conformidad de portabilidad:** La capacidad que tiene el software para cumplir con los estándares relacionados a la portabilidad.

2.6. Costo y/o Beneficio

2.6.1. Método de COCOMO

Entre los distintos métodos de estimación de costes de desarrollo de software, el modelo COCOMO (Constructive Cost Model) desarrollado por Barry M. Boehm, se engloba en el grupo de los modelos algorítmicos que tratan de establecer una relación matemática la cual permite estimar el esfuerzo y tiempo requerido para desarrollar un producto (Sanchez, 1999).

Se clasifica los proyectos software en tres tipos distintos:

- **Orgánico;** son proyectos software relativamente pequeños y sencillos, y similares a otros ya desarrollados anteriormente

- Semi-acoplado: proyectos intermedios en complejidad y tamaño, donde la experiencia en este tipo de proyectos es variable, y las restricciones intermedias.
- Empotrado: proyectos bastantes complejos, en los que apenas se tienen experiencia y se engloban en un entorno de gran innovación técnica. Además, se trabaja con unos requisitos muy restrictivos y de gran volatilidad.

Y por otro lado existen diferentes modelos que define COCOMO:

- Modelo básico: Se basa exclusivamente en el tamaño expresado en LDC.
- Modelo intermedio: Además del tamaño del programa incluye un conjunto de medidas subjetivas llamadas conductores de costes.
- Modelo avanzado: Incluye todo lo del modelo intermedio además del impacto de cada conductor de coste en las distintas fases de desarrollo.

Para nuestro caso el modelo intermedio será el que usaremos, dado que realiza las estimaciones con bastante precisión.

2.7. Seguridad

La seguridad del software es una actividad del aseguramiento del software que se centra en la identificación y evaluación de los peligros potenciales que podrían afectarlo negativamente y que podrían ocasionar que falle todo el sistema. Si los peligros se identifican al principio del proceso del software, las características de su diseño se especifican de modo que los eliminen o controlen. (Pressman, 2010).

Según un artículo publicado por Guaman, Torres, 2015, una aplicación web, especialmente que se ejecuta en Internet, es más vulnerable a ataques de los hackers que una aplicación autónoma o cliente-servidor típico. Hay varias razones para esto:

- Disponibilidad y accesibilidad: Muchas aplicaciones web están disponibles para los usuarios públicos en cualquier momento del día o de la noche. Como los servidores

web tienen que permitir el acceso a usuarios públicos y no tienen la protección completa de los cortafuegos típicos de una empresa.

- Familiaridad: La mayoría de los atacantes, incluso los menos sofisticados, conocen las interfaces web. Un navegador web es fácil de obtener y es uno de los programas de aplicación más comunes. El protocolo HTTP está bien definido, y existen muchas herramientas de hacking creadas para ayudar a los atacantes a penetrar y comprometer las aplicaciones web.
- Facilidad: La configuración de un servidor web, contenedor web y aplicación web para uso público es extremadamente compleja. Los atacantes, frecuentemente, pueden aprovechar esta complejidad y explotar deficiencias en la configuración de la aplicación o del sistema.
- Publicidad: El ego de algunos hackers experimentados es la publicidad, la fama, o un simple deseo de probar que pueden hacer algo que pocas personas pueden hacer.

2.7.1 Autenticación de Usuarios

2.7.1.1 Autenticación

Confirma la identidad del usuario: la autenticación consiste en comprobar la identidad del usuario que entra a su aplicación. Este proceso se lleva a cabo en dos pasos: primero, la identificación, donde el usuario declara quién es. El segundo paso consiste en comprobar dicha identificación. Generalmente, este proceso se realiza por medio de cuentas de usuario y contraseñas. Esta etapa es el primer nivel de seguridad (Pressman, 2010).

2.7.1.2 Autorización

Mecanismo de filtrado que permite el acceso al entorno cliente servidor, las autorizaciones definen lo que un usuario puede hacer en una aplicación: básicamente, usted define lo que el usuario podrá ver, hacer y modificar en la aplicación (Pressman, 2020).

Existen dos métodos para definir las autorizaciones:

- La forma más segura es prohibir todo desde un principio, para después otorgar los permisos y abrir posibilidades. Sin embargo, utilizando este método, usted corre el riesgo de olvidar definir algún permiso, imposibilitando así el trabajo de un usuario final u otorgando permisos a usuarios no indicados.
- La forma más rápida es autorizar toda la acción, para después asignar restricciones y así prohibir algunas de ellas. Esta forma es más rápida que la anterior puesto que generalmente existen menos restricciones que permisos.



CAPÍTULO III

MARCO APLICATIVO

En el presente capítulo contiene el diseño general del sistema web, donde mediante un aplicativo móvil proporciona información de la ubicación actual por GPS del personal de la empresa, que es responsable del cuidado de los paquetes que son trasladados en el área urbana, estos datos son registrados en la base de datos del sistema web, se genera la captura de coordenadas geográficas en tiempo real que son desplegadas en el mapa que contiene sistema.

Como principal tarea se definen los requerimientos y/o características esenciales del sistema de los cuales se desarrollan historias de usuario, presentando de esta manera un panorama amplio del flujo de información que contendrá el sistema.

La gestión del proyecto es fundamental y necesariamente se tiene que contar con directrices que nos ayuden en el desarrollo del proyecto en lo que respecta a la metodología de trabajo y documentación, de esta manera priorizando funcionalidades y tiempos de desarrollo en el proyecto.

Para llevar a cabo el modelado del sistema tenemos presente UWE que nos ayudara en la documentación del respectivo proyecto y con la metodología de trabajo ágil Scrum se llevara a cabo tareas en cierto tiempo definido, ambas metodologías están orientadas al desarrollo de aplicaciones web respectivamente. Se desarrollarán las fases que presenta Scrum, complementando con la metodología de modelado de requisitos para tener un análisis completo del sistema.

3.1. Elaboración del Product-Backlog

En esta fase definiremos el Product-Backlog, en el cual se especifican todas las tareas y requerimientos del producto.

3.1.1. Roles de usuarios

- Clientes o empresas: Son los usuarios que estarán registrados en el sistema web, y realizarán la solicitud de envío de paquete, desde un portal habilitado para dicho suceso.
- Repartidor: Es un usuario en el sistema, que estará a cargo de realizar el proceso de recolección y/o entrega de paquete. Será monitoreado por el administrador del sistema, que a la vez le designará tareas de recolección y/o entrega de paquetes, siempre y cuando esté disponible.
- Administrador del sistema y/o encargado: Se encarga de administrar la llegada de un nuevo paquete que será designada al personal más cercano al punto de recolección, como también controlar su ubicación, generar reportes, control de usuarios del sistema. Se debe mencionar también que se divide en dos roles la administración del sistema teniendo a un súper administrador que goza de todos los privilegios, y un administrador básico que tiene como tarea controlar el proceso relacionado a la actividad de recolección y entrega del paquete.

3.1.2. Requerimientos

a) Administración de datos del sistema

- R1-1 Diseño de la base de datos
- R1-2 Definición de usuario del sistema: se controlará el acceso al sistema según tipo de usuario

b) Módulo de gestión de usuario

- R2-1 Registro usuario repartidor: El registro lo realiza solo el súper administrador
- R2-2 Registro administradores: Se registrará a los empleados según la definición de rol de usuario, este trabajo será llevado a cabo por el súper administrador
- R2-3 Modificación y habilitación/inhabilitación de usuarios del sistema: Solo el súper administrador puede realizar modificaciones de usuarios pertenecientes al sistema como también los puede inhabilitar momentáneamente según sea el caso. Esta condición es aplicada a clientes, súper administrador, administrador y repartidor.
- R2-4 Generación de reportes: La generación de reportes será realizado por Súper administrador o administrador básico.

c) Módulo para usuario cliente

- R3-1 Registro de cliente en el sistema: Para el registro del usuario cliente se habilita un sitio exclusivo y tener acceso a los servicios que ofrece el sistema.
- R3-2 Registro de solicitud de servicio: El Cliente realiza la solicitud de servicio enviando la descripción y datos para envío de paquete, este suceso estará en espera de confirmación de la misma.
- R3-3 Estado de confirmación del servicio: Se realiza el despliegue de estados, que el usuario cliente a espera de confirmación, este estado puede ser afirmativa o negativa. Afirmativa se desplegará en pantalla del Cliente la confirmación que se acepta el servicio y se le asigna personal para ir por su paquete y además generar un código QR que deberá ser agregado a su paquete. Negativa será en caso que no haya personal disponible para recoger su paquete de igual manera se le desplegará tal información.

d) Módulo de control y seguimiento

- R4-1 Aceptación de solicitud de envío de paquete: El cliente una vez realiza la solicitud para enviar un paquete, esta llegará al sistema web en la cual se verifica el punto de inicio para recolectar el paquete y su destino final desplegando estos datos

en un mapa que será gestionado por el administrador. El administrador debe asignar a un usuario repartidor, según el despliegue de su ubicación del repartidor en el mapa y este se encuentre cercano al punto de recolección y/o esté disponible, una vez seleccionado a un repartidor disponible se confirma la aceptación de servicio como respuesta al cliente que solicitó el servicio.

- R4-2 Confirmaciones sobre el estado del paquete: Para la recolección y entrega del paquete debe cambiar el estado, desde la recolección del paquete, su recepción y entrega, estos estados serán enviados al sistema por medio de un escaneo de código QR.
- R4-3 Gestión del seguimiento del paquete: El seguimiento del paquete se realiza a partir de que el usuario repartidor recolecte paquete y escanee el código QR que se le asignó al paquete que está pegado en el este último. Una vez escaneado el código QR se envía el dato al sistema, de esta manera dar a conocer que el paquete fue recolectado y está por ser transportado por el repartidor. Una vez llegado a su destino o punto de entrega el usuario repartidor al momento de entregar el paquete escanea nuevamente el código QR el cual indica al sistema que el paquete está siendo entregado a la persona que corresponde, cabe resaltar que al escanear el paquete se captura las coordenadas actuales del paquete.

f) Módulo de gestión de personal

- R5-1 Aplicación móvil de control: Se proporciona acceso al sistema al usuario repartidor por medio de una aplicación móvil, el cual permite realizar inicio de sesión en el sistema.
- R5-2 Seguimiento y control de usuario repartidor: Se realiza una gestión de inicio de trabajo del usuario repartidor de la empresa, se debe mostrar su ubicación actual. Para ello se lleve a cabo la aplicación móvil gestiona él envió de coordenadas y/o ubicación

actual de este usuario, el envío de coordenadas será iniciado por el usuario al momento de estar en horario de trabajo.

- R5-3 Confirmación y rechazo de paquete por el repartidor: Al momento que un administrador asigne un paquete a ser procesado, se envía los datos del paquete por medio de una notificación Push al usuario repartidor, este último según a las condiciones que se le presente al momento se le da a conocer que tiene un nuevo proceso de envío de paquete, donde puede confirmar que realizara el proceso enviando la aceptación al sistema, o rechazar que realizar proceso el cual indica al sistema que debe asignar a otro repartidor para dicho proceso.
- R5-4 Confirmación de proceso: Una vez asignado un servicio a algún usuario repartidor este debe confirmar pasos para terminar el proceso, confirmación de recepción del paquete y confirmación de entrega de paquete por medio de lector QR especificado en el punto R4-4. En la aplicación móvil se contará con un apartado para el escaneo de código QR.

3.1.3. Historias de usuario

Una vez definido los requerimientos que gozará el sistema en la siguiente etapa de desarrollo, se procederá a detallar las historias de usuario, con el fin de poder determinar un plan de entregas.

Es necesario contar con una administración de acceso al sistema, en la tabla 3.1 se especifica que usuario tiene acceso al sistema según un rol específico.

Tabla 3.1 Historia de usuario, acceso al sistema

| | |
|--|-----------------------|
| Nº: 1 | Usuario: Todos |
| Nombre de la historia: Acceso al sistema | |
| Programador responsable: Juan Carlos Choque Alavi | |
| Descripción: Según la definición una base de datos, se realiza el control de acceso al sistema de usuario, se determina roles para tener privilegios en el mismo, los roles son de súper administrador, administrador básico, repartidor, y cliente. R1-1, R1-2 | |
| Observaciones: Se debe tener usuario registrados en la base de datos | |

Fuente: Elaboración propia

En la tabla 3.2 se especifica el rol de administrador el cual realiza el registro de otros administradores, y también repartidores. En el caso de los clientes, ellos realizan su registro.

Tabla 3.2 Historia de usuario, registro de usuario

| | |
|--|-----------------------|
| Nº: 2 | Usuario: Todos |
| Nombre de la historia: Registro de usuarios | |
| Programador responsable: Juan Carlos Choque Alavi | |
| Descripción: Según el tipo de usuario se procede a realizar su registro Usuario Administrador y repartidor: El súper administrador se encarga del registro de personal de seguimiento usuario administrador básico y repartidor. El registro para cliente se realiza por cuenta propia, ósea se tendrá un formulario en el cual los clientes puedan acceder a él para su respectivo registro. Estos datos serán desplegados en un listado para clientes, personal repartidor, personal de administración. R2-1, R2-2, R3-1 | |
| Observaciones: Se debe tener usuario registrados en la base de datos | |

Fuente: Elaboración propia

Se tendrán las actualizaciones de datos del usuario también inhabilitarlos/habilitarlos, estos sucesos solo serán gestionados por el súper administrador como describe la tabla 3.3

Tabla 3.3 Historia de usuario, modificaciones y eliminación de usuario

| | |
|---|-----------------------|
| Nº: 3 | Usuario: Todos |
| Nombre de la historia: Modificaciones en usuarios | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Las modificaciones de datos, solo serán realizadas por el súper administrador, el mismo gestionara a los usuarios del sistema ya sea administradores adicionales, repartidores y clientes que se hayan registrado.</p> <p>Para realizar modificaciones en los usuarios, se desplegará un listado según al tipo de usuario ya sea administrador, personal repartidor o cliente, habilitando botones de acción. R2-3</p> | |
| Observaciones: Solo los administradores pueden realizar tales acciones mencionadas | |

Fuente: Elaboración propia

El cliente solicita de servicio para enviar paquetes, según la descripción de la tabla 3.4

Tabla 3.4 Historia de usuario, solicitud de servicio de mensajería de paquetes

| | |
|--|-------------------------|
| Nº: 4 | Usuario: Cliente |
| Nombre de la historia: Solicitud de servicio por el cliente | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Un cliente previamente registrado en el sistema, podrá realizar la solicitud para realizar envío de paquetes, para el mismo deberá registrar los siguientes datos: Dirección inicial o de origen del paquete especificando el nombre de la persona que entregara el paquete para su envío</p> <p>Dirección final o de recepción del paquete y la persona al cual se le entregara el paquete Ingresar datos descripción del paquete, selección del punto aproximado de la dirección de origen, seleccionar el punto destino o final en el mapa. R3-2, R3-3, R4-2</p> | |
| Observaciones: El cliente realizara peticiones para transporte de paquete | |

Fuente: Elaboración propia

Según la solicitud enviado por el cliente, el administrador gestionara dichas solicitudes, asignando al repartidor como también cancelado dicha solicitud por falta de personal disponible tabla 3.5

Tabla 3.5 Historia de usuario, confirmación de solicitud de servicio al cliente

| | |
|--|-------------------------------|
| Nº: 5 | Usuario: Administrador |
| Nombre de la historia: Confirmación de solicitud de servicio al cliente | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Una vez hecha la solicitud de envío de paquete, el cliente estará a la espera de la confirmación del servicio, ya que se estará iniciando el proceso de selección de repartidor disponible. Se desplegará una pantalla indicando que “Su solicitud está siendo atendida”, se contemplan tres casos:</p> <ul style="list-style-type: none"> - Solicitud cancelada por falta de personal disponible - Solicitud aceptada en el caso que su solicitud sea procesada <p>R3-2, R3-3, R4-2</p> | |
| <p>Observaciones: La confirmación de servicio será notificada de forma automática en la sesión abierta del usuario cliente</p> | |

Fuente: Elaboración propia

Una vez aceptada la solicitud, al cliente se le confirma este suceso, y los pasos a seguir según la tabla 3.6.

Tabla 3.6 Historia de usuario, solicitud aceptada

| | |
|--|-------------------------|
| Nº: 6 | Usuario: Cliente |
| Nombre de la historia: Solicitud aceptada | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Al usuario se le habilitará la opción de detalles respecto a su solicitud aceptada, en la cual indicará el desarrollo del proceso respecto a su solicitud aceptada la misma destalla un listado de sucesos</p> <ul style="list-style-type: none"> - Generación de QR - Detalle de la persona que atenderá el proceso - Paquete recepcionado - Paquete entregado <p>Desde la generación de un código QR, recepción y posterior entrega del paquete a la dirección remitida.</p> | |

| |
|---|
| <p>- Paquete recepcionado: el repartidor confirma que ya fue recepcionado el paquete al punto indicado por el cliente, se escanea el código QR para dicha verificación.</p> <p>Así mismo el cliente contara con un historial de las solicitudes que haya realizado.</p> <p>- Paquete entregado, el repartidor confirma que ya fue entregado el paquete del cliente a su dirección del remitente, en este caso finalizara el proceso de envío del paquete.</p> <p>R3-2, R3-3, R4-2</p> <p>Observaciones: El cliente realizara el seguimiento del estado del paquete</p> |
|---|

Fuente: Elaboración propia

El administrador del sistema contara con el despliegue de datos referente a la ubicación del repartidor como también si disponibilidad según tabla 3.7.

Tabla 3.7 Historia de usuario, localización y estados del repertidor en el mapa

| | |
|--|---|
| Nº: 7 | Usuario: Administrador, repartidor |
| Nombre de la historia: Localización y estados del repertidor en el mapa | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Se realiza el despliegue de los repartidores en el mapa.</p> <p>Se tienen los estados de:</p> <ul style="list-style-type: none"> - Disponible, no tiene paquetes en proceso - Ocupado, se está atendiendo una petición de transporte de paquete - Sin conexión, pérdida de conexión al sistema <p>R4-1, R4-2, R3-3</p> <p>Observaciones: El usuario administrador gestiona a los repartidores con disponibilidad según el mapa que muestra el estado actual de los repartidores</p> | |

Fuente: Elaboración propia

Se tiene un listado de solicitudes para ser procesados por el administrador del sistema, estas solicitudes cuentan con información del paquete que desea enviar el cliente según la descripción de la tabla 3.8.

Tabla 3.8 Historia de usuario, aceptación de solicitud de transporte de paquete

| | |
|---|-------------------------------|
| N°: 8 | Usuario: Administrador |
| Nombre de la historia: Aceptación de solicitud de transporte de paquete | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Se listará todas las solicitudes de transporte de paquetes de las cuales se debe desplegar la información que envió el cliente en la pantalla de seguimiento de personal repartidor.</p> <p>Según se selecciones una solicitud esta dará a conocer detalles de punto de recepción, lugar de entrega, contenido del paquete a transportar, detalle de la persona que está realizando él envió y a que persona va dirigida.</p> <p>Para confirmar si solicitud será procesada el administrador selecciona un repartidor con disponibilidad, al repartidor se le envía un aviso a su dispositivo móvil, el repartidor acepta la solicitud la misma que llega como notificación al sistema. R4-1, R4-2, R3-3</p> | |
| Observaciones: El usuario administrador gestiona las solicitudes de envió | |

Fuente: Elaboración propia

El repartidor enviará datos de su estado del paquete al sistema, el administrador tendrá la información sobre cambios de estado, se detalla el mismo en la tabla 3.9.

