

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

DESARROLLO DE UNA APLICACIÓN MÓVIL DE TRANSPORTE Y MOVILIDAD PARA LOS USUARIOS DE LA COMUNIDAD UTN, UTILIZANDO IONIC 3.0 COMO FRAMEWORK DE DESARROLLO DE SOFTWARE

Trabajo de grado presentado ante la ilustre Universidad Técnica del Norte previo a
la obtención del título de Ingeniero en Sistemas Computacionales

Autor:

Bertil Alberto Tandayamo Lanchimba

Director:

Msc. Vicente Alexander Guevara Vega

Ibarra - Ecuador

2019



UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA

**AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD
TÉCNICA DEL NORTE**

1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información

DATOS DEL CONTACTO	
CÉDULA DE IDENTIDAD:	1004453633
APELLIDOS Y NOMBRES:	TANDAYAMO LANCHIMBA BERTIL ALBERTO
DIRECCIÓN:	CAYAMBE
EMAIL:	batandayamol@utn.edu.ec
CELULAR:	0979626724
DATOS DE LA OBRA	
TÍTULO:	DESARROLLO DE UNA APLICACIÓN MÓVIL DE TRANSPORTE Y MOVILIDAD PARA LOS USUARIOS DE LA COMUNIDAD UTN, UTILIZANDO IONIC 3.0 COMO FRAMEWORK DE DESARROLLO DE SOFTWARE
AUTOR:	TANDAYAMO LANCHIMBA BERTIL ALBERTO
FECHA:	09/09/2019
TÍTULO POR EL QUE OPTA:	INGENIERÍA EN SISTEMAS COMPUTACIONALES
DIRECTOR:	MSC. GUEVARA VEGA VICENTE ALEXANDER

2. CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y la desarrollo sin violar los derechos de autor de terceros, por lo tanto, la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de esta y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 9 días del mes de septiembre de 2019

EL AUTOR:

A handwritten signature in blue ink, appearing to read 'Bertil Alberto Tandayamo', is written over a horizontal dotted line.

Bertil Alberto Tandayamo

C.I 1004453633



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

Ibarra, 09 de septiembre de 2019

CERTIFICACIÓN DIRECTOR DE TESIS

En mi calidad de tutor del Trabajo de Grado presentado por el egresado **BERTIL ALBERTO TANDAYAMO LANCHIMBA** para otra por el Título de Ingeniería en Sistemas Computacionales con tema: **DESARROLLO DE UNA APLICACIÓN MÓVIL DE TRANSPORTE Y MOVILIDAD PARA LOS USUARIOS DE LA COMUNIDAD UTN, UTILIZANDO IONIC 3.0 COMO FRAMEWORK DE DESARROLLO DE SOFTWARE.** Considero que el presente trabajo reúne los requisitos y méritos suficientes para ser sometido a la presentación y publica y evaluación por arte del tribunal examinador que se designe.

En la ciudad de Ibarra, a los 09 días del mes de septiembre de 2019

Ing. Alexander Guevara, Msc.

DIRECTOR

AUTORÍA

Yo, BERTIL ALBERTO TANDAYAMO LANCHIMBA, portador de la cedula de ciudadanía número 1004453633, declaro bajo juramento que el trabajo aquí es descrito es de mi autoría, **DESARROLLO DE UNA APLICACIÓN MÓVIL DE TRANSPORTE Y MOVILIDAD PARA LOS USUARIOS DE LA COMUNIDAD UTN, UTILIZANDO IONIC 3.0 COMO FRAMEWORK DE DESARROLLO DE SOFTWARE**, que no ha sido previamente presentado para ningún grado, ni calificación profesional, y que se ha respetado las diferentes fuentes y referencias.



.....

Bertil Alberto Tandayamo

C.I 1004453633

DEDICATORIA

*“Mira que te mando que te esfuerces y seas valiente no temas ni desmayes porque yo
estoy contigo...”*

Josué 1:9

El presente trabajo está dedicado a mis padres Luis Alberto y María Eloísa por todo ese amor incondicional, el esfuerzo y el cariño que le han dedicado para que Bertil pudiera estudiar una carrera profesional y enfrentar la vida de otra manera diferente a la que ustedes lo hicieron.

Viejitos todo su esfuerzo está plasmado aquí me faltaría la vida y la otra para agradecerles nada más que puedo honrarles y dedicarles este reconocimiento es por ustedes. Aunque para mí no sea un escalón más o “una meta más” esto recién comienza y gracias por poner la cimiento para que algún día pueda decir meta cumplida.

Bertil Tandayamo L.

AGRADECIMIENTOS

“Si no lo intentas nada sucede, y si lo intentas al menos tienes esperanzas”

Willard Carroll Smith Jr.

A mi tutor Msc. Alex Guevara gracias por depositar su entera confianza en mí y mucho más en este proyecto, aunque había momentos en que ni yo mismo lo hacía;
por sus enseñanzas impartidas durante todo este tiempo.