Tabla 3.9 Historia de usuario, seguimiento del paquete

| | |
|---|-------------------------------|
| N°: 9 | Usuario: Administrador |
| Nombre de la historia: Seguimiento del paquete | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: El sistema recepciona datos de localización del repartidor y por ende el estado de paquete en caso de ser asignado un servicio de envió, estos datos se reflejarán en el mapa. Se tendrá estados con los cuales identificamos si el repartidor ya recolecto el paquete, si ya lo entrego o aún está en camino para su destino final.</p> <p>En el sistema mostrara como mensajes de entrada el estado del paquete, desde la aceptación de solicitud hasta la entrega del mismo al destinatario, notificando cada estado según lo anterior descripción. R4-1, R4-2, R4-3</p> | |
| Observaciones: El usuario administrador se le notifica automáticamente los estados del paquete | |

Fuente: Elaboración propia

El personal de repartidor se le proporcionara una aplicación móvil con la cual contara con formulario de acceso ello se describe en la siguiente tabla 3.10.

Tabla 3.10 Historia de usuario, acceso al sistema por el repartidor

| | |
|--|----------------------------|
| N°: 10 | Usuario: Repartidor |
| Nombre de la historia: Acceso al sistema por el repartidor | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Para el personal repartidor se tendrá un pequeño aplicativo móvil. Debe contar con un formulario de inicio de sesión contado como usuario su correo electrónico y su respectiva contraseña.</p> <p>Si sus credenciales son correctas se inicia sesión desde la aplicación móvil Si sus credenciales son incorrectas no debe permitir el ingreso o inicio de sesión. R4-1, R4-2, R5-1, R5-2, R5-3, R54</p> | |
| Observaciones: Se cuenta con un formulario de acceso propiamente para el repartidor | |

Fuente: Elaboración propia

Los usuarios repartidores enviaron información respecto a su estado, desde que se conectan con el sistema, su localización, si se encuentran disponibles, tabla 3.11.

Tabla 3.11 Historia de usuario, envió de locación del repartidor

| | |
|---|----------------------------|
| N°: 11 | Usuario: Repartidor |
| Nombre de la historia: Envío de localización del repartidor | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Una vez iniciado sesión el repartidor debe enviar su localización si inicia su horario laboral, una vez acabado el día laboral debe cancelar el envío de localización, de esta forma se realizará el seguimiento del usuario repartidor ya sea que este libre o con una solicitud de envío de paquete en proceso</p> <p>El repartidor enviara datos de su localización por medio de latitud y longitud activando su GPS desde el aplicativo hasta la finalización de su hora laboral desactivando su localización. R4-1, R4-2, R5-1, R5-2, R5-3, R54</p> | |
| Observaciones: La latitud y longitud de los repartidores serán desplegados en un mapa dando a conocer su ubicación precisa | |

Fuente: Elaboración propia

Los usuarios repartidores se les notifica cuando se les asigna una nueva solicitud de transporte de paquete se describe en la tabla 3.12.

Tabla 3.12 Historia de usuario, notificación de envío de paquete al repartidor

| | |
|--|----------------------------|
| N°: 12 | Usuario: Repartidor |
| Nombre de la historia: Notificación repartidor y sistema | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: Desde el sistema, el administrador le designará un nuevo trabajo de envío de paquete al personal repartidor por medio de notificación push a la aplicación móvil.</p> <p>Dicha notificación contendrá el detalle de la tarea a ser procesada, el repartidor confirmará si podrá realizar dicha tarea. R4-1, R4-2, R5-1, R5-2, R5-3, R54</p> | |
| Observaciones: Se debe generar tokens para él envío de notificaciones | |

Fuente: Elaboración propia

Los repartidores deben realizar el escaneo del código QR generado que ira pegado al paquete para el control del estado del paquete que se les asigno y que aceptaron procesarlo, tabla 3.13.

Tabla 3.13 Historia de usuario, cambios de estado del paquete

| | |
|---|----------------------------|
| N°: 13 | Usuario: Repartidor |
| Nombre de la historia: Cambios de estado del paquete | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: En caso que el personal repartidor confirmara la solicitud de envío de paquete, se le proporcionara datos de dicho paquete</p> <p>Una vez que el repartidor recolecte el paquete este debe confirmar dicho suceso. Una vez que haya llegado a su destino final y entregado el paquete, el repartidor deberá dar a conocer tal suceso y se dará por concluido él envío de paquete. R4-1, R4-2, R5-1, R5-2, R5-3, R54</p> | |
| Observaciones: El usuario administrador realiza el seguimiento de forma global del personal como de los paquetes | |

Fuente: Elaboración propia

Para la generación de reportes tanto el súper administrador y administrador básico que pueden realizar tal acción, las mismas se describe en la siguiente tabla 3.14.

Tabla 3.14 Historia de usuario, generación de reportes

| | |
|---|---------------------------------|
| Nº: 14 | Usuario: Administradores |
| Nombre de la historia: Generación de reportes | |
| Programador responsable: Juan Carlos Choque Alavi | |
| <p>Descripción: La generación de reportes será realizada por un usuario con privilegios de administrador, el reporte refleja los casos de transportes realizados por el personal encargado.</p> <p>Se desplegará un detalle del donde se tendrá a los clientes ya sea de empresa o personas que hayan solicitado el servicio, a la vez se tendrá asociado el nombre del personal que atendió tal solicitud, se tendrá el costo asignado.</p> <p>El reporte será cargado en un listado general del cual se realizar un filtrado por fechas y a la vez un filtrado de coincidencias por cualquier campo que se desee buscar, una vez que el usuario Administrador tenga la información requerida este podrá generar un archivo en Excel o una impresión directa del reporte deseado.</p> R2-4 | |
| Observaciones: El usuario debe ser administrador | |

Fuente: Elaboración propia

Diseño de Base de Datos

Utilizando el modelo de desarrollo UWE se requiere que dentro de la implementación se ejecute el diseño de la base de datos, puede que la base de datos sufra cambios dependiendo de los cambios que se realice durante el desarrollo del sistema.

a) Diagrama Entidad Relación

En la figura 3.1 se puede observar el diagrama entidad relación que se ejecutó con los requerimientos obtenidos anteriormente.

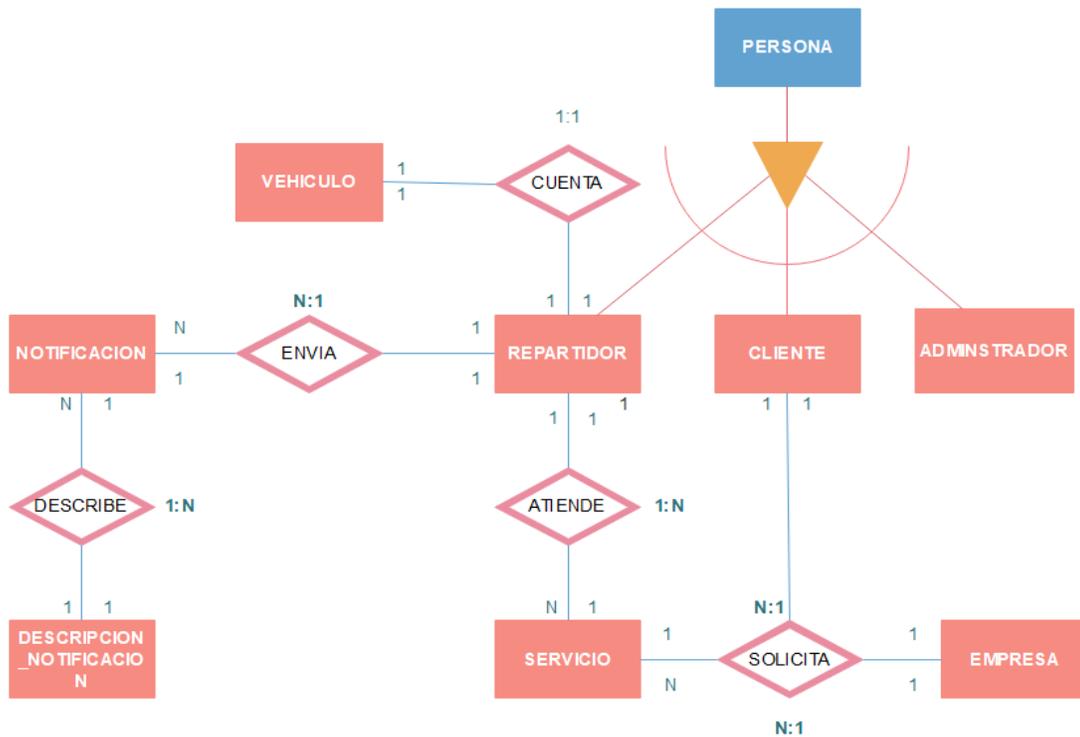


Figura 3.1 Diagrama entidad relación
Fuente: Elaboración propia

b) Modelo Relacional de la Base de Datos

Persona(**persona_id**, nombre, primer_apellido, segundo_apellido, teléfono, dirección)

Repartidor(**persona_id**, token, imagen, **vehiculo_id**, numero_placa, marca, modelo, estado, latitud, longitud, fecha_hora_envio_ubicacion)

Administrador(**persona_id**, contraseña, estado)

Cliente(**persona_id**, teléfono, estado, contraseña)

Descripcion_Notificacion(**descripcion_id**, descripcion_notificacion, advertencia, color)

Notificacion(**notificación_id**, **repartidor_id**, **servicio_id**, descripcion, fecha_hora_notificacion, estado)

Servicio(**servicio_id**, latitud_inicial, longitud_inicial, descripción_paquete, fecha_hora_solicitud, **cliente_id**, estado_servicio, **repartidor_id**, latitud_final, longitud_final, hora_recoleccion_paquete, hora_entrega_paquete, descripcion_lugar_recoleccion, descripcion_lugar_entrega, costo_envio, tipo)

3.2. Primera Iteración (Sprint1)

En esta etapa se considera importante el diseño de la base de datos, definir las entidades según el requerimiento según los requerimientos que presenta el Product Owner. Una vez definido e implementado la base de datos, se desarrollará el control de acceso al sistema, la gestión de personal, clientes. En la tabla 3.15 se puede observar la implementación de las fases que se realizarán con su tiempo de duración para cada tarea, así mismo se puede señalar que se está utilizando las fases principales de la metodología UWE.

Tabla 3.15 Detalle de tareas a realizar en el Sprint 1

| Primera Iteración | | Sprint | | |
|-------------------|--|------------|-----------------|-----------|
| | | 1 (uno) | | |
| ID | Tareas | Tipo | Días de Trabajo | Estado |
| 1.1 | Análisis para la gestión de usuarios | Análisis | 2 | Terminado |
| 1.2 | Análisis para la autenticación de usuarios | Análisis | 1 | Terminado |
| 1.3 | Diseñar los marcos de la aplicación web de gestión cabeceras menú superior, menú lateral | Diseño | 1 | Terminado |
| 1.4 | Diseñar los marcos de la aplicación web del lado del cliente para la gestión cabeceras menú superior | Diseño | 1 | Terminado |
| 1.5 | Diseñar el Registro de usuarios (administradores, repartidores, clientes y empresas) | Diseño | 1 | Terminado |
| 1.6 | Diseñar las modificaciones en usuarios (administradores, repartidores, clientes) | Diseño | 1 | Terminado |
| 1.7 | Diseñar formularios de autenticación | Diseño | 1 | Terminado |
| 1.8 | Desarrollar el proceso de autenticación de usuario | Desarrollo | 1 | Terminado |
| 1.9 | Desarrollar el módulo de registro y modificaciones de usuario administrador | Desarrollo | 2 | Terminado |
| 1.10 | Desarrollar el módulo de registro y modificaciones de usuario repartidor | Desarrollo | 2 | Terminado |
| 1.11 | Desarrollar el registro para clientes y empresas | Desarrollo | 2 | Terminado |
| 1.12 | Desarrollar el módulo modificaciones de usuario cliente y empresas | Desarrollo | 3 | Terminado |

Fuente: Elaboración propia

a) Análisis del Sprint 1

Modelo Lógico Conceptual

Se tiene en la figura 3.2 la administración de usuarios desde su creación como modificación e inhabilitar y habilitación.

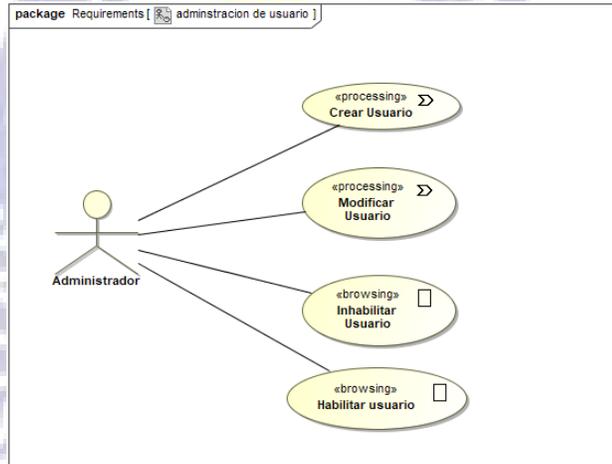


Figura 3.2 Diagrama de caso de uso – Administración de usuarios
Fuente: Elaboración propia

En la figura 3.3 el registro de usuario en este caso de los clientes que quieran formar parte del sistema, se puede observar a la vez, que el registro es una extensión de la creación de usuario.

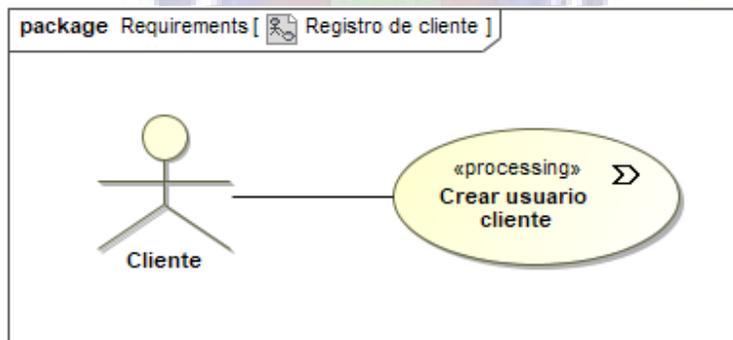


Figura 3.3 Diagrama de caso de uso – Registro de cliente
Fuente: Elaboración propia

El diagrama de autenticación se muestra en la figura 3.4 especifica la validación de usuario que contenga una cuenta activa en el sistema.

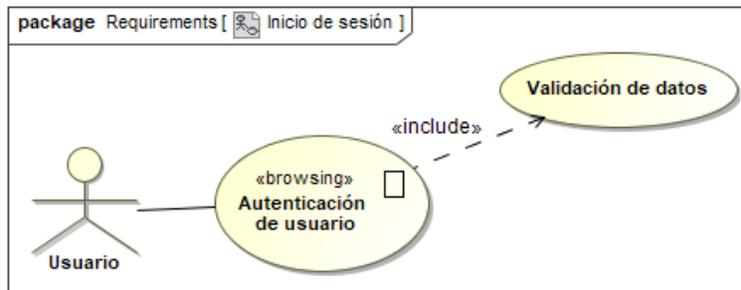


Figura 3.4 Diagrama de caso de uso – Autenticación de usuario
Fuente: Elaboración propia

b) Modelo Navegacional de Administración de Usuario

A continuación, en la figura 3.5 se tiene el modelo navegacional de administración de usuarios, el cual tiene la responsabilidad del súper administrador para realizar modificaciones en caso que sea necesario, otro tipo de administrador no tiene tales privilegios

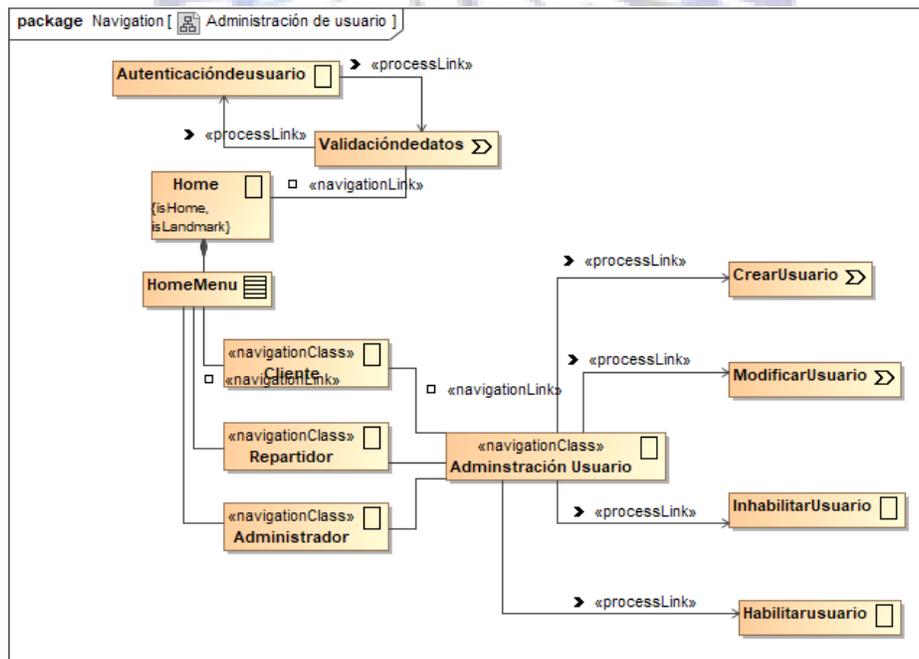


Figura 3.5 Diagrama de navegación – Administración de usuarios
Fuente: Elaboración propia

c) Modelo de Presentación

En la figura 3.6 contamos con parte de la definición del sistema para el actual sprint, posteriormente se complementará esta definición con otras funcionalidades requeridas.

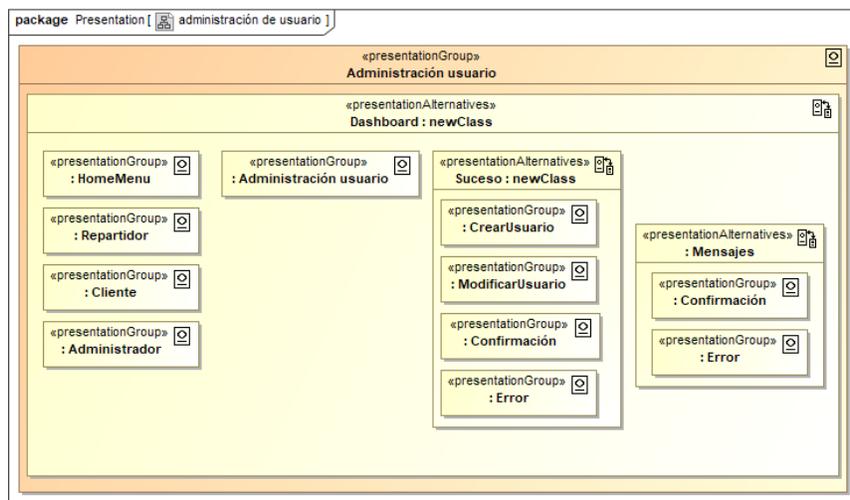


Figura 3.6 Diagrama de presentación – Administración de usuarios
Fuente: Elaboración propia

d) Escenario Web

En las figuras 3.7 y 3.8 tenemos los formularios de autenticación de usuario administrador y cliente en la primera figura y en la segunda figura se tiene la opción de registro de usuario cliente.

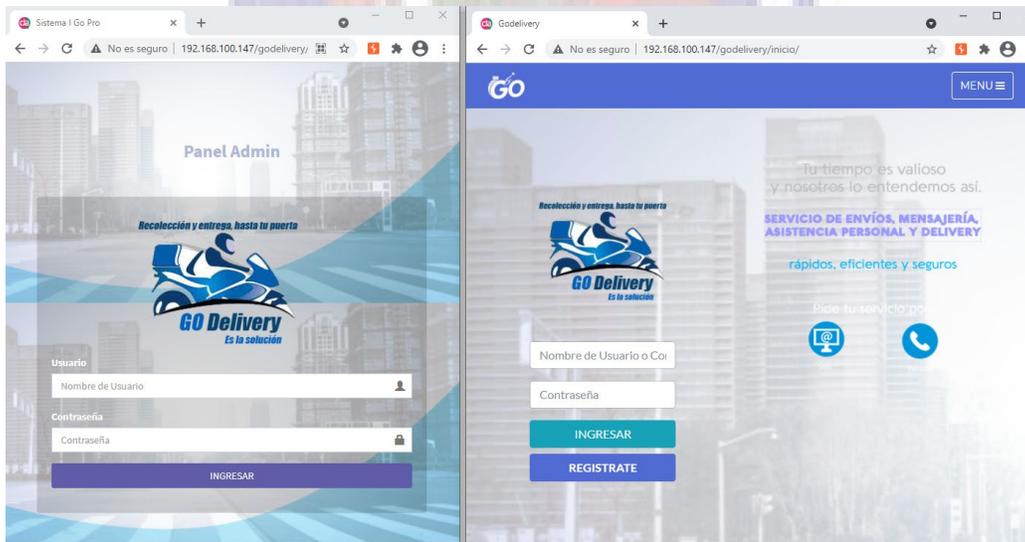


Figura 3.7 Autenticación de usuarios
Fuente: Elaboración propia

Figura 3.8 Registro para clientes y/o empresas
Fuente: Elaboración propia

Se detallará a continuación el proceso de la administración de las cuentas de usuario, en la figura 3.9 tenemos como primera pantalla tres opciones para administrar usuarios, a partir de allí accederemos a estas opciones, administradores, repartidores y clientes.

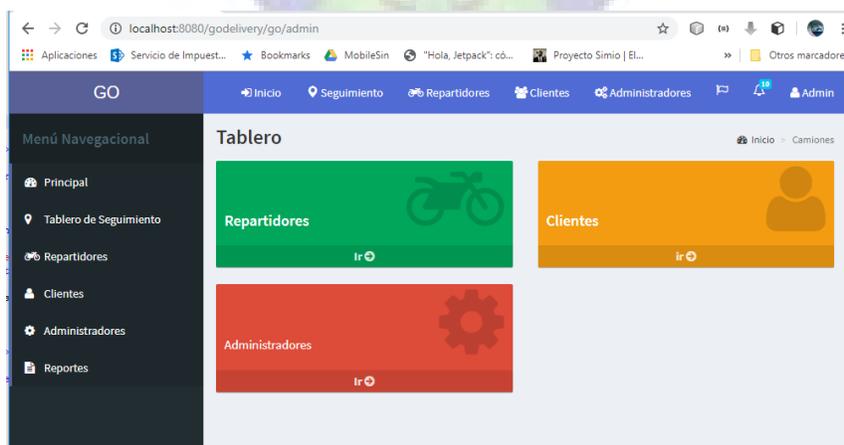


Figura 3.9 Opciones para la administración de usuarios
Fuente: Elaboración propia

Ingresamos a la opción para gestionar a los administradores como se muestra en la figura 3.10, se lista todos los administradores activos, se presenta también dos tipos de administradores un súper administrador que tiene los privilegios de modificar datos, y el administrador básico este último no cuenta con la posibilidad de modificar datos en el sistema.

| Nombre | Dirección | Telefono/celular | Tipo | Cambiar | Inactivo |
|-----------------|--|------------------|---------------------|---------|----------|
| admin | Av. Siete Enanos | 22010701 | Super administrador | | |
| Alberto Barros | Av. Ramos Gavilán Zona Achachicala Nro 1094, Calle 7 | 78864971 | Normal | | |
| Javier77 | Simpre viv | 789456 | Normal | | |
| Pablo Jaramillo | Av. Ramos | 60594586 | Normal | | |

Figura 3.10 Gestión de administradores
Fuente: Elaboración propia

Se tiene la opción de adicionar Nuevos Administradores, en la figura 3.11 tenemos el mencionado proceso, del cual todos los campos de entrada son obligatorios, se da a conocer que como primera contraseña se tiene al número de CI, que posteriormente el sistema pedirá que realice el cambio del mismo.

| | |
|---------------------------|----------------------------|
| Nombre*: | Nombre de Usuario |
| CI*: | Carnet de Identidad |
| Telefono/Celular*: | Número de telefono/celular |
| Dirección*: | Dirección Av/Calle/Nro |
| Tipo*: | Admin Normal |

Nota: La contraseña del nuevo administrador será su carnet de identidad, el usuario debe actualizarlo.

Registrar Cerrar

Figura 3.11 Gestión de administradores – Registrar nuevo administrador
Fuente: Elaboración propia

La opción que nos permite modificar se encuentra en el listado de usuario administradores desplegando una ventana que tiene por defecto los campos a modificar como se puede observar a continuación en la figura 3.12.

| | |
|---------------------------|--------------------------|
| Nombre*: | Alberto Barros |
| CI*: | 6126333 |
| Telefono/Celular*: | 78864971 |
| Dirección*: | Av. Ramos Gavilan Zona A |
| Tipo*: | Admin Normal |

Modificar Cerrar

Figura 3.12 Gestión de administradores – Modificar usuario administrador
Fuente: Elaboración propia

El sistema no te permitirá eliminar usuarios, en cambio si se tiene la opción de inhabilitar a los usuarios, que posteriormente si se da el caso se los vuelven a habilitar, se presenta una advertencia si se desea inhabilitar algún usuario como se tiene en la figura 3.13.

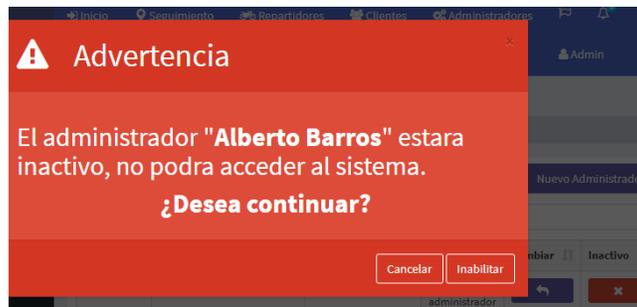


Figura 3.13 Gestión de administradores – Inhabilitar usuario administrador
Fuente: Elaboración propia

En la opción de usuarios repartidores que se puede observar en la figura 3.14, se desplegará el listado de usuarios activos, donde además de adicionar nuevo usuario, modificarlo e inhabilitarlo, se cuenta con el detalle de la información de dicho personal o repartidor expuesto en la figura 3.15.

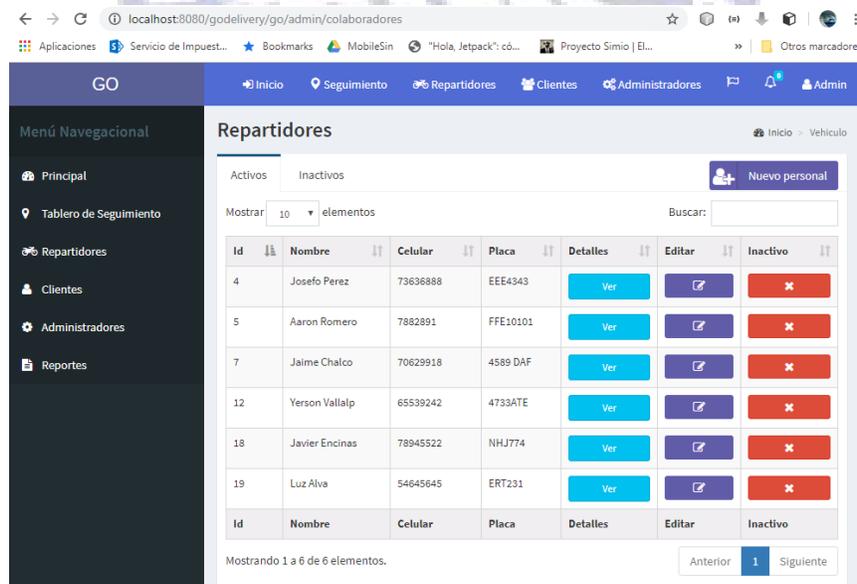


Figura 3.14 Gestión de usuarios repartidores
Fuente: Elaboración propia



Figura 3.15 Gestión de usuarios repartidores – Detalles de personal repartidor
Fuente: Elaboración propia

A continuación, se tiene en la figura 3.16 listado de empresas registradas, como también en la figura 3.17 se tiene los clientes registrados. Los procesos de creación, modificación, detalles e inactivar, son procesos similares a los que presentamos anteriormente.

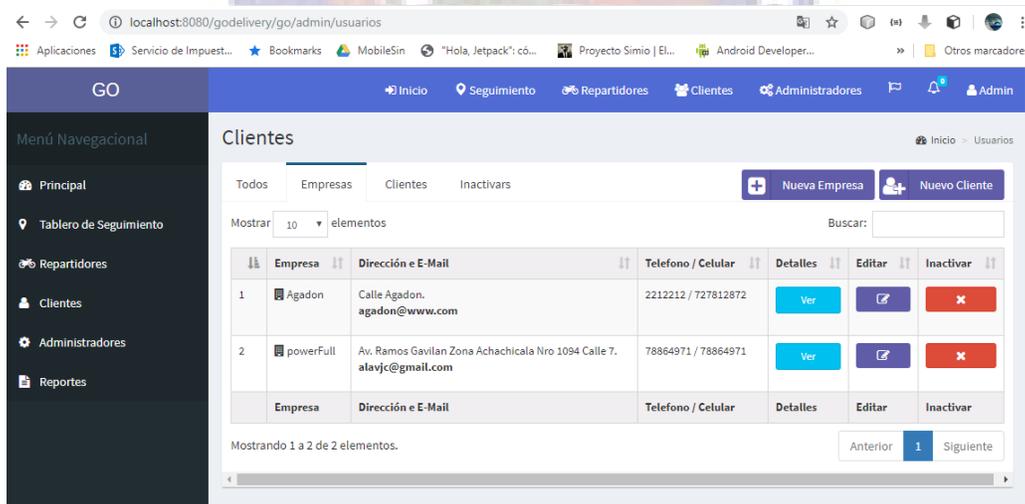


Figura 3.16 Gestión de empresas
Fuente: Elaboración propia

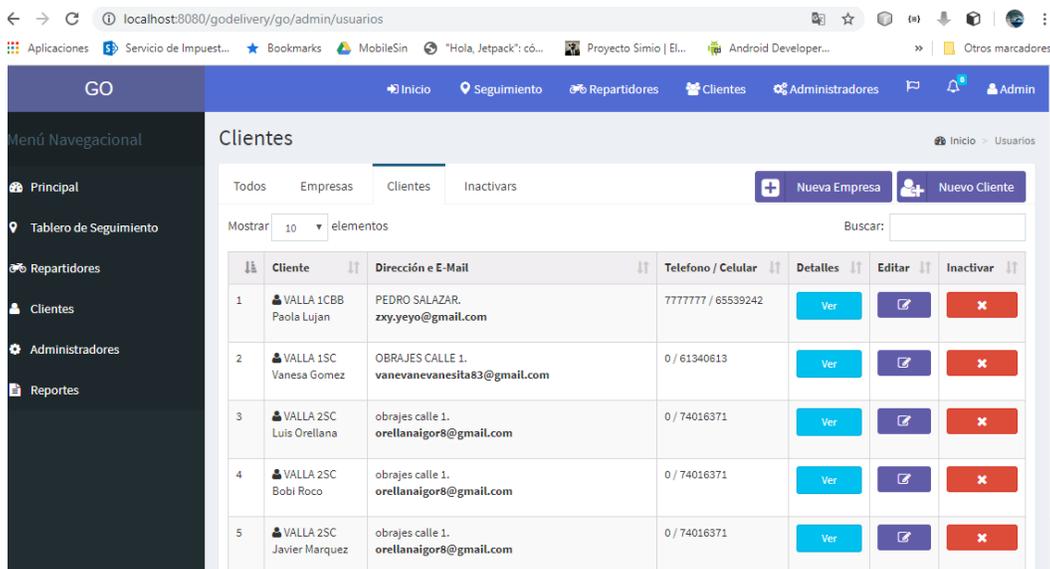


Figura 3.17 Gestión de clientes
Fuente: Elaboración propia

e) Revisión de Sprint1

En esta etapa se revisó el cumplimiento de las tareas planificadas en el Sprint

Las pruebas de funcionalidad para el registro de nuevos usuarios se muestran en la tabla 3.16, esta prueba se aplica tanto para administradores y repartidores.

Tabla 3.16 Prueba de registro de usuarios

| | |
|---|-------------------------------------|
| Prueba 1.1 | Operación: Registro usuarios |
| Precondición: | |
| <ul style="list-style-type: none"> • Sesión con el Rol correcto (súper administrador) | |
| Datos/Proceso | |
| <ul style="list-style-type: none"> • Del menú superior o menú lateral seleccionar la opción “Administradores” • Pulsar sobre el botón “Nuevo Administrador, ” | |
| Resultados esperados: | |
| Se registrara al nuevo usuario con un rol específico, es necesario llenar todos los datos que presenta el formulario | |
| Postcondición: | |
| Registro exitoso | |
| Resultados obtenidos: | |
| Registro exitoso del usuario administrador | |

Fuente: Elaboración propia

En la tabla 3.17 se tiene la prueba de funcionalidad para la modificación de usuarios que es aplicada para Administradores, Repartidores, Clientes y Empresas

Tabla 3.17 Prueba de modificación de usuarios

| | |
|--|--|
| Prueba 1.2 | Operación: Modificaciones de usuarios |
| Precondición: | |
| <ul style="list-style-type: none"> • Sesión con el Rol correcto (súper administrador) | |
| Datos/Proceso | |
| <ul style="list-style-type: none"> • El administrador que se encuentra dentro del sistema e ingresado a alguna opción de usuario, se le presentara un listado de usuarios activos donde se puede notar la opción de modificación por cada usuario listado • Pulsar sobre el botón “Modificar ” | |
| Resultados esperados: | |
| Según al campo que desea modificar será guardada | |
| Postcondición: | |
| Modificación exitosa | |
| Resultados obtenidos: | |
| Modificar datos del usuario | |

Fuente: Elaboración propia

Para la funcionalidad de autenticación de usuario se describe la prueba de aceptación en la tabla 3.18

Tabla 3.18 Prueba de autenticación de usuarios

| | |
|--|--|
| Prueba 1.3 | Operación: Autenticación de usuario |
| Precondición: | |
| <ul style="list-style-type: none"> • Usuario registra y activo en el sistema | |
| Datos/Proceso | |
| <ul style="list-style-type: none"> • Para cualquier usuario, debe contar con un nombre identificativo de usuario y su contraseña • Presionar el botón ingresar, si es correcto el usuario iniciara sesión en el sistema • De ser incorrecto los campo usuario o contraseña, se desplegara un mensaje de error indicando que algún dato ingresado no es correcto | |
| Resultados esperados: | |
| <ul style="list-style-type: none"> • Inicio de sesión • Error al ingresar datos de usuario | |

| |
|---|
| Postcondición: Inicio de sesión |
| Resultados obtenidos: <ul style="list-style-type: none"> • Sesión inicializada con éxito • Despliegue de error al ingresar datos |

Fuente: Elaboración propia

3.3. Segunda Iteración (Sprint 2)

Para la realización de la segunda iteración, se procedió a utilizar la metodología UWE con sus respectivas fases, como se observa en la tabla 3.19.

Tabla 3.19 Detalle de tareas a realizar en el Sprint 2

| Primera Iteración | | Sprint | | |
|-------------------|--|------------|-----------------|-----------|
| | | 2 (dos) | | |
| ID | Tareas | Tipo | Días de Trabajo | Estado |
| 2.1 | Análisis del proceso de solicitud de servicio y seguimiento del proceso Análisis del desplazamiento de repartidor | Análisis | 1 | Terminado |
| 2.3 | Diseño del módulo de seguimiento (componentes a usar: mapa, notificaciones de nuevas peticiones de transporte de paquete, despliegue del detalle de paquete, notificaciones de estado) | Diseño | 2 | Terminado |
| 2.4 | Diseñar las opciones en la app móvil (autenticación, habilitar ubicación, listado de solicitudes, confirmación de solicitudes, cambios de estados referente al paquete) | Diseño | 2 | Terminado |
| 2.5 | Desarrollo del registro de solicitud de servicio para un nuevo paquete del cliente y registro en base de datos del sistema | Desarrollo | 1 | Terminado |
| 2.6 | Gestión y despliegue de los datos de ubicación en el mapa del módulo de seguimiento de todos los usuarios repartidores activos | Desarrollo | 1 | Terminado |
| 2.7 | Desarrollar la función donde se recupera la solicitud del cliente para enviar un paquete, donde se mostrara un listado de solicitudes de los cuales se podrán ver su detalle cuando se presiona en alguna solicitud listada y asignar el | Desarrollo | 2 | Terminado |

| | | | | |
|------|--|------------|---|-----------|
| | paquete a un determinado repartidor que se encuentre libre mediante notificaciones push | | | |
| 2.8 | Desarrollar servicios para el consumo del aplicativo móvil según el siguiente listado: <ul style="list-style-type: none"> • Autenticación de usuario repartidor • Envío de latitud y longitud para registrarlo en base de datos del sistema • Solicitud para obtener el detalle de paquete • Envío de estado del paquete (Confirmar solicitud, paquete recepcionado y paquete entregado) | Desarrollo | 2 | Terminado |
| 2.9 | Consumo de servicio autenticación para la app móvil que usara el repartidor Configurar el uso de GPS del dispositivo móvil para capturar la latitud y longitud, consumir el servicio para recepción de los datos de latitud y longitud | Desarrollo | 2 | Terminado |
| 2.10 | Listar las solicitudes de asignación de paquetes que fueron enviadas por notificación push Consumir el servicio para obtener el detalle del paquete a trasladar Consumir el servicio para realizar cambios de estado | Desarrollo | 1 | Terminado |

Fuente: Elaboración propia

a) Análisis del Sprint 2

Modelo Lógico Conceptual

Se resalta el proceso del flujo importante del sistema, y la interacción entre los actores involucrados del sistema.

En la figura 3.18 se resalta la solicitud del servicio el cual deriva a otras funcionalidades del sistema, como ser notificaciones, estas llegan al administrador el cual deriva la petición de servicio a un repartidor disponible, este último genera cambios de estado que son interpretados como notificaciones tanto para el cliente como para el administrador que realiza el seguimiento del proceso.

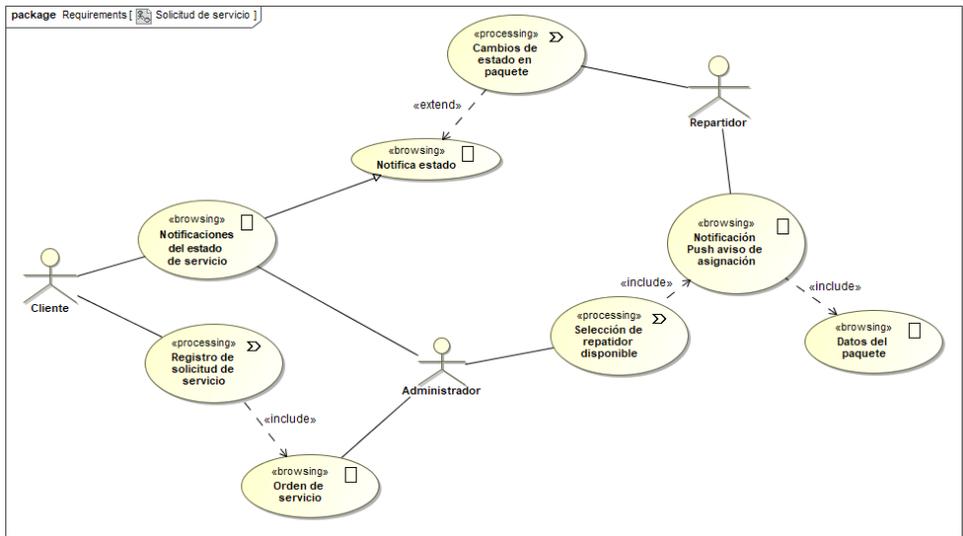


Figura 3.18 Diagrama de caso de uso – Solicitud de servicio
Fuente: Elaboración propia

Para tener el control del seguimiento de repartidores, el administrador debe ingresar al caso de seguimiento el cual realiza peticiones sobre los cambios de ubicación que envía el repartidor desde su dispositivo, como se muestra en la figura 3.19.