Los docentes quienes mediante su conocimiento académico y valores supieron impartir sus clases, hoy su enseñanza se ve reflejada en la persona en que me he convertido.

A esta casa universitaria que me supo abrir las puertas y enseñarme el valor de estar sentado en una de sus aulas.

A toda mi familia a mi primo Patricio, mis tíos Ricardo y Pastora me supieron dar la mano en todos esos momentos difíciles.

Bertil Tandayamo L.

Gracias totales

Tabla de contenidos

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD	II
DEDICATORIA	VI
AGRADECIMIENTOS.....	VII
ÍNDICE DE FIGURAS	XI
ÍNDICE DE CUADROS.....	XIII
ÍNDICE DE FÓRMULAS.....	XIV
RESUMEN	XV
ABSTRACT	XVI
INTRODUCCIÓN	XVII
Antecedentes	XVII
Situación Actual.....	XVII
Prospectiva.....	XVIII
Planteamiento del Problema	XVIII
Objetivos	XX
Alcance.....	XX
Justificación.....	XXI
Contexto	XXII
CAPÍTULO I.....	1
MARCO TEÓRICO DE GESTIÓN	1
1.1. TRANSPORTE.....	1
1.1.1. Evolución	1
1.2. MOVILIDAD	2
1.2.1. Movilidad sustentable	2
1.2.2. Auto Compartido	3
1.2.3. Futuro de la Movilidad	3
1.2.4. Aplicaciones Móviles	4
1.2.5. Iniciativas	4
1.3. APLICACIONES MÓVILES.....	5
1.3.1. Nativas.....	5
1.3.2. Híbridas	6
1.3.3. Web progresivas.....	7
1.4. TIPOS DE CONEXIÓN A INTERNET.....	9
1.4.1 E	9
1.4.2 GPRS.....	9
1.4.3 3G	9
1.4.4 4G LTE.....	10
1.4.5 5G	10
1.4.6 Conexiones externas de los dispositivos.....	11

1.5	TIPOS DE DISPOSITIVOS MÓVILES.....	12
1.5.1	Handheld PC.....	12
1.5.2	Teléfono móvil.....	13
1.5.3	Personal digital assistant.....	13
1.5.4	Web-enabled phone.....	13
1.5.5	Smartphones.....	14
1.5.6	Tablet PC.....	14
1.5.7	Tableta.....	14
1.6	DESARROLLO DE APLICACIONES MÓVILES.....	15
1.6.1	Arquitectura.....	15
1.6.2	Lenguajes de desarrollo.....	17
1.6.3	Frameworks de desarrollo.....	19
1.6.4	Metodologías.....	21
1.7	GNOSEOLOGÍA DE LAS APLICACIONES MÓVILES Y LA MOVILIDAD.....	33
	PERIODO.....	33
	DESCRIPCIÓN.....	33
1.8	ESTÁNDAR Y BUENAS PRÁCTICAS.....	37
1.8.1	Estándar para la calidad de software ISO 25010.....	37
1.8.2	Scrum como marco de trabajo.....	42
1.9	IONIC COMO FRAMEWORK DE DESARROLLO.....	45
CAPÍTULO II.....		47
DESARROLLO.....		47
2.1.	PROCESO DE COMPARTIR UN VIAJE EN AUTOMÓVIL.....	47
2.2.	PLANEACIÓN DEL PROYECTO.....	50
2.2.1.	Definición de requerimientos.....	50
2.2.2.	Definición del product backlog.....	55
2.2.3.	Definición de roles del proyecto.....	56
2.2.4.	Planificación de Sprints.....	57
2.2.5.	Desarrollo de los Sprints.....	61
CAPÍTULO III.....		75
VALIDACIÓN DE RESULTADOS.....		75
3.1	PRUEBAS.....	75
3.2	ANÁLISIS E INTERPRETACIÓN DE RESULTADOS.....	81
3.3	ANÁLISIS DE IMPACTOS.....	88
CONCLUSIONES.....		90
RECOMENDACIONES.....		91
GLOSARIO DE TÉRMINOS.....		92
BIBLIOGRAFÍA.....		94
ANEXOS.....		104

ANEXO 1. IMPLEMENTACIÓN Y DESPLIEGUE DE ARQUITECTURA DE SOFTWARE	104
1.1 INSTALACIÓN DE AMBIENTES DE TRABAJO	104
1.1.1 Ionic 3.9.2	105
1.1.2 Cordova	106
1.1.1 Android SDK & Android Studio	106
1.2 EJECUCIÓN DEL PROYECTO	108
1.2.1 Creación	108
1.2.2 Ejecución	109
1.2.3 Proceso de ejecución de una aplicación en ionic	111
ANEXO 2. INTERFACES DE APLICACIÓN MÓVIL	112
ANEXO 3. VALIDACIÓN DE RESULTADOS	115
3.1 INSTALACIÓN HERRAMIENTA APACHE JMETER	115
3.2 SET UP DE AMBIENTE DE PRUEBA	116