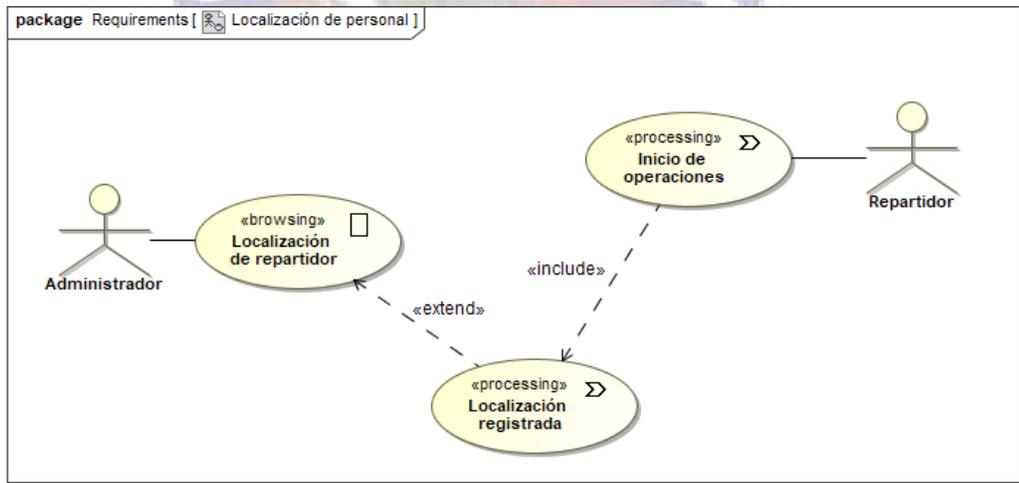


Figura 3.19 Diagrama de caso de uso – Localización de personal
Fuente: Elaboración propia

A continuación, en la figura 3.20 se tiene la interacción que realiza el repartidor a los cambios de estado del paquete durante el proceso de envío de paquete.

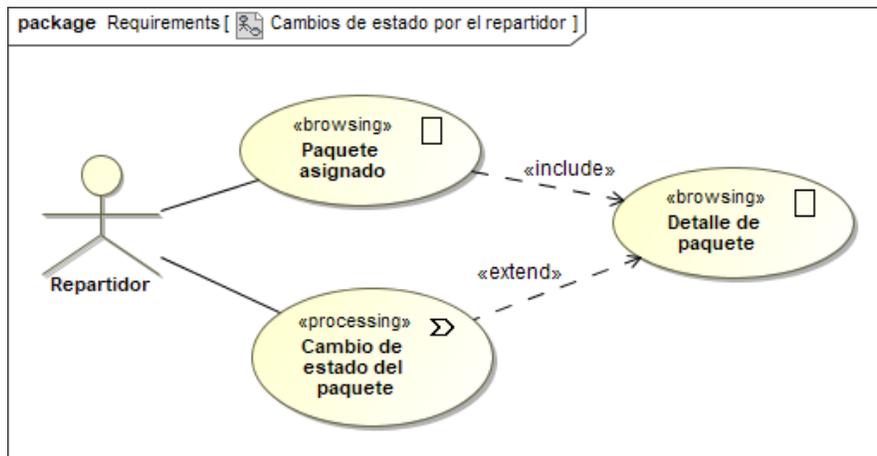


Figura 3.20 Diagrama de caso de uso – Cambios de estado por el repartidor
Fuente: Elaboración propia

b) Modelo Navigacional

Se tiene un esquema de navegación interactuando con los usuarios que intervienen en el sistema, como se puede observar en la figura 3.21.

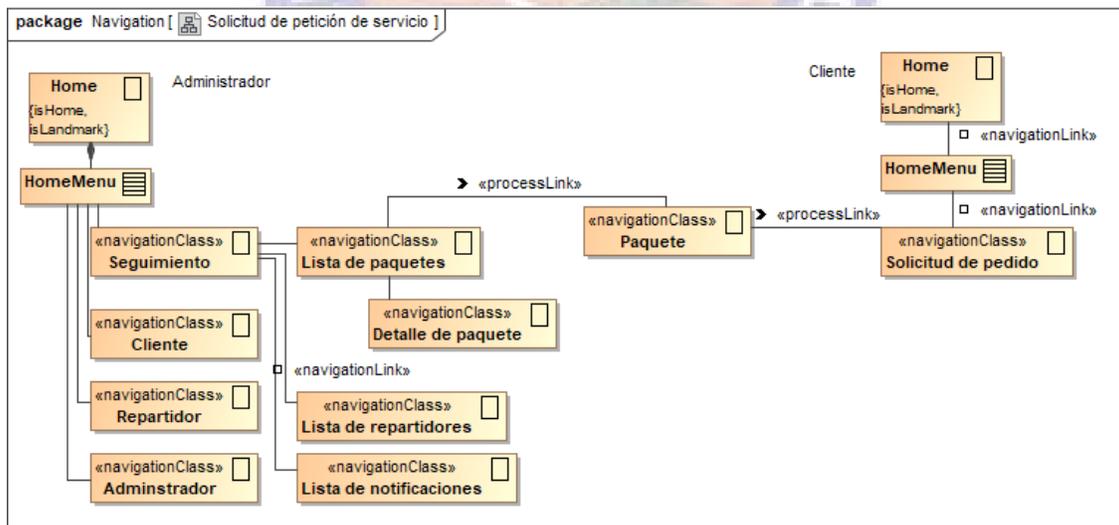


Figura 3.21 Diagrama de navegación – Solicitud de petición de servicio
Fuente: Elaboración propia

El usuario repartidor interactuara con la aplicación móvil, en la figura 3.22 se tiene las opciones con las cuales tratara este usuario.

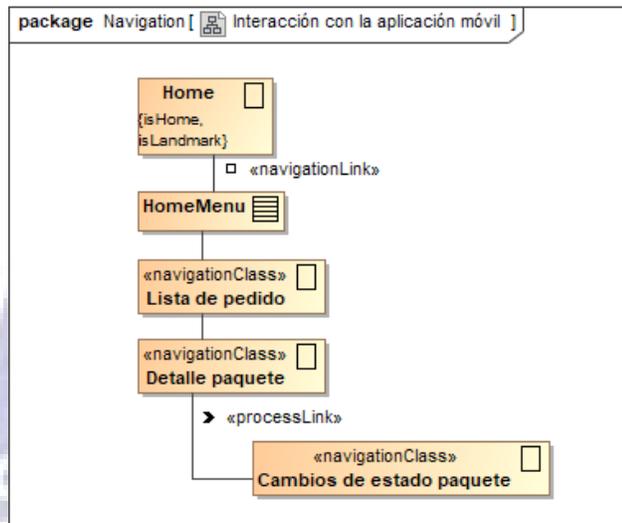


Figura 3.22 Diagrama de navegación – Interacción con la aplicación móvil
Fuente: Elaboración propia

c) Modelo de Presentación

Describimos a continuación en la figura 3.23 los componentes que contiene el tablero de seguimiento.



Figura 3.23 Diagrama de presentación – Tablero de seguimiento
Fuente: Elaboración propia

d) Escenario Web

En la figura 3.24 se tiene la solicitud de servicio desde la cuenta del cliente detallando los datos del paquete a ser transportado, también puntos de localización de recolección y entrega capturados desde el mapa, todos estos datos serán proporcionados al repartidor asignado.

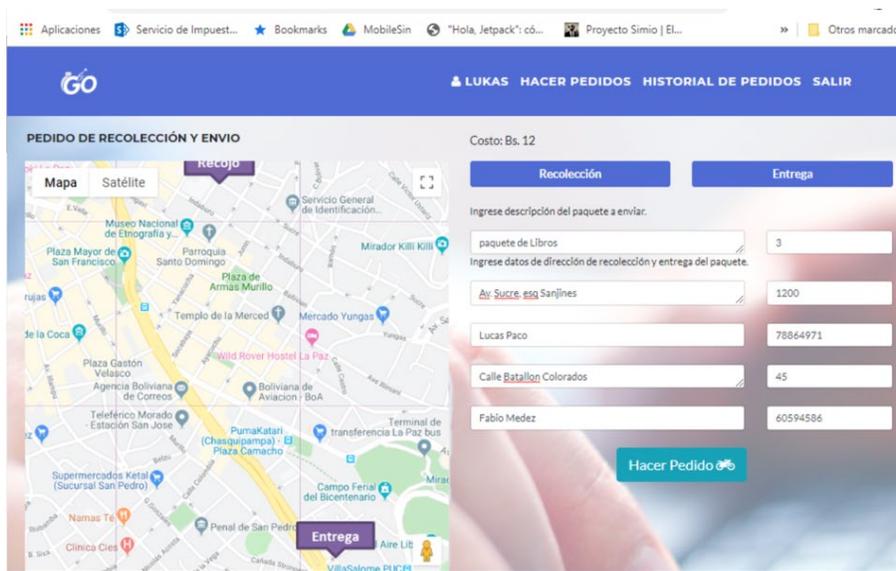


Figura 3.24 Solicitud de servicio
Fuente: Elaboración propia

En la siguiente figura 3.25 se puede observar en la cuenta del administrador el despliegue de la notificación de pedido del cliente en el lado izquierdo y en el derecho el despliegue de los datos del paquete, también se tiene desplegado en el mapa la ubicación del repartidor.

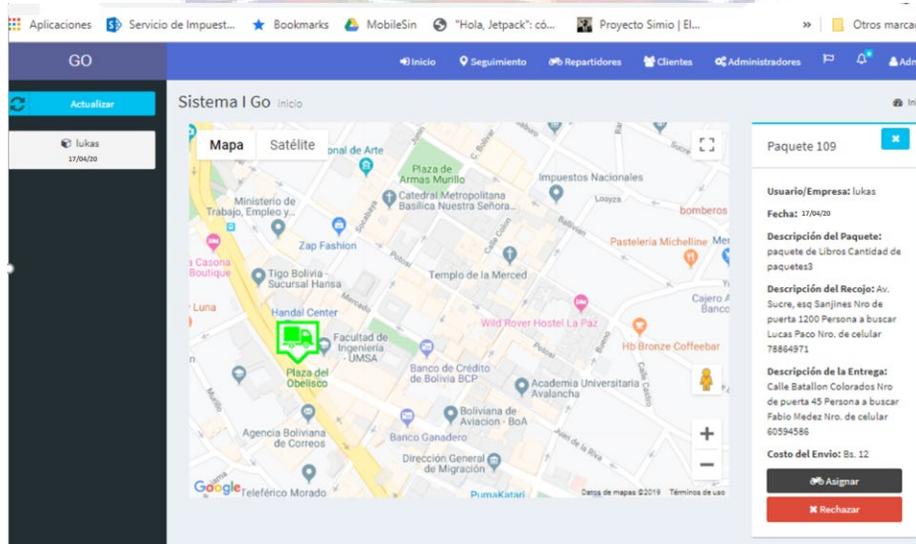


Figura 3.25 Despliegue de datos del paquete
Fuente: Elaboración propia

A continuación, se tiene la asignación del servicio de transporte de paquete al repartidor más cercano al punto de recolección figura 3.26.

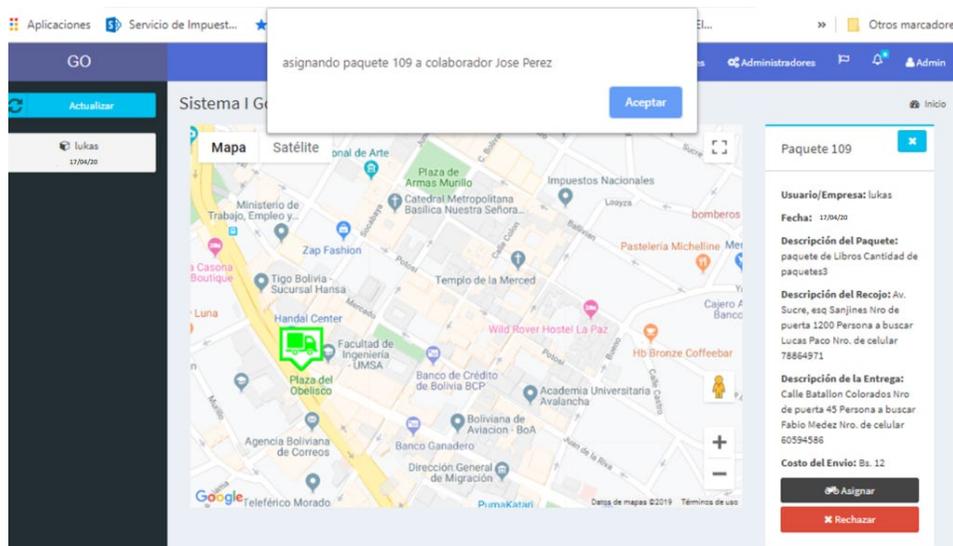


Figura 3.26 Asignación de paquete al repartidor
Fuente: Elaboración propia

El usuario repartidor debe iniciar sesión desde la aplicación móvil, para que le puedan asignar un paquete a ser transportado, una vez asignado le llega una notificación push a su dispositivo móvil como se observa en la figura 3.27.

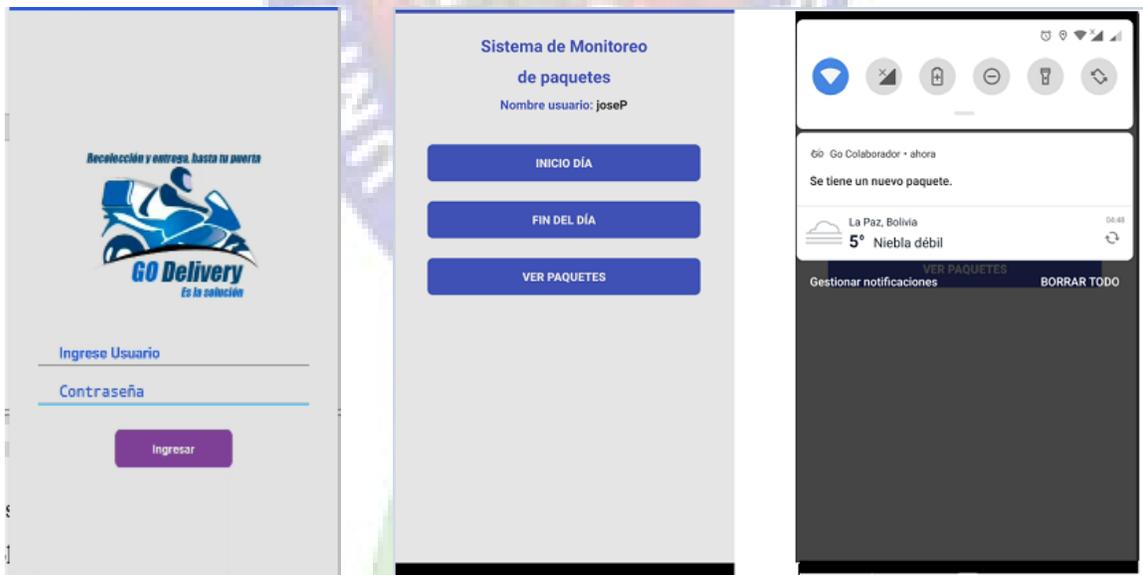


Figura 3.27 Aplicación móvil y notificación de asignación de paquete
Fuente: Elaboración propia

El repartidor podrá observar el listado de paquetes y a la vez tener el detalle del paquete asignado figura 3.28, en detalles se puede observar también que se cuenta con botones de

recepción y entrega de paquete, esta función nos despliega el punto de recolección y el punto de entrega como se muestra en la figura 3.29.

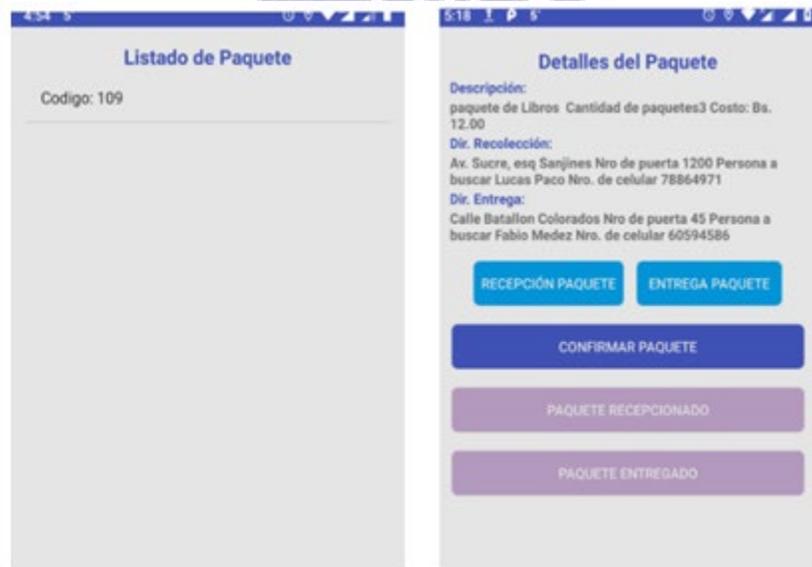


Figura 3.28 Aplicación móvil listado y detalle del paquete
Fuente: Elaboración propia

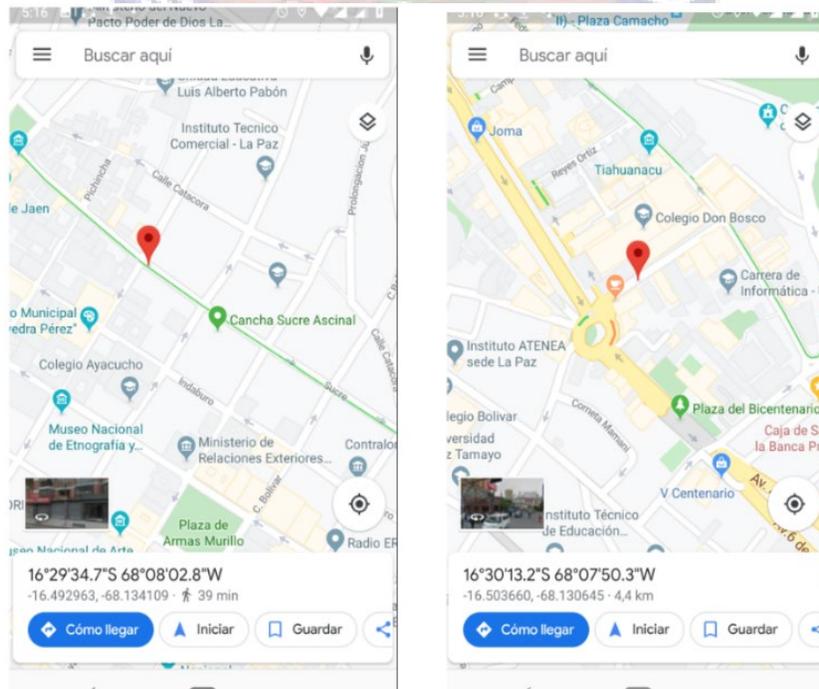


Figura 3.29 Aplicación móvil punto de recolección y entrega del paquete
Fuente: Elaboración propia

d) Revisión de Sprint 2

Para la funcionalidad de solicitud de servicio que el cliente realiza se describe la prueba de aceptación en la tabla 3.20.

Tabla 3.20 Prueba de solicitud de servicio del cliente

| Prueba 2.1 | Operación: Solicitud de servicio |
|---|----------------------------------|
| Precondición: <ul style="list-style-type: none">• Usuario registrado y activo en el sistema | |
| Datos/Proceso <ul style="list-style-type: none">• El usuario debe iniciar sesión• Ingresar a la opción de pedidos• Llenar todos los datos respecto a la solicitud de servicio transporte de paquete ya que si falta algún dato no permitirá concretar la solicitud | |
| Resultados esperados: <ul style="list-style-type: none">• Llenado de formulario de solicitud• Envío de datos de solicitud | |
| Postcondición: Envío de datos | |
| Resultados obtenidos: <ul style="list-style-type: none">• Registro de datos de solicitud en el sistema• Despliegue de mensaje con envío exitoso | |

Fuente: Elaboración propia

En la asignación de solicitud a repartidor se describe la prueba de aceptación en la tabla 3.21

Tabla 3.21 Prueba de asignación de transporte de paquete

| Prueba 2.2 | Operación: Asignación de paquete |
|---|----------------------------------|
| Precondición: <ul style="list-style-type: none">• Usuario registrado y activo en el sistema• Notificación de solicitud de nuevo servicio | |
| Datos/Proceso <ul style="list-style-type: none">• Despliegue de solicitudes de servicio• Despliegue de usuarios repartidores en el mapa• Administrador selecciona al repartidor más cercano del punto de recolección | |
| Resultados esperados: <ul style="list-style-type: none">• Asignación al repartidor más cercano al punto de recolección | |
| Postcondición: Asignación al repartidor | |

Resultados obtenidos:

- Registro de datos de la asignación
- Despliegue de mensaje de asignación

Fuente: Elaboración propia

Una vez asignado el paquete debe hacerse el aviso al repartidor seleccionado, tal aviso llegara como una notificación al dispositivo del repartidor tabla 3.22.

Tabla 3.22 Notificación de aviso

| | |
|--|---|
| Prueba 2.3 | Operación: Notificación de aviso |
| Precondición: | |
| <ul style="list-style-type: none"> • Usuario registrado y activo en el sistema | |
| Datos/Proceso | |
| <ul style="list-style-type: none"> • En la notificación se le envía datos básicos respecto al paquete a transportar | |
| Resultados esperados: | |
| <ul style="list-style-type: none"> • Aviso de notificación al repartidor | |
| Postcondición: | |
| Llegada de datos del paquete | |
| Resultados obtenidos: | |
| <ul style="list-style-type: none"> • Aviso de datos del paquete | |

Fuente: Elaboración propia

Al paquete se la asigna un código de proceso el cual es listado en el dispositivo del repartidor, a la vez obtiene el detalle de datos del paquete a ser transportado tabla 3.23.

Tabla 3.23 Lista y detalle del paquete

| | |
|--|---|
| Prueba 2.4 | Operación: Lista y detalle del paquete |
| Precondición: | |
| <ul style="list-style-type: none"> • Usuario registrado y activo en el sistema | |
| Datos/Proceso | |
| <ul style="list-style-type: none"> • Se listan los paquetes asignados • Se despliega el detalle del paquete a ser transportado | |
| Resultados esperados: | |
| <ul style="list-style-type: none"> • Despliegue del detalle del paquete | |
| Postcondición: | |
| Observar datos del paquete | |
| Resultados obtenidos: | |
| <ul style="list-style-type: none"> • Listado de paquetes • Detalles de paquetes | |

Fuente: Elaboración propia

3.4. Tercera Iteración (Sprint 3)

En esta iteración se presentará las tareas a llevar a cabo en la tabla 3.24, posteriormente se pasará al modelo de presentación del sprint.

Tabla 3.24 Detalle de tareas a realizar en el Sprint 3

| Primera Iteración | | Sprint | | |
|-------------------|--|------------|-----------------|-----------|
| | | 3 (tres) | | |
| ID | Tareas | Tipo | Días de Trabajo | Estado |
| | Realizar un historial de pedidos para el cliente en el cual se desplegar información de su pedido si fue aceptado se genera código QR, si fue rechazado y por ultimo si fue entregado su paquete a su destino final | Desarrollo | 1 | Terminado |
| 3.1 | Generar código QR para el cliente una vez se haya aceptado la solicitud de envío de paquete. Desplegarlo en la pantalla de historial de pedidos para que pueda descargarlo el cliente y ser colocado a su paquete a enviar | Desarrollo | 1 | Terminado |
| | Por medio de código QR escaneado del paquete por el usuario repartidor, se debe notificar al administrador cambios de estado del paquete recolección y entrega del mismo, será listado de notificaciones | Desarrollo | 1 | Terminado |
| 3.2 | Diseño para reportes de transporte de paquetes | Diseño | 2 | Terminado |
| 3.3 | Desarrollo del despliegue de reportes de paquetes en general | Desarrollo | 1 | Terminado |
| 3.4 | Desarrollo de búsquedas por fecha y filtrar datos por campo de los reportes de paquete | Desarrollo | 2 | Terminado |
| 3.5 | Exportar el listado de reportes en formato Excel y tener la opción de imprimir | Desarrollo | 2 | Terminado |

Fuente: Elaboración propia

a) Análisis del Sprint 3

Modelo Lógico Conceptual

Continuando con el análisis tenemos la notificación del cambio de estado del paquete el mismo llega a dar a conocer tanto al administrado como al cliente el estado en que se encuentra el paquete como se puede apreciar en la figura 3.30.

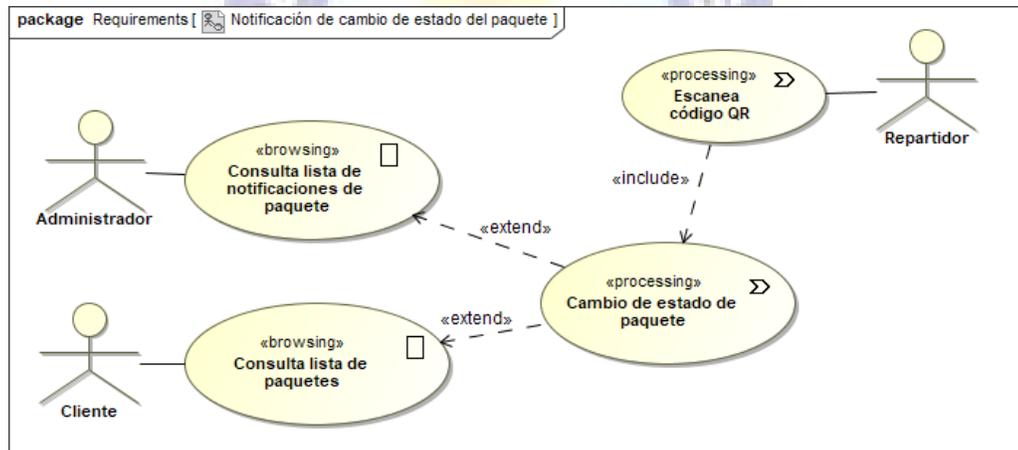


Figura 3.30 Diagrama de caso de uso – Notificación de cambio de estado del paquete
Fuente: Elaboración propia

En la siguiente figura 3.31 tenemos el detalle de la generación de reporte sobre los casos trabajos concluidos.

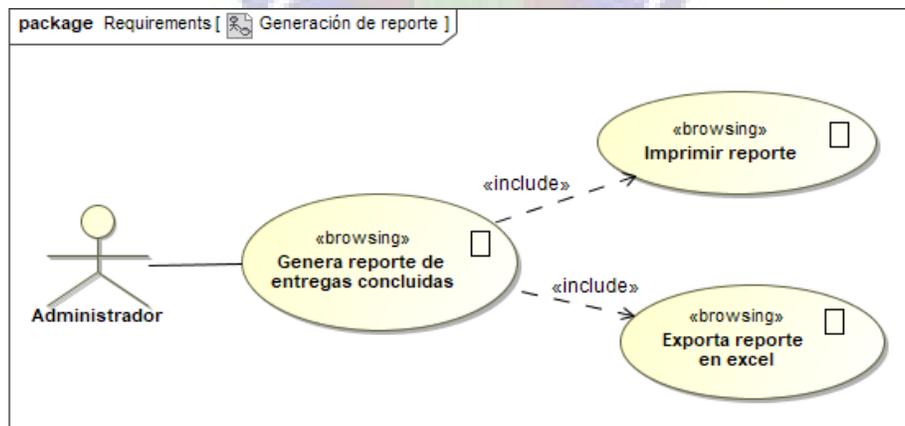


Figura 3.31 Diagrama de caso de uso – Generación de reporte
Fuente: Elaboración propia

b) Modelo Navegacional

Tenemos en la figura 3.32 la representación de la navegación e interacción de las interfaces ya sean del administrador repartidor y cliente para el cambio de estado del paquete.

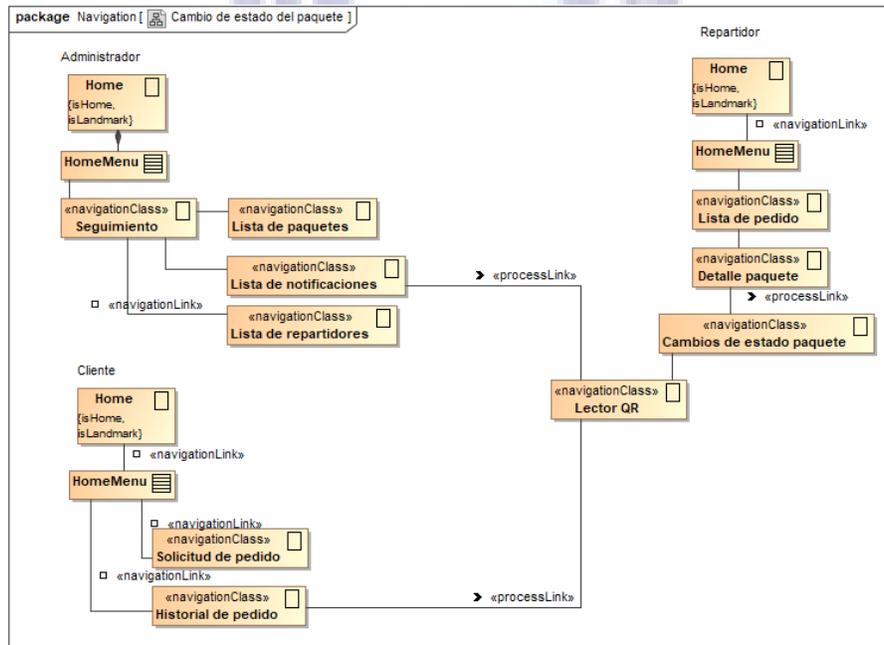


Figura 3.32 Diagrama de navegación – Cambio de estado del paquete
Fuente: Elaboración propia

c) Escenario Web

Del lado del módulo del cliente se desarrolló un historial de pedidos figura 3.33 y el detalle del pedido, el cual se complementa con las notificaciones a la base de datos y al módulo de administración de paquetes donde se da a conocer cambios de estado que también se refleja en el detalle de pedido figura 3.34, una vez asignado al personal que pasará por el paquete se habilita la impresión de código QR el cual será pegado al paquete figura 3.35.

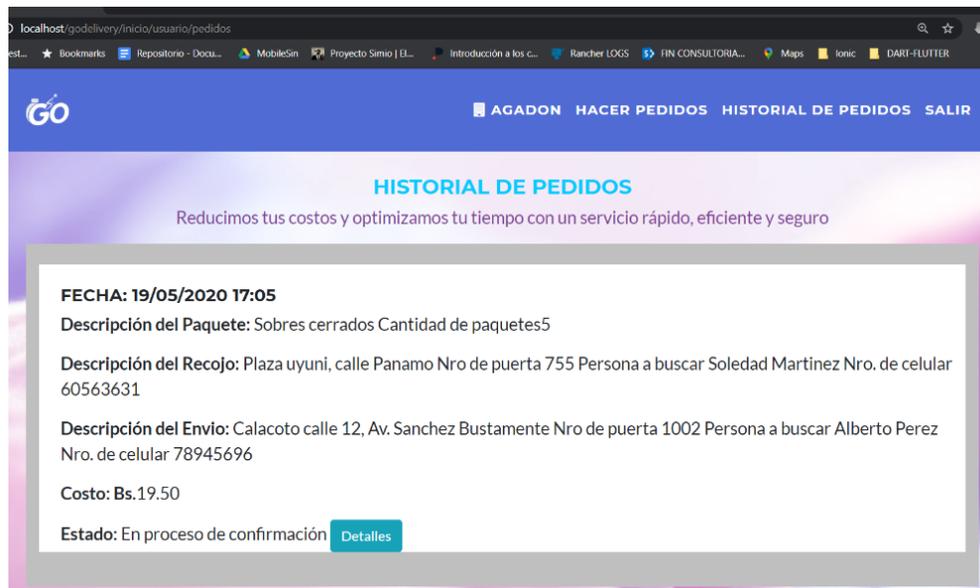


Figura 3.33 Historial de pedidos
Fuente: Elaboración propia

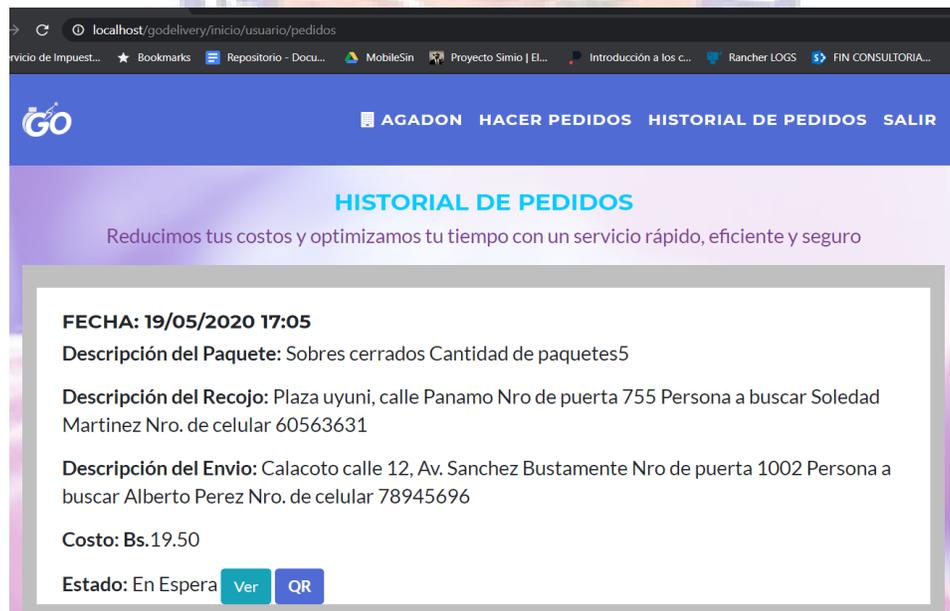


Figura 3.34 Cambio de estado del historial de pedidos
Fuente: Elaboración propia

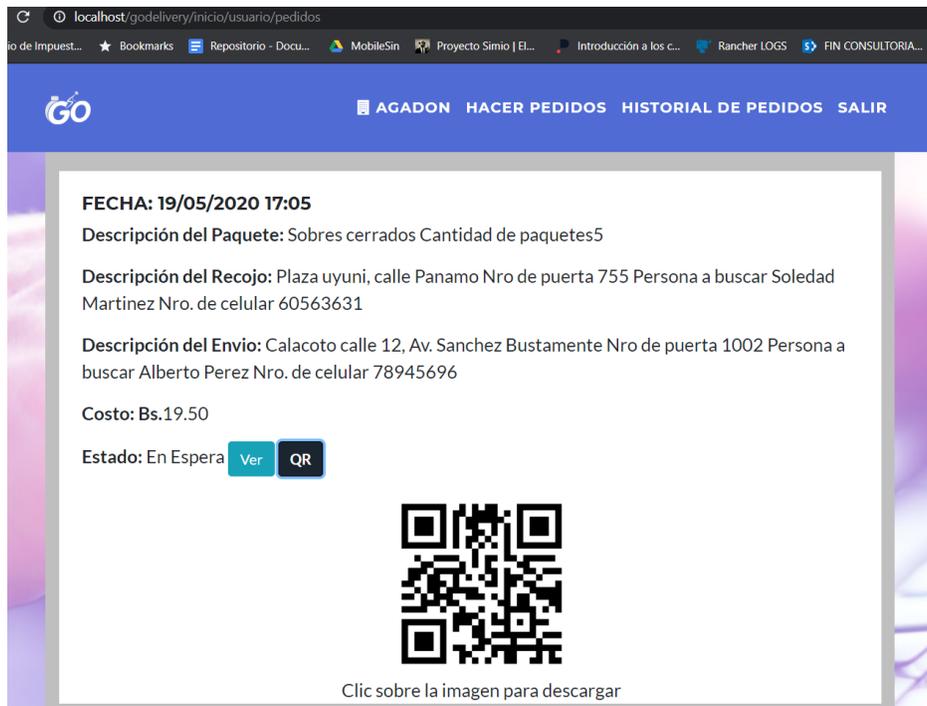


Figura 3.35 Generación de código QR
Fuente: Elaboración propia

En figura 3.36, se da a conocer al cliente que su paquete ya fue entregado a su destinatario.

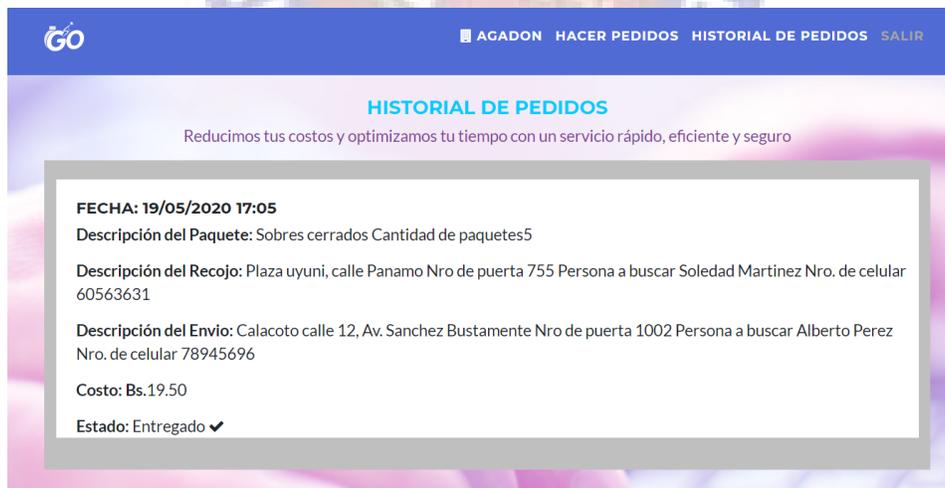


Figura 3.36 Conclusión de envío de paquete
Fuente: Elaboración propia

En detalle del paquete tenemos la opción del botón ver el cual despliega datos del personal asignado al recojo del paquete figura 3.37.



Figura 3.37 Personal de la empresa que recolecta el paquete
Fuente: Elaboración propia

Se tiene el proceso de recolección y entrega de paquete a la vez se despliega notificaciones del cambio de estado de recolectado a entregado este proceso se puede apreciar en las figuras 3.38, 3.39. Como también se tiene en la figura 3.40 el envío de cambios de estado el paquete desde el dispositivo del repartidor.