ÍNDICE DE FIGURAS

Figura 1.- Árbol de problemas.....	XIX
Figura 2.- Evolución del transporte.....	2
Figura 3.- Movilidad sustentable.....	2
Figura 4.- Auto compartido mediante aplicación móvil.....	3
Figura 5.- Aplicaciones móviles.....	4
Figura 6.- Aplicaciones móviles.....	7
Figura 7.- Híbridas y progresivas.....	9
Figura 8.- Red 5G.....	10
Figura 9.- Velocidad y Movilidad de sistemas inalámbricos.....	11
Figura 10.- Dispositivo envía señal inalámbrica.....	11
Figura 11.- Figura de un PDA.....	13
Figura 12.- Figura de un PDA.....	15
Figura 13.- Arquitectura Orientada a Servicios.....	16
Figura 14.- Aplicación de tres niveles.....	17
Figura 15.- Ciclo de vida del software en modelo cascada.....	21
Figura 16.- Fases de desarrollo de la metodología RAD.....	22
Figura 17.-Ciclo de vida del AUP.....	24
Figura 18.- Metodología Crystal.....	25
Figura 19.- Ciclo de vida de CC.....	26
Figura 20.- Kanban Desk.....	27
Figura 21.- Ciclo de vida de Extreme Programming (XP).....	32
Figura 22.- Ciclo de vida de la metodología Scrum.....	45
Figura 23.- Proceso de auto compartido – caso beneficiario.....	48
Figura 24.- Proceso de auto compartido – caso conductor.....	49
Figura 25.- Arquitectura del sistema.....	62
Figura 26.- Diagrama entidad relación base de datos.....	63
Figura 27.- Publicación de aplicación en Play Store.....	64
Figura 28.- Test de normalidad. Validación de datos.....	84
Figura 29.- Gráfica de normalidad. True C1.....	85
Figura 30.- Gráfica de normalidad. False C1.....	85
Figura 31.- Gráfica de normalidad. True C2.....	86
Figura 32.- Gráfica de normalidad. False C2.....	86
Figura 33.- Gráfica de normalidad. True C3.....	87
Figura 34.- Gráfica de normalidad. False C3.....	87
Figura 35.- Gráfica de normalidad. True C4.....	88
Figura 36.- Gráfica de normalidad. False C4.....	88
Figura 37.- Descarga de Visual Studio Code.....	104
Figura 38.- Instalación de Visual Studio Code.....	104
Figura 39.- Descarga de Node.js.....	105
Figura 40.- Instalación Ionic Framework.....	105
Figura 41.- Instalación de cordova.....	106
Figura 42.- Descarga de herramienta Android Studio.....	107
Figura 43.-Instalación Android Studio.....	107
Figura 44.- Configuración de Android SDK Tools.....	108
Figura 45.-Creación de proyecto Ionic.....	108
Figura 46.- Ejecución de proyecto.....	109
Figura 47.-Proyecto visualización en modo navegador.....	110
Figura 48.- Carátula para Play Store.....	112
Figura 49.- Inicio de sesión / Login.....	112
Figura 50.- Home usuario modo pasajero.....	112
Figura 51.- Universidad origen o destino.....	113

Figura 52.- Seleccionar destino.....	113
Figura 53.- Seleccionar lugares de paso	113
Figura 54.- Detalles adicionales publicar ruta	113
Figura 55.- Viajes disponibles para este día	114
Figura 56.- Buscar viajes por lugares	114
Figura 57.- Viajes reservados usuario pasajero	114
Figura 58.- Detalles de viajes reservados	114
Figura 59.- Descarga de herramienta Apache JMeter	115
Figura 60.- Ventana inicial de Apache JMeter	115
Figura 61.- Creación de template	116
Figura 62.-Thread Group.....	116
Figura 63.- Heap de conexión Apache JMeter	117
Figura 64.- Ejecución en modo consola	118
Figura 65.- Archivo .csv con resultados de prueba.....	118