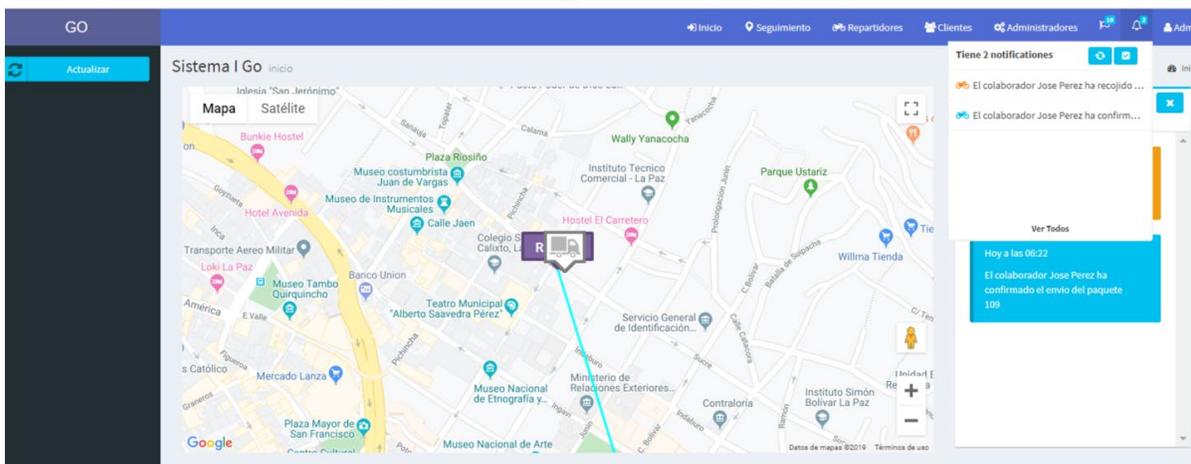


Figura 3.38 Punto de recolección de paquete
Fuente: Elaboración propia

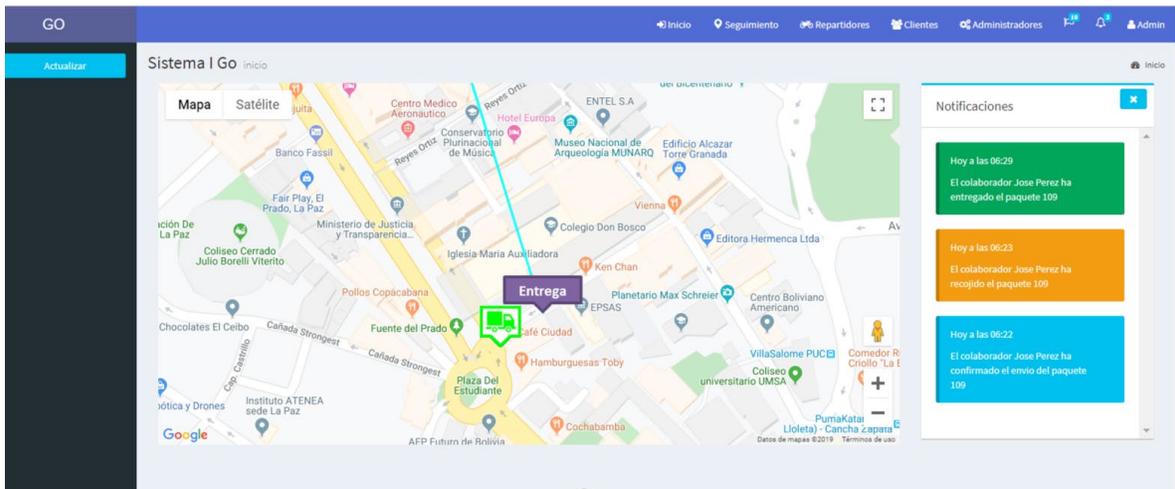


Figura 3.39 Punto de entrega de paquete
Fuente: Elaboración propia

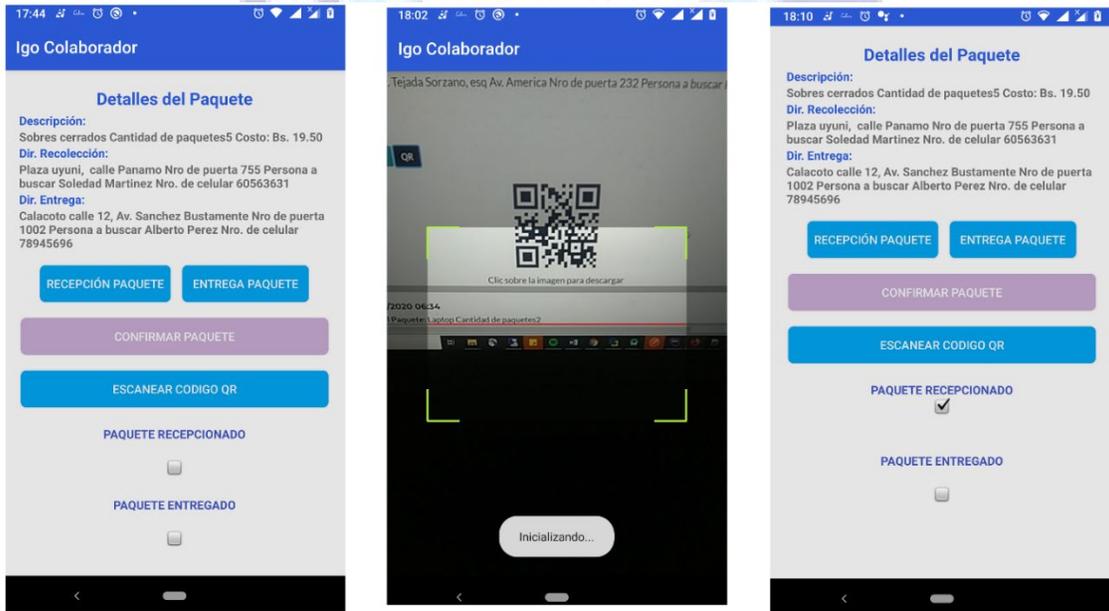


Figura 3.40 Cambios de estado desde el dispositivo móvil
Fuente: Elaboración propia

Para un control de cuantos pedidos se tiene por cliente y el repartidor que se asignó se detalla en un reporte que se observa en la figura 3.41, este contiene búsqueda por fechas y un filtrado por campo, y a la vez se tiene la funcionalidad de exportarlo a formato Excel figura 3.42 o realizar una impresión del misma.

GO Inicio Seguimiento Repartidores Clientes Administradores Admin

Menú Navegacional

- Principal
- Tablero de Seguimiento
- Repartidores
- Clientes
- Administradores
- Reportes

Administradores

Desde: 14/04/20 Hasta: 20/04/20

Buscar:

| Codigo | Cliente | Tipo | Descripción del Paquete | Fecha de Petición | Descripción del Recojo | Descripción de la Entrega | Entregado por | Hora de Recojo | Hora de Entrega | Costo |
|--------|-------------|---------|---|-------------------|---|---|---------------|----------------|-----------------|-------|
| 103 | luis choque | Persona | 1 Cantidad de paquetes2 | 14/04/20 08:00 | Av. Ramos Gavilan Zona Achachicala Nro 1094 Calle 7 Nro de puerta 12 Persona a buscar Juan Carlos Nro. de celular 2 | Av. Ramos Gavilan Zona Achachicala Nro 1094 Calle 7 Nro de puerta 12 Persona a buscar 12 Nro. de celular 34234234 | Jose Perez | 15:18 | 15:47 | 12.00 |
| 108 | luis choque | Persona | Tahuatagua Cantidad de paquetes88 | 16/04/20 20:18 | Av. Ramos Gavilan Zona Achachicala Nro 1094 Calle 7 Nro de puerta 2 Persona a buscar Juan Carlos/Nro. de celular 2 | Av. Ramos Gavilan Zona Achachicala Nro 1094 Calle 7 Nro de puerta 2 Persona a buscar Roberto/Nro. de celular 2 | Jose Perez | 20:43 | 20:01 | 12.00 |
| 109 | luis choque | Persona | paquete de Libros Cantidad de paquetes3 | 17/04/20 04:15 | Av. Sucre, esq Sanjines Nro de puerta 1200 Persona a buscar Lucas Paco Nro. de celular 78864971 | Calle Batallon Colorados Nro de puerta 45 Persona a buscar Fabio Medez Nro. de celular 60594586 | Jose Perez | 06:11 | 06:21 | 12.00 |

Mostrando 1 a 3 de 3 elementos.

Figura 3.41 Reporte de paquetes procesados
Fuente: Elaboración propia

Sistema | GoDelivery.xlsx [Vista protegida] - Excel

VISTA PROTEGIDA Cuidado—los archivos de Internet pueden contener virus. Si no tiene que editarlo, es mejor que siga en Vista protegida.

| Cliente | Tipo | Descripción del Paquete | Fecha de Petición | Descripción del Recojo | Descripción de la Entrega | Entregado por | Hora de Recojo | Hora de Entrega | Costo |
|-------------|---------|--|-------------------|--|---------------------------|---------------|----------------|-----------------|-------|
| luis choque | Persona | Paquete de vidrios Cantidad de paquetes1 | 17/05/2020 10:44 | Av. chacaltalla Nro de puerta 789 Person Miraflores Nro de puerta 7 | Jose Perez | 10:57 | | | |
| luis choque | Persona | Sobre de papel Cantidad de paquetes2 | 17/05/2020 16:05 | Av. Ramos Gagilan zona Norte Nro de pu Villa Fatima Nro de puerta | Jose Perez | 16:35 | | | |
| luis choque | Persona | Camara digital en caja Cantidad de paquetes1 | 17/05/2020 17:32 | Calle indaburo Nro de puerta 74532 Pers Obrajes calle 2 Nro de pue | Lucia Perez | 17:13 | | | |
| luis choque | Persona | Sobre con documentos Cantidad de paquetes1 | 17/05/2020 17:25 | calle Chuquisaca Nro de puerta 7 Person; Obrajes calle2 Nro de pue | Lia Gomes | 18:21 | | | |
| luis choque | Persona | Caja de madera Cantidad de paquetes2 | 17/05/2020 17:12 | Calle Junin, plaza Murillo Nro de puerta 7 Plaza Avaroa Nro de puert | Jose Perez | 18:03 | | | |
| luis choque | Persona | Pastel en caja Cantidad de paquetes1 | 17/05/2020 20:25 | Calle Catacora Nro de puerta 123 Person Laikacota Nro de puerta 1 | Lucia Perez | 09:10 | | | |
| luis choque | Persona | Sobres Cantidad de paquetes10 | 18/05/2020 04:01 | Inicio Armentia Nro de puerta 12 Person; parada puma parke centra | Xavi Noa | 04:52 | | | |
| luis choque | Persona | Libros Cantidad de paquetes3 | 18/05/2020 05:54 | C/ Guerrero Lanza Nro de puerta 1223 Pæ Av. Del Maestro Nro de pu | Xavi Noa | 01:00 | | | |
| luis choque | Persona | Laptop Cantidad de paquetes2 | 18/05/2020 06:34 | Landaeta Nro de puerta 45 Persona a bu: Calle Colon Nro de puerta | Xavi Noa | 07:36 | | | |
| Agadon | Empresa | Sobres cerrados Cantidad de paquetes5 | 19/05/2020 17:05 | Plaza uyuni, calle Panamo Nro de puerta; Calacoto calle 12, Av. Sanc | Xavi Noa | 17:09 | | | |
| luis choque | Persona | paquete de oficina Cantidad de paquetes5 | 19/05/2020 17:45 | Calle Colon, entre ballivian y comercio Ni Av. Tejada Sorzano, esq | Av Lucia Perez | 01:00 | | | |

Figura 3.42 Reporte de paquetes procesados
Fuente: Elaboración propia

CAPÍTULO IV

CALIDAD DE SOFTWARE Y COSTO BENEFICIO

El presente capítulo, calidad de software y seguridad, presenta las metodologías usadas para poder usar, tanto la calidad y a su vez en base a que se disponen los diferentes niveles de seguridad requeridos para este proyecto.

La calidad de software es el desarrollo de software basado en estándares con la funcionalidad y un rendimiento total, los cuales satisfacen los requerimientos del cliente.

Se debe medir la calidad del producto, y a su vez se deben definir los niveles de seguridad con los que se debe contar, en esta oportunidad, para medir la calidad del producto se utilizó el estándar ISO 9126.

4.1. Calidad de Software

4.1.1. Norma de Calidad

Una norma de calidad es un papel, establecido por consenso y aprobado por un organismo reconocido nacional o internacional, que se proporciona para un uso común y repetido, una serie de reglas, directrices o características para las actividades de calidad o sus resultados, con el fin de conseguir un grado óptimo de orden en el contexto de la calidad (ISO, 2020). La calidad de software se define como el cumplimiento de los requisitos funcionales y del rendimiento establecido, en relación de los estándares de desarrollo documentados y características implícitas esperadas.

4.1.2. Métricas de Calidad

Para la realización del presente proyecto, específicamente para las propiedades de calidad que se identifiquen. En el siguiente párrafo se muestran ejemplos de métricas de calidad y como se las llegan a determinar para las propiedades del sistema.

Este estándar, ISO/ICE 9126 ha sido desarrollado en un intento de identificar atributos claves de calidad de software. El estándar identifica seis atributos claves de calidad, los cuales son; la funcionalidad, usabilidad, mantenimiento, portabilidad, confidencialidad y eficiencia (Sánchez, 1999).

4.1.2.1. Funcionalidad

El sistema debe ser capaz de proveer las funciones que cumplen con las necesidades explícitas cuando es utilizado en las condiciones especificadas por el cliente (Sánchez, 1999).

Para calcular la funcionalidad del sistema, se llegaron a determinar los siguientes elementos de representación del sistema:

- Entradas externas: entradas de usuario.
- Salidas externas: salidas que proporcionan información al usuario.
- Consultas externas: peticiones interactivas.
- Ficheros externos: interfaces con otros sistemas.
- Ficheros internos: ficheros lógicos del sistema.

Tabla 4.1 Conteo de elementos de representación del sistema

| Elementos de representación | Procesos | Total |
|-----------------------------|---|-------|
| Entradas Externas | Pantalla de ingreso al sistema web administrador Pantalla de ingreso al sistema web cliente Pantalla de ingreso a la aplicación móvil Registro de administradores Registro de clientes Registro de repartidores Modificación de administradores Modificación de clientes Modificación de repartidores | 12 |

| | | |
|--------------------|--|---|
| | Asignación de paquete al repartidor, (pantalla de seguimiento) Registro de solicitudes de envío de paquete Cambio de estado de paquetes | |
| Salidas Externas | Reporte de paquetes Detalles de paquete Generación de código QR Listado de administradores Listado de clientes Listado de repartidores Reportes de paquetes procesados y/o finalizados Notificación de cambios de estado del paquete | 8 |
| Consultas Externas | Menú principal de sistema web Menú principal web del cliente Opciones de aplicación móvil Búsqueda por coincidencia en de listado de administradores Búsqueda por coincidencia en de listado de clientes Búsqueda por coincidencia en de listado de repartidores Búsqueda por fecha en reportes Seguimiento de repartidores | 8 |
| Ficheros Externos | Generar y recuperar de token Firabase de celulares Firebase para notificaciones push | 2 |
| Ficheros Internos | Tablas de base de datos | 7 |

Fuente: Elaboración propia

A cada elemento se le asignó un índice de complejidad entre: simple, media y compleja según (IFPUG, 2020), a su vez a cada índice le corresponde un factor de ponderación. En el análisis se trabaja con el proceso medio como se puede ver en la tabla 4.2

Tabla 4.2 Métricas de punto función

| Factor De Ponderación | | | | | |
|-------------------------------------|---------------|---------------|--------------|-----------------|--------------|
| Parámetros de medición | Cuenta | Simple | Medio | Complejo | Total |
| Número de entradas externas | 12 | 3 | 4 | 6 | 48 |
| Número de salidas externas | 8 | 4 | 5 | 7 | 40 |
| Número de consultas externas | 8 | 3 | 4 | 6 | 32 |
| Número de ficheros externos | 2 | 5 | 7 | 10 | 14 |
| Número de ficheros internos | 7 | 5 | 10 | 15 | 70 |
| TOTAL | | | | | 204 |

Fuente: Web IFPUG, 2020

Los valores del factor de complejidad técnica se llegaron a obtener respondiendo las siguientes interrogantes acerca del sistema, ver tabla 4.3

Tabla 4.3 Componentes del factor de complejidad técnica

| F | Factores de Complejidad Técnica | Sin | Inc | Moderado | Medio | Significativo | Esencial |
|----------|---|-----|-----|----------|-------|---------------|----------|
| | | 0 | 1 | 2 | 3 | 4 | 5 |
| F_1 | Copias de seguridad y de recuperación fiables | | | | | | X |
| F_2 | Comunicación de datos | | | | | | X |
| F_3 | Procedimientos distribuidos | | | | | X | |
| F_4 | Es critico el rendimiento | | | | X | | |
| F_5 | Configuración muy cargada | | X | | | | |
| F_6 | Entrada de datos interactiva | | | | | | X |
| F_7 | Facilidad operativa | | | | | | X |
| F_8 | Actualización interactiva | | | | | X | |
| F_9 | Interfaces complejas (entradas, salidas, archivos, entre otros) | | | | | X | |
| F_{10} | Procesamiento interno complejo | | | | | X | |
| F_{11} | Diseño del código reutilizable | | | | | X | |
| F_{12} | Facilidad de instalación | | | | | | X |
| F_{13} | Soporta múltiples instalaciones en diferentes sitios | | | | | X | |
| F_{14} | Facilidad de cambios | | | | | X | |

Fuente: Elaboración propia

Los valores constantes y los factores de peso aplicados en las encuestas de los ámbitos han sido determinados empíricamente.

Por tanto:

$$\sum F_i = F_1 + F_2 + F_3 + F_4 + F_5 + F_6 + F_7 + F_8 + F_9 + F_{10} + F_{11} + F_{12} + F_{13} + F_{14}$$

$$\sum F_i = 5 + 5 + 4 + 3 + 1 + 5 + 5 + 4 + 4 + 4 + 4 + 5 + 4 + 4 = 57$$

Para medir la funcionalidad del sistema se utiliza la siguiente formula (Sánchez, 1999)

$$PF = Total * [Grado de confiabilidad + Tasa de error * \sum F_i]$$

Dónde: PF = Medida de funcionalidad

$$\sum F_i = \text{Ajuste de la complejidad segun el dominio de la informacion (i = 1 a 14)}$$

Reemplazando en la fórmula de medición de funcionalidad (PF), para un nivel de confianza de 65% se tiene:

$$PF = 204 * [0.65 + 0.01 * 57]$$

$$PF = 248.88$$

Ahora se pasa a calcular para un nivel de confianza del 100%

$$PF = 204 * [1 + 0.01 * 57]$$

$$PF = 320.28$$

El porcentaje de funcionalidad se define por la siguiente formula:

$$PF = \frac{PF_{REAL}}{PF_{ESPERADO}}$$

$$PF = \frac{248.88}{320.28} = 0.777$$

Por lo tanto, la funcionalidad del sistema es de 77.71% tomando el punto función máximo.

4.1.2.2. Confiabilidad

Grado en que el sistema responde bajo las condiciones definidas durante un intervalo de tiempo dado (Sánchez, 1999).

Diagrama de bloque de confiabilidad

Un diagrama de bloque de confiabilidad o reliability block diagram, es un método de modelar la forma en que los componentes y las fallas de subsistemas, se combinan para causar una falla de sistema. Los diagramas de bloque de confiabilidad pueden ser analizados para predecir la disponibilidad de un sistema y determinar los componentes críticos desde el punto de vista de la confiabilidad

El análisis de confiabilidad se realiza en base al diagrama de bloque planteado en la figura 4.1, que muestra el funcionamiento en serie y en paralelo de los subsistemas que contiene el Sistema Web de Seguimiento y Localización de Personal de Servicio de Delivery. Cada subsistema se muestra en un bloque y se denota con la letra R.

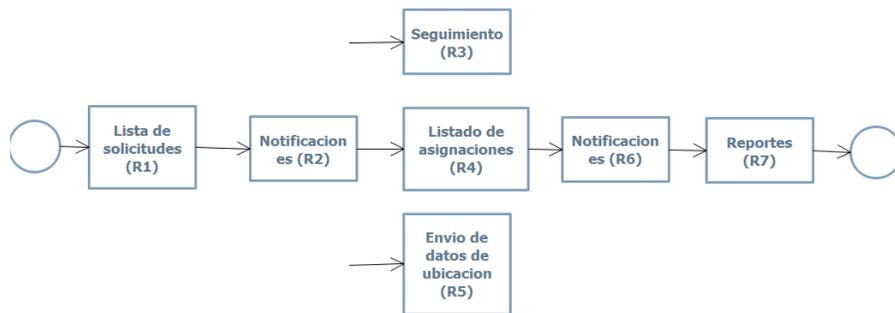


Figura 4.1 Diagrama de bloque de confiabilidad
Fuente: Elaboración propia

Luego en base a la figura 4.1, se llega a elaborar la tabla 4.4, donde se realiza el cálculo de la confiabilidad con un tiempo de 10 horas en todos los casos

Tabla 4.4 Cálculo de confiabilidad

| R_i | s | t [hrs] | e^{-s*t} |
|-------|-------|-----------|------------|
| R_1 | 0.001 | 10 | 0.99 |
| R_2 | 0.005 | 10 | 0.95 |
| R_3 | 0.003 | 10 | 0.97 |
| R_4 | 0.001 | 10 | 0.99 |
| R_5 | 0.003 | 10 | 0.97 |
| R_6 | 0.002 | 10 | 0.92 |
| R_7 | 0.001 | 10 | 0.99 |

Fuente: Elaboración propia

Teorema 1: Si n componentes funcionan independientes conectados en serie y el i -ésimo componente tiene confiabilidad $R_i(t)$

Según Sánchez (1999) la confiabilidad total está dada por:

$$R(t) = R_1 * R_2 * R_3 * \dots * R_n$$

Teorema 2: Si n componentes funcionan independientemente y actúa en paralelo y el i -ésimo componente tiene confiabilidad $R_i(t)$, entonces según Sánchez (1999) la confiabilidad está dada por:

$$R(t) = 1 - [1 - R_1(t)] * [1 - R_2(t)] * \dots * [1 - R_n(t)]$$

Utilizando los dos teoremas se calcula:

$$R(t) = \{1 - [1 - R_1(t)] * [1 - R_2(t)] * R_3(t) * R_4(t) * R_5(t) * R_6(t) * R_7(t)\}$$

$$R(t) = \{1 - [1 - 0.99] * [1 - 0.95]\} * 0.99 * 0.95 * 0.97 * 0.99 * 0.97 * 0.92 * 0.99$$

$$R(t) = \{1 - 0.0005\} * 0.99 * 0.95 * 0.97 * 0.99 * 0.97 * 0.92 * 0.99 = 0.7975$$

$$\text{Confiabilidad} = R(t) * 100\% = 79.75\%$$

El resultado anterior indica que el Sistema presenta una confiabilidad del 80%, es decir que el sistema es aceptable y se asegura que los componentes de todos los niveles funcionan. El 20% restante indica que el sistema puede llegar a fallar cuando se excede un determinado tiempo de uso continuo.

4.1.2.3. Usabilidad

Es la capacidad del sistema de ser entendido, aprendido, usado y de resultado atractivo para el usuario final (Sánchez, 1999).

- **Comprensión:** el uso del sistema debe ser fácilmente comprensible para el usuario final.
- **Facilidad de aprendizaje:** indica las características del software que influyen en el esfuerzo del usuario para aprender su aplicación.

- **Operatividad:** es la dificultad que presenta el software para ser controlado por el usuario final.
- **Atractivo:** indica las características del software que influyen en la satisfacción de los deseos del usuario y en las preferencias a través de servicios, comportamiento y presentación.

Para medir la usabilidad del sistema se hizo un cuestionario, el cual que puede observar en la tabla 4.5, con el objetivo de facilitar la evaluación de la usabilidad del sistema, basándose en una escala de evaluación.

Tabla 4.5 Evaluación de usabilidad

| Nro. | FACTOR | VALOR |
|--------------|---|-----------|
| 1 | ¿Se ha satisfecho todos los requerimientos establecidos por la empresa? | 5 |
| 2 | ¿Es sencillo acceder a los datos? | 4 |
| 3 | ¿Presenta una interfaz amigable para el usuario? | 5 |
| 4 | ¿Los reportes generados son representativos? | 5 |
| 5 | ¿El sistema cuenta con la seguridad necesaria? | 5 |
| 6 | ¿Está de acuerdo con el funcionamiento del sistema? | 4 |
| 7 | ¿El sistema facilita de gran manera el trabajo que realiza? | 5 |
| TOTAL | | 33 |

Fuente: Elaboración propia

En la tabla 4.6 se puede visualizar la escala de evaluación con la que se trabajó para el desarrollo de la evaluación de usabilidad.

Tabla 4.6 Escala de evaluación

| Factor | Escala |
|------------------|--------|
| Pésimo | 1 |
| Malo | 2 |
| Regular | 3 |
| Bueno | 4 |
| Muy Bueno | 5 |

Fuente: Elaboración propia

Según Sánchez (1999), la facilidad de uso se calcula mediante la siguiente formula, la cual se encuentra en base al cuestionario:

$$Facilidad\ de\ Uso = \left[\left(\frac{Valor}{n} \right) * 100 \right] / 5$$

Reemplazando datos se tiene:

$$Facilidad\ de\ uso = \frac{\left[\left(\frac{33}{7} \right) * 100 \right]}{5}$$

$$Facilidad\ de\ uso = 94.28\%$$

Observando y analizando el resultado de 94.28% que se obtuvo al momento de medir la usabilidad del sistema, se puede concluir que la facilidad de uso es buena.

4.1.2.4. Eficiencia

Suministrar un desempeño correcto en relación a los recursos utilizados, bajo condiciones establecidas (Sánchez, 1999)

- **Comportamiento temporal:** características que influyen en el tiempo de respuesta, procesado y productividad cuando se ejecuta su función.
- **Utilización de recursos:** usar cantidades y tipos de recursos apropiados, en condiciones especificadas (no influye a los recursos humanos)

Según Sánchez (1999), el factor de eficiencia viene dado por la siguiente formula:

$$EDD = \frac{E}{E + D}$$

Dónde:

E = es el número de errores encontrados antes de la entrega del software al usuario final.

D = es el número de defectos encontrados después de la entrega.

El valor ideal de EED es 1, donde simboliza que no se encontraron defectos en el software. De manera realista $D > 0$, pero el valor de EED todavía se puede aproximar a 1 cuando E aumenta.

El número de errores encontrados en 20 pruebas realizadas, antes y después de la entrega del sistema se muestran en la tabla 4.7.

Tabla 4.7 Conteo de errores antes y después de la entrega

| Error antes o después de la entrega | Número de errores |
|-------------------------------------|-------------------|
| E | 8 |
| D | 1 |

Fuente: Elaboración propia

Reemplazando los datos se tiene:

$$EDD = \frac{8}{8 + 1}$$

$$EDD = 0.88$$

$$EDD = 88\%$$

El resultado anterior presenta una eficiencia del 88%, es decir que el sistema es eficiente y no muestra ningún problema en 9 de cada 10 veces en que se la usa.

4.1.2.5. Mantenibilidad

Indica el esfuerzo requerido para implementar cambios. (Sánchez, 1999)

- **Capacidad para ser analizado:** indica la cantidad de esfuerzo requerido para diagnosticar la causa de un fallo o un nuevo requerimiento.
- **Capacidad de modificación:** se describe como la cantidad de esfuerzo requerido para la modificación o eliminación de un defecto.
- **Estabilidad:** indica el volumen de riesgo y de efectos inesperados luego de realizar una modificación.

- **Facilidad para ser probado:** capacidad del software para permitir que sea validado después de ser modificado.

El factor de mantenibilidad viene dado por el estándar IEEE 928.1-1988, por las métricas de Índice de Madurez del Software IMS, que proporciona una indicación de la estabilidad del producto software, basado en los cambios que ocurre con cada versión del producto, Sánchez (1999) determina este factor mediante la siguiente formula:

$$IMS = \frac{Mt - (F_a + F_c + F_d)}{Mt}$$

Dónde:

MT = Numero de módulos en la versión actual

F_c = Numero de módulos en la versión actual que se ha cambiado

F_a = Numero de módulos en la versión actual que se ha añadido

F_d = Numero de módulos en la versión anterior que se borraron en la versión actual

El sistema empieza a estabilizarse si el valor del IMS se acerca a 1.

Entonces los valores encontrados son: $Mt = 5$, $F_a = 2$, $F_d=1$, $F_c= 0$, reemplazando en la

ecuación:

$$IMS = \frac{[5 - (1 + 0 + 0)]}{5}$$

$$IMS = 0.8 * 100 = 80\%$$

Observando el resultado de 80% en mantenibilidad del Sistema Web de Seguimiento y Localización de Personal de Servicio de Delivery, puede indicarse que el sistema tiende a estabilizarse.

4.1.2.6. Portabilidad

Es la capacidad del software para ser transferido de un ambiente de operaciones a otro. La portabilidad del software se enfoca en tres aspectos: a nivel de aplicaciones, a nivel de sistema operativo y a nivel de hardware (Sánchez, 1999).

Además de ver estos tres aspectos, también se considera la facilidad de instalación, ajuste y adaptación al cambio. Dado por la siguiente fórmula (Sánchez, 1999):

$$GP = 1 - (CT / CRD)$$

Dónde:

GP = Grado de portabilidad

CT = Costo de transportar, es el costo máximo de un dispositivo en el cual puede llegar a transportar la aplicación.

CRD = Costo de re-desarrollo, el cual es menor al costo de Desarrollo.

Y tendremos que

Si $GP > 0$, la portabilidad es más rentable que el re-desarrollo.

Si $GP = 1$, la portabilidad es perfecta.

Si $GP < 0$, el re-desarrollo es más rentable que la portabilidad. Reemplazando datos en la fórmula se tiene:

$$GP = 1 - (50/1200)$$

$$GP = 0.96$$

Observando el resultado, el Sistema Web de Seguimiento y Localización de Personal de Servicio de Delivery, tiene la facilidad de transportarse de un entorno a otro de un 96%.

Nivel de aplicaciones

La aplicación móvil fue desarrollada en Android Studio 3.6.0, es portable ya que la aplicación móvil puede ser distribuida mediante dispositivos móviles.

El modulo web fue desarrollado en phpStorn 19.2, es portable ya que el modulo web puede ser distribuido en CD' s.

Nivel de sistema operativo

La aplicación móvil, es portable para dispositivos móviles con una versión Android 5.0 o mayores.

El modulo web, es portable para los sistemas operativos de Windows 2000, XP, Vista, 7, Windows 7, 8, 10.

Nivel de hardware

La aplicación móvil, es portable para dispositivos móviles que cuenten con GPS y cámara. El modulo web, es portable para todas las computadoras mayores o iguales a la tecnología Core 2 dúo.

4.1.2.7. Análisis de Resultados

Una vez aplicadas las métricas se prosigue a realizar un análisis de los resultados obtenidos para determinar la calidad del sistema, para ello se realiza un resumen de los datos obtenidos, ver tabla 4.8

Tabla 4.8 Resultados de métricas de calidad

| Métrica de Calidad | Resultado |
|---------------------------|------------------|
| Funcionalidad | 77.71% |
| Confiabilidad | 79.75% |
| Usabilidad | 94.28% |
| Eficiencia | 88.00% |
| Mantenibilidad | 80.00% |
| Portabilidad | 96.00% |
| Calidad Global | 85.95% |

Fuente: Elaboración propia

De acuerdo con los indicadores obtenidos en las métricas de calidad, analizando y sacando un promedio de la suma de los resultados, se llega a concluir que el Sistema Web de

Seguimiento y Localización de Personal de Servicio de Delivery, está dentro del nivel satisfactorio con un valor de 85.95% de calidad global.

4.2. Seguridad

Para la seguridad de la aplicación se toman en cuenta cuatro áreas principales que cubre la seguridad informática.

- **Confiabledad:** Solo los usuarios autorizados pueden acceder a nuestros recursos, datos e información.
- **Integridad:** Solo los usuarios autorizados deben ser capaces de modificar los datos cuando sea necesario.
- **Disponibilidad:** Los datos deben estar disponibles para los usuarios cuando sea necesario.
- **Autenticación:** Estar realmente comunicándose con los usuarios registrados en el sistema.

4.2.1. Seguridad de Software

En el avance del proyecto no se debe obviar la implementación de medidas de seguridad referente al software; En mención a este punto se resalta la implementación de autenticación de usuarios, roles según usuario, encriptación de contraseñas.

4.2.2. Seguridad de Hardware

La seguridad del hardware se refiere a la protección de computadoras o dispositivos frente a intromisiones o amenazas. No solo hay que proteger el software, sino que también el hardware que estamos utilizando en nuestras actividades diarias es importante y requiere protección.

El método más utilizado es el manejo de cortafuegos o firewalls de hardware y servidores proxy. Existe otro método que emplea módulos de seguridad de hardware que utilizan claves criptográficas para el cifrado, descifrado y autenticación en los sistemas.

4.2.3. Seguridad de Red

La seguridad de red se refiere a cualesquiera actividades diseñadas para proteger la red. En concreto, estas actividades protegen la facilidad de uso, fiabilidad, integridad y seguridad de su red y datos.

La seguridad de red efectiva se dirige a una variedad de amenazas y la forma de impedir que entren o se difundan en una red de dispositivos. ¿Y cuáles son las amenazas a la red? Muchas amenazas a la seguridad de la red hoy en día se propagan a través de Internet. Los más comunes incluyen:

- Virus, gusanos y caballos de Troya
- Software espía y publicitario
- Ataques de día cero, también llamados ataques de hora cero
- Ataques de hackers
- Ataques de denegación de servicio
- Intercepción o robo de datos
- Robo de identidad

4.3. Análisis de Costo y Beneficio

En las secciones que se detallan a continuación, se examinarán diversos aspectos de los cálculos de costo/beneficio:

- Análisis de costos.
- Análisis de beneficios

- Costo/Beneficio

4.3.1. Análisis de Costos

En este paso se deben calcular los costos anticipados asociados con el sistema. Para determinar el costo total del proyecto, se tomará en cuenta los siguientes costos (Boehm, 1995):

- Costo del Software Desarrollado
- Costo de Implementación del Proyecto
- Costo de Elaboración del Proyecto

4.3.1.1. Costo del Software Desarrollado

Para determinar el respectivo costo del software desarrollado, se optó por utilizar el Modelo constructivo de costes COCOMO II, modelo orientado a puntos de función.

En la tabla 4.9 se puede ver el cálculo de punto función no ajustado, el detalle de los parámetros de medición se puede visualizar en la tabla 4.5.

Tabla 4.9 Cálculo punto función no ajustado

| Parámetro de medición | Cuenta | | Factor de ponderación | | Total web | Total móvil |
|------------------------------|--------|-------|-----------------------|-------|------------|-------------|
| | Web | Móvil | Web | Móvil | | |
| Número de entradas externas | 22 | 6 | 4 | 2 | 88 | 12 |
| Número de salidas externas | 8 | 2 | 4 | 2 | 32 | 4 |
| Número de consultas externas | 13 | 3 | 2 | 3 | 26 | 9 |
| Número de ficheros externos | 2 | 2 | 3 | 2 | 6 | 4 |
| Número de ficheros internos | 7 | 1 | 3 | 2 | 21 | 2 |
| CUENTA TOTAL | | | | | 173 | 31 |

Fuente: Elaboración propia

Cálculo de valores de ajuste de la complejidad tomando los valores de la Tabla 4.10, en la cual se determina la complejidad

Tabla 4.10 Cálculo de punto función ajustado

| Factores de complejidad | Valor Web | Valor móvil |
|--|------------------|--------------------|
| Requiere el sistema copias de seguridad y de recuperación fiables | 0 | 0 |
| Se requiere comunicación de datos | 5 | 3 |
| Existen funciones de procesamientos distribuido | 2 | 3 |
| Es crítico el rendimiento | 4 | 4 |
| Se ejecutará el sistema con un entorno operativo existente y fuertemente utilizado | 5 | 4 |
| Requiere el sistema entrada de datos interactiva | 5 | 3 |
| Facilidad operativa | 2 | 0 |
| Se actualiza los archivos maestros de forma interactiva | 4 | 0 |
| Son complejos las entradas, las salidas, los archivos o las peticiones. | 3 | 0 |
| Procesamiento interno complejo | 3 | 0 |
| Diseño del código reutilizable | 3 | 3 |
| Facilidad de instalación | 3 | 3 |
| Soporta múltiples instalaciones en diferentes sitios | 4 | 3 |
| Facilidad de cambios | 4 | 4 |
| TOTAL | 47 | 30 |

Fuente: Elaboración propia

$$\text{Factor de ajuste} = (0.65 + 0.01 * 47)$$

$$\text{Factor de Ajuste}_W = 1.12$$

$$\text{Factor de ajuste} = (0.65 + 0.01 * 30)$$

$$\text{Factor de Ajuste}_M = 0,95$$

Según Boehm (1981), el cálculo de los puntos función se basa en la siguiente fórmula:

$$PF = \text{Cuenta Total} * \text{Factor de Ajuste}$$

Cálculo de punto función

$$PF_W = 173 * 1.12$$

$$PF_W = 193,76$$

$$PF_M = 31 * 0.95$$

$$PF_M = 29,45$$

Conversión de los puntos función a cantidad de líneas de código distribuidas en miles KDLC. En este paso se convierten los PF a miles de líneas de código; para conocer los factores LDC/PF (Líneas de código y punto de función) ver la tabla 4.11.

Tabla 4.11 Conversión de Puntos Función a KLDC

| Lenguaje | Nivel | Factor LDC/PF |
|--------------|-------|---------------|
| C | 2.5 | 128 |
| Ansi Basic | 5 | 64 |
| Java | 6 | 53 |
| Ansi Cobol | 3 | 107 |
| Visual Basic | 7 | 46 |
| ASP | 9 | 36 |
| PHP | 11 | 29 |
| Visual C++ | 9.5 | 34 |

Fuente: Boehm, 1996

Según Boehm (1981) el factor LDC se define:

$$LDC = (PF * Factor LDC)$$

Reemplazando datos:

$$LDC_W = 193.76 * 29$$

$$LDC_W = 5619$$

$$LDC_M = 29,45 * 53$$

$$LDC_M = 1561$$

Calculando el KLDC

$$KLDC = (LDC)/1000$$

$$KLDC_W = 5619/1000$$

$$KLDC_W = 5,619$$

$$KLDC_M = 1561/1000$$

$$KLDC_M = 1,561$$

Según Boehm (1981) las ecuaciones de COCOMO II tienen la siguiente forma:

$$E = a_b(KLDC)b_b$$

$$D = c_b(E)d_b$$

Donde:

- E: Esfuerzo aplicado en personas por mes
- D: Tiempo de desarrollo en meses
- KLDC: Número estimado de líneas de código distribuidas (en miles)

En la tabla 4.12 se muestra los tipos de proyecto de software.