ÍNDICE DE CUADROS

Tabla 1.- Comparativa aplicaciones móviles nativas, híbridas y progresivas	8
Tabla 2.- Definición de ciclo de vida Crystal Clear	27
Tabla 3.- Prácticas de Kanban.....	29
Tabla 4.- Valores y principios de la metodología XP	30
Tabla 5.- Gnoseología de las aplicaciones móviles en la movilidad	37
Tabla 6.- Características de la norma ISO/IEC 25010	40
Tabla 7.- Métricas para evaluación en la característica Eficiencia de Desempeño.....	41
Tabla 8.- Valoración para las historias de usuario	50
Tabla 9.- Historia de usuario H101.....	51
Tabla 10.- Historia de usuario H102.....	52
Tabla 11.- Historia de usuario H103.....	52
Tabla 12.- Historia de usuario H104.....	53
Tabla 13.- Historia de usuario H105.....	53
Tabla 14.- Historia de usuario H106.....	54
Tabla 15.- Historia de usuario H107.....	54
Tabla 16.- Historia de usuario H108.....	55
Tabla 17.- Product Backlog.....	56
Tabla 18.- Roles del proyecto.....	57
Tabla 19.- Sprint 1	58
Tabla 20.- Sprint 2	58
Tabla 21.- Sprint 3	59
Tabla 22.- Sprint 4	60
Tabla 23.- Sprint 5	61
Tabla 24.- Diseño y prototipado de aplicación	68
Tabla 25.- Gestión de creación de rutas	69
Tabla 26.- Gestión de mapas	70
Tabla 27.- Gestión de imágenes	71
Tabla 28.- Gestión de reserva de rutas	72
Tabla 29.- Visualización de rutas	73
Tabla 30.- Gestión de usuario	74
Tabla 31.- Posibles combinaciones de conjuntos de datos.....	76
Tabla 32.- Iteraciones para pruebas	77
Tabla 33.- Conjunto 1. Número de estados (true/false) en los procesos	78
Tabla 34.- Conjunto 2. Número de estados (true/false) en los procesos	79
Tabla 35.- Conjunto 3. Número de estados (true/false) en los procesos	79
Tabla 36.- Conjunto 4. Número de estados (true/false) en los procesos	80
Tabla 37.- Media de todos los conjuntos.....	80
Tabla 38.- Descripción de datos procesados.....	84

ÍNDICE DE FÓRMULAS

Ecuación 1.- Fórmula para obtener la cantidad de procesos.....	77
Ecuación 2.- Definición de ecuación de medias.....	78
Ecuación 3.- Calculo de medias	78

RESUMEN

Este proyecto tiene como finalidad desarrollar una aplicación móvil destinado para transporte y movilidad para la comunidad universitaria en modalidad de auto compartido. Un conductor o dueño de automóvil publica una ruta, y por otra parte el pasajero visualiza las rutas que están disponibles y reserva una ruta. De esta forma se están compartiendo la misma ruta que trazo el conductor y de igual forma le sirvió al pasajero que la reservó.

Mediante la norma IOS/IEC 25010 se aplica el principio de eficiencia de desempeño, se propone una aplicación desarrollado bajo ciertos normas y métricas a fin de que el proceso a resolver sea más eficiente.

Antecedentes acerca de la movilidad y como estas han ido evolucionando de la mano de las aplicaciones móviles.

Aplicaciones y dispositivos móviles. Evolución de las redes móviles y como estas han permitido la conexión.

Ionic como framework de desarrollo para aplicaciones móviles híbridas. Este framework permite desarrollar aplicaciones robustas con fragmentos de código nativo permitiendo el acceso de hardware del teléfono y con la versatilidad de páginas web (HTML5, CSS) para la vista.

Desarrollado con metodologías ágiles SCRUM y con normas y estándares de calidad ISO/IEC 25010.

Se valida la eficiencia desempeño con pruebas que comprueban las tareas ejecutadas y no ejecutadas con la herramienta Apache JMeter, como herramienta de análisis y performance de plataforma seguido de una validación de resultados mediante método estadísticos y de probabilidad (mean, median, std deviation).

Palabras clave: Apache JMeter, Ionic, Oracle Apex, Apex Rest, Movilidad, Transporte, Scrum, aplicación móvil, aplicaciones móviles, Android

ABSTRACT

This project aims to develop a mobile application to transport and mobility to the university community in a car-sharing way. A driver or car owner publishes a route, the other hand passenger visualizes routes that are available and reserves a route. In this way, they are sharing the same route that the driver drew and in the same way, it served the passenger who booked it.

Through the ISO / IEC 25010 standard performance efficiency prime is applied, an application developed under certain standards and metrics is proposed in order to make the process to be more efficient.

Background about mobility and how these have evolved in the hands of mobile applications.

Mobile applications and devices. Evolution of mobile networks and how they have allowed the connection.

Ionic as a development framework for hybrid mobile applications. This framework allows developing robust applications with fragments of native code allowing access to the phone's hardware and with the versatility of web pages (HTML5, CSS) for the view.

Developed with agile SCRUM methodologies and with ISO / IEC 25010 quality standards and standards

Performance efficiency is validated with tests that verify the tasks executed and not executed with the Apache JMeter tool, as a platform analysis and performance tool followed by a validation of results using statistical and probability methods (mean, mediate, std deviation).

Keywords: Apache JMeter, Ionic, Oracle Apex, Apex Rest, Mobility, Transport, Scrum, mobile application, Android

INTRODUCCIÓN

Antecedentes

Por el excesivo crecimiento vehicular y el alto grado de contaminación ambiental se constituye a la congestión vehicular como uno de los problemas más críticos que afectan a nuestro planeta, causando efectos devastadores y adversos para el hombre, animales, medio ambiente y biodiversidad, generados por la quema de combustibles fósiles (carbón, gas o petróleo) que emiten CO₂ y otros gases de efecto invernadero, principal responsable del calentamiento global y cambio climático. (La Hora, 2015)

En el año 2016 (Bolaños, 2016) Nelson López ex jefe de tránsito de la provincia manifestó que en la ciudad el crecimiento demográfico y por ende el parque automotor va en aumento por lo cual se hace más complejo el mantenimiento del orden en el tránsito vehicular.