Tabla 4.12 Coeficiente a_b y b_b

| Proyecto de Software | a_b | b_b | c_b | d_b |
|----------------------|-------|-------|-------|-------|
| Orgánico | 2.4 | 1.05 | 2.5 | 0.38 |
| Semi-acoplado | 3 | 1.12 | 2.5 | 0.35 |
| Empotrado | 3.6 | 1.2 | 2.5 | 0.32 |

Fuente: Boehm, 1996

Como este es un proyecto intermedio, en tamaño y complejidad, se elige Seme-acoplado

$$E_W = 3 * (5,619)^{1.12}$$

$$E_W = 20,74$$

$$D_W = 2.5 * (20,74)^{0.35}$$

$$D_W = 7.22 \text{ meses}$$

$$E_M = 3 * (1,561)^{1.12}$$

$$E = 4,94$$

$$D_M = 2.5 * (4,94)^{0.35}$$

$$D_M = 4.37 \text{ meses}$$

Según Boehm (1981), el personal requerido se obtiene con la siguiente forma:

$$\text{Numero de Programadores} = \frac{E}{D}$$

Reemplazados datos:

$$\text{Numero de Programadores}_W = \frac{20,74}{7,22}$$

$$\text{Numero de Programadores}_W = 2,87$$

$$\text{Numero de Programadores}_M = \frac{4,94}{4,37}$$

$$\text{Numero de Programadores}_M = 1,13$$

El salario de un programador aproximadamente es de 300\$us., cifra que se tomara en cuenta para hallar el costo del software, con la siguiente formula (Boehm, 1996)

$$\text{Costo de software} = \text{Numero Programadores} * \text{Salario Programador}$$

$$\text{Costo de Software} = 4 * 300$$

$$\text{Costo del Software} = 1200 \$us$$

Analizando los resultados obtenidos, se concluye que el proyecto debió terminarse en un tiempo aproximado de 12 meses por un equipo de 4 programadores y que el costo de software por mes es la de 1200 \$us, lo que lleva a la conclusión de que el costo total de realización del costo en el lapso de los 12 meses es de 14400 \$us.

4.3.1.2. Costo de Despliegue del Proyecto

Como el Bolivia Tech Hub ya cuenta con sus servidores de aplicación y base de datos, completamente instalados y configurados y se tiene planeado el uso del aplicativo móvil en los celulares de los repartidores por lo que el costo de implementación es cero para el sistema.

4.3.1.3. Costo de Elaboración del Proyecto

Este paso se refiere a los costos que se llegan a cubrir durante el análisis del sistema y diseño del sistema, en esta etapa de análisis los costos que se llegaron a observar se detallan en la tabla 4.13.

Tabla 4.13 Costo de elaboración del proyecto

| Descripción | Costo total (\$us) |
|--------------------------------|--------------------|
| Análisis y diseño del proyecto | 250 |
| Bibliografía (investigación) | 40 |
| Material de escritorio | 70 |
| Otros | 30 |
| Total | 390 |

Fuente: Elaboración propia

4.3.1.4. Costo Total

El costo total se refiere al costo que tendrá el desarrollo del software en todos sus aspectos como son: el costo de desarrollo de software, cuyo detalle y desarrollo puede llegar a observarse en el punto 4.3.2.1, de la misma manera el costo de implementación, el cual es el costo que se refiere al material que se llega a usar durante la implementación del sistema, y por último el costo de elaboración del proyecto que es el gasto que se llega a cubrir ya sea en servidores, equipos informáticos entre otros.

Según Boehm (1981), el costo total es la suma del costo de software de desarrollo y el costo de elaboración del proyecto, que se puede observar en la tabla 4.14.

Tabla 4.14 Costo total de proyecto

| Descripción | Costo Total (\$us) |
|-----------------------------------|--------------------|
| Costo de desarrollo de software | 14400 |
| Costo de Despliegue | 0 |
| Costo de elaboración del proyecto | 390 |
| Total | 14790 |

Fuente: Elaboración propia

4.3.2. Análisis de Beneficios

Los beneficios para el presente proyecto son de tipo intangible, donde en el sitio web Utecno (2020), menciona que el análisis de beneficios, sirve para tomar decisiones sobre un proyecto de inversión al evaluar su rentabilidad.

4.3.2.1. Valor Neto Actual

El Valor Neto Actual (VAN), es un identificador financiero que mide los flujos de los futuros ingresos y egresos que tendrá un proyecto, para determinar, si luego de descontar la inversión inicial, quedaría alguna ganancia. Si el resultado es positivo o da como resultado cero, el proyecto es viable.

$$VAN = -Inv + \sum_{j=1}^n \frac{F_j}{(1+i)^j}$$

Donde:

- VAN: Valor Actual Neto
- F: Ingreso de flujo anual, flujo neto efectivo
- Inv: Inversión en el proyecto
- n: Numero de período
- i: Tasa de descuento o tasa de interés al préstamo

A continuación, en la tabla 4.15 según el valor obtenido del VAN tomaremos criterios de decisión respecto a la rentabilidad del proyecto.

Tabla 4.15 Criterio de interpretación del VAN

| Valores del VAN | Criterio de decisión |
|-------------------|--|
| VAN > 0 | La inversión producirá ganancias por encima de la rentabilidad dada. El proyecto es aceptable |
| VAN = 0 | La inversión no producirá ni pérdidas ni ganancias. En vista que el proyecto no suma valor monetario, se debe tomar en cuenta otros factores para obtener ganancias de la misma. |
| VAN < 0 | La inversión producirá ganancias por debajo de la rentabilidad dada. Dado esta condición el proyecto debe rechazarse ya que se obtendrá pérdidas del mismo. |

Fuente: Elaboración propia

Entonces para el actual proyecto estimamos como inversión inicial \$us 14790, donde esperamos producir ganancias en un periodo de 7 años, se utilizará una tasa de descuento del 10%. Estos datos lo reflejamos en la tabla 4.16.

Tabla 4.16 Datos para el cálculo del VAN

| Datos | Valores |
|----------------------------|---------|
| Número de periodos Anuales | 5 |
| Tipo de periodo | Anual |
| Tasa de descuento | 10% |

Fuente: Elaboración propia

A la vez estimamos unas ganancias anuales en la tabla 4.17.

Tabla 4.17 Estimación de ganancias anuales

| Detalle | Periodo | | | | | |
|--------------------------------|------------|----------|----------|----------|----------|----------|
| | 0 | 1 | 2 | 3 | 4 | 5 |
| Flujo neto efectivo proyectado | -14.790,00 | 3.350,00 | 3.750,00 | 3.950,00 | 4.330,00 | 4.650,00 |

Fuente: Elaboración propia

A continuación en la tabla 4.18, realizamos el cálculo del VAN según su respectiva formula,

$$VAN = -Inv + \sum_{j=1}^n \frac{F_j}{(1+i)^j}$$

Tabla 4.18 Calculo del VAN

| Periodo = j | FNE | (1+i)^j | FNE/(1+i)^j |
|-------------|------------|---------|-------------|
| 0 | -14.790,00 | | -14.790,00 |
| 1 | 3.350,00 | 1,10 | 3.045,45 |
| 2 | 3.750,00 | 1,21 | 3.099,17 |
| 3 | 3.950,00 | 1,33 | 2.967,69 |
| 4 | 4.330,00 | 1,46 | 2.957,45 |
| 5 | 4.650,00 | 1,61 | 2.887,28 |
| VAN | | | 167,05 |

Fuente: Elaboración propia

En vista que el VAN = 167,05 y este es mayor a cero y considerando los criterios de decisión de la tabla 5.7, concluimos que el proyecto es rentable con una tasa de descuento del 10%.

4.3.2.2. Tasa Interna de Retorno

La tasa interna de retorno TIR es la tasa de descuento TD de un proyecto de inversión que permite que el VAN sea igual a la inversión. TIR es la máxima TD que puede tener un proyecto para que sea rentable.

$$TIR = VAN \cong 0$$

Realizando un cálculo por tanteo tendremos la tasa de interés de retorno buscando el dato que se acerque a cero como vemos en la tabla 4.19 y con la ayuda de la figura 4.2 apreciaremos la proximidad máxima a cero.

Tabla 4.19 Calculo por tanteo TIR

| Tasa de interés | VAN |
|-----------------|-----------|
| 0,0% | 5.240,00 |
| 2,0% | 4.032,77 |
| 4,0% | 2.933,04 |
| 6,0% | 1.928,88 |
| 8,0% | 1.009,90 |
| 10,0% | 167,05 |
| 10,4% | 6,90 |
| 10,5% | -32,71 |
| 16,0% | -1.979,26 |
| 18,0% | -2.587,81 |
| 20,0% | -3.151,40 |
| 22,0% | -3.674,27 |
| 24,0% | -4.160,17 |

Fuente: Elaboración propia

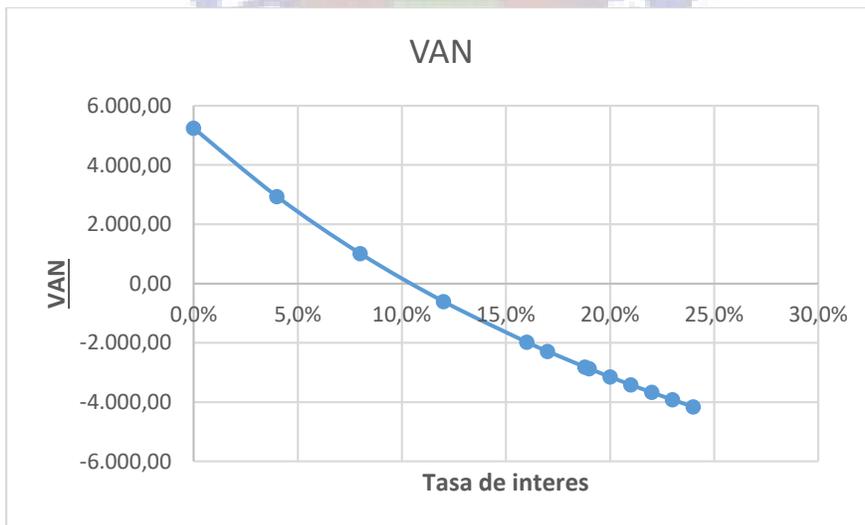


Figura 4.2 Calculo por tanteo TIR

Fuente: Elaboración propia

Se aprecia que el valor que se aproxima a cero es de 10.4% el cual resulta como candidato al TIR. Analizando el resultado obtenido, se concluye que la elaboración del proyecto es rentable ya que el valor de la tasa de descuento es relativamente mínimo, también se llega a concluir que el beneficio neto actualizado será mayor a la tasa de descuento ya que se tendrá resultados mayores a los de la inversión.

4.3.3. Costo / Beneficio

Para hallar el costo/beneficio de un proyecto se aplica la siguiente ecuación:

$$\text{Costo/beneficio} = \sum_{j=1}^n F_j / \sum \text{Costos}$$

En la tabla 5.9 contamos con los datos necesarios para ser reemplazado a la anterior formula.

$$\text{Costo/beneficio} = 20030 / 14790$$

$$\text{Costo/beneficio} = 1.35 \text{ \$us}$$

Interpretando este resultado concluimos que por cada dólar invertido en el proyecto de software se genera una ganancia de 1.35 \$us.

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

El Sistema Web de Seguimiento y Localización de Personal de Servicio de Delivery se llega a desarrollar en su totalidad tomando en cuenta los objetivos planteados según los requisitos de la institución.

A continuación, se describirá el cumplimiento y el desarrollo de los objetivos específicos:

- Mediante el desarrollo de una aplicación para dispositivos celulares, con sistema operativo Android, obtenemos la georreferenciación como se detalla en la tabla 3.19, del Capítulo 3. La aplicación captura la latitud y longitud del sistema GPS que contiene el dispositivo celular, estos datos son enviados al sistema web para luego ser desplegados en un mapa como se muestra en la figura 3.25, además para que sea identificado el usuario repartidor, se implementó un formulario de autenticación como se ve en la figura 3.27.
- Se capturan el punto de recolección y entrega del paquete, conlleva a facilitar la ubicación exacta al cual se dirigirá el usuario repartidor. Desde la aplicación web de que es usado por el usuario cliente, al momento de proporcionar los datos necesarios de registro del paquete se le pide que indique en un mapa los puntos de recolección y entrega del paquete obteniendo es esta manera su georreferenciación como se muestra en la figura 3.24 del Capítulo 3, estos puntos serán desplegados en la aplicación móvil como se puede apreciar en la figura 3.29.

- El usuario cliente contiene una cuenta para realizar la solicitud de envío de paquete, para obtener el servicio el usuario cliente debe realizar su registro desde la aplicación web abierta y desplegada para la comunidad como se puede observar en la figura 3.8.
- Se le proporciona información detallada del personal asignado figura 3.37 para el proceso de la solicitud de envío de paquete al usuario cliente, promoviendo de esta manera la confianza del servicio prestado. En el desarrollo no solo se captura la ubicación personal repartidor, esto quiere decir que cada paso realizado por el usuario repartidor se está registrando en el sistema, para luego notificarlo al usuario cliente, para ello se cuenta con un historial de pedidos figura 3.36.
- Para un seguimiento del paquete en el momento de su recolección y entrega, se implementa su registro por medio de un código QR figura 3.35 y figura 3.40, que debe contener el paquete para ser transportado, esta información se le notifica al sistema de administración como también al usuario cliente en su historial de paquetes
- Se implementó también la generación de reportes en el cual detalla información sobre el paquete, cliente, repartidor, fecha y hora del registro del paquete. A la vez se puede aplicar filtros de búsqueda ya sea por un rango de fechas, o filtrar por alguna palabra clave que este desplegado en el detalle del reporte, por último, el reporte tiene la posibilidad de ser exportado en archivos pdf o de Excel.

Por lo anteriormente expuesto, contamos con un sistema que realiza el seguimiento del personal repartidor, como también se conoce el momento en que este último recolecta y entrega paquetes, se localiza al personal repartidor en todo el momento de su hora laboral, facilitando una información inmediata para el proveedor del servicio como para sus clientes.

5.2. Recomendaciones

A partir del presente trabajo se propone las siguientes recomendaciones

- Como se puede notar hoy en día el auge que tiene los deliverys de comida rápida, el sistema propuesto llega a acoplarse para realizar tal control y no solo estar enviando encomiendas, sino llegar a abarcar este otro tipo de negocio.
- Con el uso de la tecnología, actualmente se puede llegar a romper las barreras de lo impredecible, es por ello que también se puede enfocar a distintos tipos de control como ser ciudades digitales, realizando controles de semáforos o incluso control de vías, etc., siempre y cuando se realice una inversión más grande en cuanto a los usos de API que proporciona Google.



BIBLIOGRAFÍA

- Boehm, D. (1981). *Software Engineering Economics*. Prentice-Hall
- Boehm, D. (1996). *Ada and Beyond: Software Policies for the Department of Defense*. National Academy Press.
- Castellón, G. y Velásquez, R. (2012). “*Desarrollo de aplicación de geolocalización con teléfonos celulares con sistema operativo Android utilizando el simulador SDK 1.6 y Eclipse 3.7.*” Universidad nacional Autónoma de Nicaragua UNAN-MANAGUA Recinto Universitario Rubén Darío Facultad de Ciencias e Ingenierías Departamento de Tecnología, Mangua, Nicaragua.
- DeveloperAndroid, (2020). Arquitectura de la Plataforma Android. Recuperado 7 de julio 2020, de Developer Android website: <https://developer.android.com/guide/platform?hl=es-419>
- Ecured, (2020). Metodología de Desarrollo de Software. Recuperado 10 de agosto 2020, de ecured website: https://www.ecured.cu/Metodologias_de_desarrollo_de_Software
- EllisLab, (2020). CodeIgniter. Recuperado 10 de agosto 2020 de Wikipedia website: <https://en.wikipedia.org/wiki/CodeIgniter>
- Galiano, L. (2020). Metodología UWE Aplicada (2020). Recuperado 10 de agosto 2020, de elproyecto website: <http://elproyectodeluisgaliano.blogspot.com/2012/11/metodologia-uwe-aplicada-mi-solucion.html>
- Gallego, A. (2012). Plataforma Android, Recuperado 10 de julio 2020, de Jtech website: <http://www.jtech.ua.es/dadm/restringido/android/sesion01-apuntes.html>
- GPS, (2020). Recuperado 11 agosto 2020, de Wikipedia website: <https://es.wikipedia.org/wiki/GPS>
- Guachalla, F. (2015). “*Plataforma de Geolocalización de personas mediante dispositivo móvil*” Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales, Carrera Informática. La Paz, Bolivia.

- ISO, (2020). ISO/IEC 9126-1:2001 *Software Engineering – Product quality*. Recuperado 15 de agosto 2020, de iso.org website: <https://www.iso.org/standard/22749.html>
- Kniberg, H., (2007). *Scrum y XP desde las Trincheras, 1ra Edición*, USA: C4Media Inc
- Leteriel, P., Penades, P. (2006). *Metodologías ágiles para el desarrollo de software: eXtremme Programming (XP)*. Buenos Aires, Argentina: Técnica Administrativa
- Mamani, D. (2016), “*Sistema web de seguimiento y control de supervisores y obras basado en la geolocalización en dispositivos móviles caos: Agencia Estatal de Vivienda*”. Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales, Carrera Informática. La Paz, Bolivia.
- Omicrono-EE, (2020). Del astrolabio al GPS: Historia de la Geolocalización I. Recuperado 12 de agosto 2020, de El español website: https://www.elespanol.com/omicrono/tecnologia/20130421/astrolabio-gps-historia-geolocalizacion/22247913_0.html
- Omicrono-EE, (2020). Del astrolabio al GPS: Historia de la Geolocalización II. Recuperado 12 de agosto 2020, de El español website: https://www.elespanol.com/omicrono/tecnologia/20130425/astrolabio-gps-historia-geolocalizacion-ii/8249301_0.html
- Palenque, W. (2018). *Plan de negocios para la creación de una empresa de Courier y Delivery en zonas estratégicas de la ciudad de La Paz*. Universidad San Francisco de Asis, Carrera de Administración de Empresas. La Paz, Bolivia.
- Pérez, A. (2014) “*Software móvil de geolocalización para la banca en la ciudad de La Paz*” Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales, Carrera Informática. La Paz, Bolivia.
- Piattini, M. (2020). Gestión de Requisitos para el Desarrollo de Software. Recuperado 20 de julio 2020, de Virtual.itca.edu website: https://virtual.itca.edu.sv/Mediadores/stis/22___gestin_de_requisitos_para_el_desarrollo_de_software.html

- Pressman, R. (2010) *Ingeniería de Software un Enfoque Práctico, 7th Edición*, México: McGraw-Hill
- Sánchez, F, (1999) *Medida del Tamaño Funcional de Aplicaciones de Software*, Universidad de Castilla – La Mancha, Escuela Superior de Informática de Ciudad Real
- Somerville, I. (2011) *Ingeniería de Software, 9no Edición*, México: Pearson Educación
- Utecno, (2020). *Mezcla de Mercadotecnia Análisis de Coste-Beneficio*. Recuperado 15 de septiembre 2020 de Utecno: <https://utecno.files.wordpress.com/2013/09/anacc81lisis-costo-beneficio.pdf>
- UWE, (2020). *UML-based QWeb Engineering*, Recuperado 7 de agosto 2020 de uwe website: <https://uwe.pst.ifi.lmu.de>

DOCUMENTOS