Todo esto se enfoca en el malestar que genera en los conductores la saturación y caos vehicular, así como falta de políticas y estrategias en generar una cultura al cuidado ambiental misma que no se está tomando en cuenta actualmente, este contexto se ajusta y va de acuerdo con los objetivos de desarrollo sostenible promovidos por la UNESCO y la ONU publicados en mayo de 2016.

Situación Actual

En la ciudad de Ibarra es claro el aumento del parque automotor, esto se puede notar en las horas pico cuando la visibilidad de automóviles crece exponencialmente, los conductores se muestran molestos y esperan soluciones por parte de las autoridades. (Botsman & Rogers, 2012)

En el sector de la UTN (Avenida 17 de Julio y calle general José María Córdova) se ha determinado que en horas pico desde las 6:00 hasta las 7:00 y desde las 12:30 hasta las 14:30 la cantidad de vehículo automotor puede llegar a 2500 unidades, creando congestión y caos vehicular. Por lo tanto, cuando se habla de unidades automotores esta inmediatamente ligada a la contaminación ambiental, ya sea por el tiempo de uso o un incorrecto mantenimiento que este tenga.

Prospectiva

Con lo expuesto anteriormente se plantea resolver el problema mediante un concepto ya conocido como economía colaborativa; en donde las plataformas digitales establecen un marco para que cada uno de los usuarios puedan interactuar entre ellos en una misma plataforma (PNUD (Programas de las Naciones Unidas para el Desarrollo), 2017). Una aplicación móvil nativa será la tecnología que permitirá que este concepto sea posible adaptarlo al entorno.

Con la implementación de la aplicación en la comunidad universitaria, aportará al cumplimiento del objetivo 11 en los incisos: 11.2, 11.6 de la Agenda 2030 y los Objetivos de Desarrollo Sostenible una oportunidad para América Latina y el Caribe publicado por la UNESCO y ONU en mayo de 2016(Naciones Unidas/CEPAL, 2016) y de igual forma va de acuerdo con el Objetivo No. 1.- Estrategia de territorio Nacional del Plan Nacional de Desarrollo 2017-2021 Todo una Vida publicado por la Secretaria Nacional de Planificación y Desarrollo – SENPLADES Ecuador en septiembre de 2017, en la cual hace énfasis en la solución de problemas con tecnologías de innovación.

La propuesta determinará, mediante el desarrollo de una aplicación móvil híbrida utilizando métricas de calidad de software, solucionar la problemática de la comunidad universitaria mediante vehículo compartido hecho a la medida para el uso de la comunidad universitaria, controlando de esta forma la seguridad de cada uno de los usuarios (personal, docentes y estudiantes).

El proyecto podrá ser implementado en las demás universidades de la región norte, creando una red de servicios de transporte compartido entre IES. El beneficio obtendrá ambas partes, ahorrando tiempo en el trayecto por del pasajero y el conductor estará aportando a la minimización de contaminación ambiental como se consigue esto reduciendo el tráfico vehicular.

Planteamiento del Problema

El 8,40% de la comunidad universitaria poseen un vehículo, en el cual viaja solo el conductor, y se está desperdiciando espacio dentro del vehículo mismo que

se puede aprovechar compartiendo con el resto de los usuarios quienes se dirigen también al mismo sitio

Tecnológico

Al momento en el mercado no existe una aplicación puramente académica que permita la movilización mediante auto compartido, especificando estándares que brinden la calidad de software que el usuario demanda en la actualidad.

Ambiental

La comunidad universitaria no hace concientización la forma en que hace uso del automóvil, esto conlleva a que no haya participación de cuidado del medio ambiente, lo que está colaborando al incremento de CO₂ en el medio ambiente. El ingreso de las unidades vehiculares en horas pico, es de 1200 unidades, lo cual genera congestión y caos. Un automóvil que se dirige hacia la universidad en promedio tiene 3.933333333 km de recorrido desde un punto promedio, llegaría a generar 2.27 kg de CO₂.

Social

La ausencia de la cultura de colaboración entre compañeros de trabajo y estudio para el bien común.

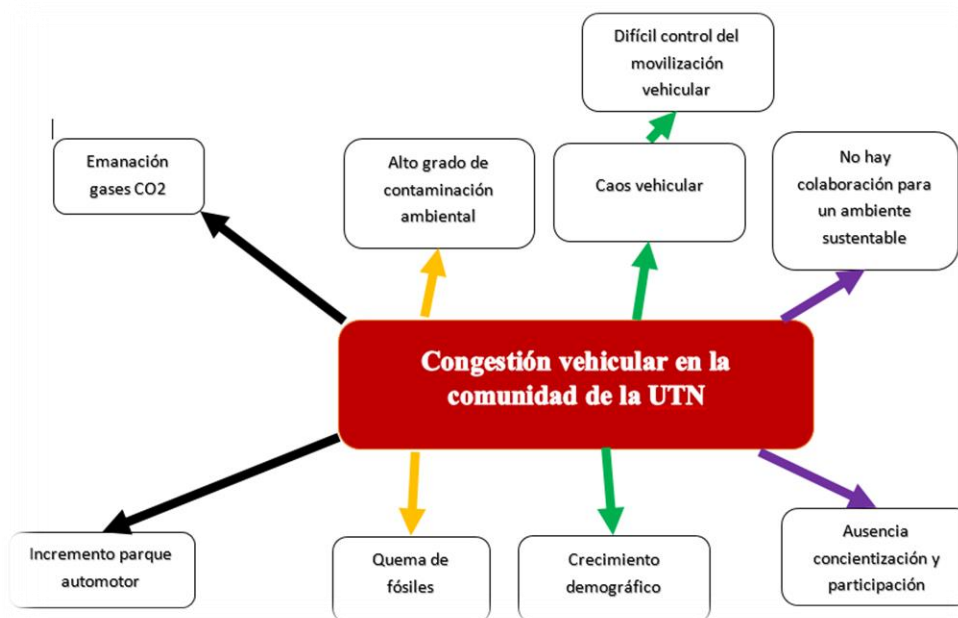


Figura 1.- Árbol de problemas

¿Como se puede tener una mejor movilidad vehicular en la comunidad de la UTN?

Objetivos

Objetivo General

Desarrollar una aplicación móvil que contribuya al mejoramiento de la movilidad y transporte para los usuarios en la comunidad universitaria.

Objetivos Específicos

1. Crear un marco teórico sobre las aplicaciones móviles nativas, enfatizando en el framework de desarrollo Ionic 3.0
2. Establecer indicadores que permitan evaluar la calidad del software basados en la característica: eficiencia de desempeño de la norma ISO 25010.
3. Implementar una aplicación móvil nativa que permita fortalecer la movilidad vehicular en la comunidad UTN.
4. Validar y evaluar mediante un modelo matemático-estadístico el impacto que se obtendrá con la aplicación móvil.

Alcance

La implantación de esta herramienta estará dirigida a la comunidad universitaria; para el registro y uso de la aplicación el método de validación será la cuenta de email institucional, atributo habilitante para el acceso será tener esta cuenta activa.

Contará con los siguientes módulos:

Autenticación

La aplicación será de uso exclusivo de la comunidad de la UTN, haciendo uso de las mismas credenciales utilizados en el portafolio A para administrativos, D docentes y E estudiantes.

Gestión de rutas

El conductor podrá crear y eliminar una ruta, de igual forma revisar las solicitudes de pasajeros que hayan escogido esa ruta.

Para la creación de las rutas se estableció realizar con la herramienta de Google Maps, para el trazo de rutas, selección de lugares y posterior revisión de ruta.

Por otro lado, el pasajero podrá reservar y/o cancelar la ruta que más le convenga, pudiendo revisar todos los detalles de esa ruta y de igual forma los datos del conductor.

Justificación

Social

El presente proyecto va a tener un enfoque hacia los objetivos de desarrollo sostenible planteado por la ONU y UNESCO: Ciudades y Comunidades Sostenibles (Naciones Unidas/CEPAL, 2016) literal

11:2” De aquí a 2030, proporcionar acceso a sistemas de transporte seguros, asequibles, accesibles y sostenibles para todos y mejorar la seguridad vial, en particular mediante la ampliación del transporte público, prestando especial atención a las necesidades de las personas en situación de vulnerabilidad, las mujeres, los niños, las personas con discapacidad y las personas de edad” y

11:6 “De aquí a 2030, reducir el impacto ambiental negativo per cápita de las ciudades, incluso prestando especial atención a la calidad del aire y la gestión de los desechos municipales y de otro tipo”

De igual forma se relaciona al Plan del Buen Vivir (SENPLADES, 2017) promulgado en la Constitución de la República del Ecuador, haciendo hincapié en el acceso a transporte sostenible y calidad ambiental: Objetivos Nacionales de Desarrollo, Eje 1: Derechos para Todos Durante Toda la Vida, Objetivo 1: Garantizar una vida digna con iguales oportunidades para todas las personas , en la cual está orientado al bienestar común de las personas garantizando la seguridad de los pasajeros e impulsar la movilidad inclusiva, alternativa y sostenible.

Tecnológico

Las aplicaciones móviles nativas son más convenientes a diferencia de híbridas, permiten aprovechar los recursos de un dispositivo de mejor manera; pero al ser nativas requieren que sean desarrolladas en entornos propios del sistema operativo para el cual va a ser ejecutada lo que recaen. Por el tiempo dispuesto para la elaboración de este tema de tesis se ha establecido realizar con una herramienta que permita el desarrollo de aplicaciones híbridas.

Social

Se aporta de igual forma a la convivencia en la comunidad universitaria, creando relaciones de compañerismo colectivo. La aplicación lo que busca es que la universidad sea participe en la movilidad sustentable a través de una aplicación móvil nativa que minimizar el impacto de CO₂ en el medio ambiente.

Metodológica

La entrega del producto debe ser lo más práctica y rápida posible, por la tanto se ha elegido una metodología de desarrollo de software ágil específicamente SCRUM para que el desarrollo y avance se realicen permanente de acuerdo con los Sprint que se establezca.

Contexto

En el repositorio bibliográfico de la universidad no existen proyecto alguno que este enfocado a la movilidad y transporte sostenible mediante aplicaciones móviles.

CAPÍTULO I

MARCO TEÓRICO DE GESTIÓN

1.1. TRANSPORTE

Metodología de movilización que el hombre utiliza en el traslado de personas y/o bienes de un territorio hacia otro, al igual que en la comunicación. Se utiliza para el acto consecuencia de trasladar algo de un lugar a otro. (RAE, 2019)

Considerado como actividad del sector terciario, permite el crecimiento económico. El transporte facilita el intercambio comercial entre las regiones y los países, favoreciendo a las actividades económicas. (Torres, 2013)

1.1.1. Evolución

El transporte es tan antiguo como la misma civilización. En sus inicios se realizaba a pie y usando la fuerza de los animales por lo que se necesitó domesticarlos.

Hubo la necesidad de transportar objetos con mayor volumen cuyo peso el animal no podía soportar, (Cardona, 2016) con esto supuso la creación y posterior impulso de la rueda. Este ha sido uno de los inventos más revolucionarios de la historia, y en la actualidad seguimos utilizando diariamente.

Posteriormente apareció la bicicleta, (Torres, 2013) dando origen a la motocicleta y después al automóvil. El hombre en la búsqueda del aparato que lo ayude a transportarse de una forma rápida y cómoda; en el 1882 descubrió el petróleo lo refinó y de eso salió el combustible eso dio cabida a que el automóvil vea la luz.

Con la primera guerra mundial la transportación se incrementó y así surgieron los autobuses y la gran industria del motor existente, hoy entre ellos se encuentra el ferrocarril, metro, etc.

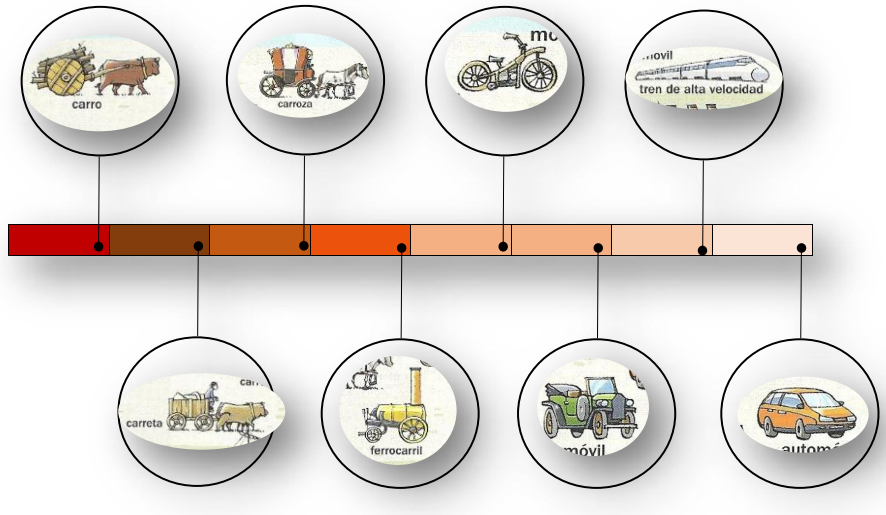


Figura 2.- Evolución del transporte

1.2. MOVILIDAD

Es un parámetro para medir el desplazamiento que realiza un objeto (personas o bienes) producidos en un entorno físico. Se utilizan diferentes medios de transporte para el desplazamiento con el objetivo de facilitar la accesibilidad a determinados lugares.(UPV, 2012)

1.2.1. Movilidad sustentable

También conocida como movilidad sostenible, es un concepto para definir a una movilidad más amigable con el medio ambiente teniendo un mínimo impacto ambiental, es decir con una huella de carbono más reducida que a cualquier otro medio de transporte tradicional. (UPV, 2012) Este significado comprende varios



Figura 3.- Movilidad sustentable

enfoques; el primero es reducir el número de vehículos que circulan, logrando así disminuir el impacto ambiental que esta actividad conlleva.

1.2.2. Auto Compartido

Es una práctica que consiste en compartir viajes; más de una persona puede viajar en un solo automóvil previniendo la necesidad de sacar todos cada uno de sus vehículos. Al estar en rodaje una solo unidad se evita el costo de combustible, peajes y el estrés de conducir.(Deloach & Tiemann, 2012)

Es una modalidad donde se comparte un automóvil con otras personas en viajes periódicos y para trayectos exacto. (Wauters, 2014) Esta actividad persigue como objetivo reducir la congestión de tránsito especialmente en las ciudades grandes de igual forma facilitar el desplazamiento de las personas que no tengan automóvil propio.



Figura 4.- Auto compartido mediante aplicación móvil

Fuente: bit.ly/2Usj7am

1.2.3. Futuro de la Movilidad

Tiempo atrás cuando alguien quería dirigirse a su puesto de trabajo tenía que elegir entre caminar, ir en bicicleta, transporte público o auto propio. En la actualidad existen un gran abanico de posibilidades para transportarse.

No es incierto lo que este persigue, ya que todos apuntan a un solo objetivo, transportar al mayor número de personas con la menor cantidad de vehículo automotor circulando en las calles. (Corwin, Vitale, Kelly, & Cathles, 2015) Cada vez

más son nuevas iniciativas que salen a la luz presentándose como alternativas para un transporte más sostenible.

1.2.4. Aplicaciones Móviles

También conocido como apps móviles. Es un programa o aplicación informática diseñada para ejecutarse en teléfonos inteligentes y otros dispositivos móviles. (Dredge, 2013) Estas aplicaciones originalmente fueron creadas para ser una ayuda con el email, calendario y los contactos del teléfono, pero la demanda de estas aplicaciones causó una expansión hacia otras áreas como juegos, automatización, mapas, tracking, tickets, por lo tanto, en la actualidad existen millones de estas disponibles. Se encuentran disponibles para diferentes sistemas operativos móviles Android, iOS, BlackBerry OS, Windows Phone, entre otros.



Figura 5.- Aplicaciones móviles

Fuente: bit.ly/2TVaTTQ

1.2.5. Iniciativas

En esta nueva era de digitalización han surgido numerosos servicios de transporte entre los que se destacan:

Free floating. – Alquiler de bicicletas o scooters sin estaciones. El mundo de las bicicletas está en clara expansión junto a los cambios que deben producirse, aparecen nuevos servicios y nuevas formas para el uso de la bici. Mediante un sistema de alquiler de bicicletas sin necesidad de estaciones. Son bicicletas distribuidas por toda la ciudad y equipadas con un GPS y una app móvil permite su

localización y desbloqueo a través de una señal bluetooth desde el móvil del usuario. (Kortum, Schönduwe, Stolte, & Bock, 2016) La empresa Lime es un claro ejemplo.

Car pooling. – servicio de viajes compartidos. La empresa francesa Bla bla Car es pionera en esta modalidad. Esta empresa permite compartir viaje con la persona dueña del automóvil. A diferencia de servicio de transporte privado este aprovecha la ruta que esté realizando.

Car sharing. - uso de temporal de vehículos. Amovens es una empresa española que presta este servicio. La empresa como tal no posee flota de automóviles utiliza los vehículos de otras personas que quieran rentar y del otro lado están los usuarios que van a hacer uso.

Ride sharing. - servicio de transporte privado. Empresas como Uber Lyft, Cabify están en delantera. A diferencia del carpooling esta metodología es más de un servicio privado al igual que un taxi.

Todas estas son iniciativas que ya forman parte de nuestra vida diaria constituyéndose alternativas al transporte público tradicional. (Aratani, 2016)

MARCO TEÓRICO DE TECNOLÓGICO

1.3. APLICACIONES MÓVILES

1.3.1. Nativas

Se llaman aplicaciones nativas por que se desarrollan a medida, es decir una aplicación específica para un sistema operativo. De esta manera se aprovecha mucho mejor los recursos de hardware que posee el teléfono como GPS, galería de fotos, etc. Con rendimiento optimizado y su interfaz mucho más representativa de acuerdo con el sistema operativo. En cuanto a usabilidad y diseño y eficiencia son los mejores y los que el mercado siempre prefiere.

De acuerdo con la tabla 1, una desventaja es muy costosa desarrollar una aplicación de este tipo, ya que es desarrollada específicamente para un solo sistema operativo (Android, iOS o Windows Phone) y no puede reutilizar ese código para el desarrollo multiplataforma.

Ventajas

- Acceso completo al dispositivo
- Mejora experiencia de usuario
- Envío de notificaciones a usuarios
- Actualización de aplicaciones constante

Desventajas

- Diferentes herramientas para el desarrollo
- Desarrollo más caro
- Código fuente no es reutilizable

1.3.2 Híbridas

Enfoque de programación para dispositivos que combina tecnologías web con contenedores nativos propios con acceso a la API del dispositivo y programación nativa. Basados en tecnologías web HTML5, CSS y JavaScript, junto con frameworks que permiten desplegar cualquier tipo de vista web en un dispositivo móvil. Como ventajas de esta tecnología se tiene que su desarrollo va dirigido para aplicaciones multiplataforma ejecutándose de forma nativa en el dispositivo. (González Verdugo, 2016)

A diferencia de una aplicación nativa todos sus componentes se utilizarán de una forma menos efectiva, haciendo que estos se cargan de una forma más lenta a diferencia que se realizaría en una aplicación nativa. Pero como gran ventaja con tecnología híbrida es multiplataforma y se puede todo el proceso de realizar todo desde cero.(G. Martinez, 2017)

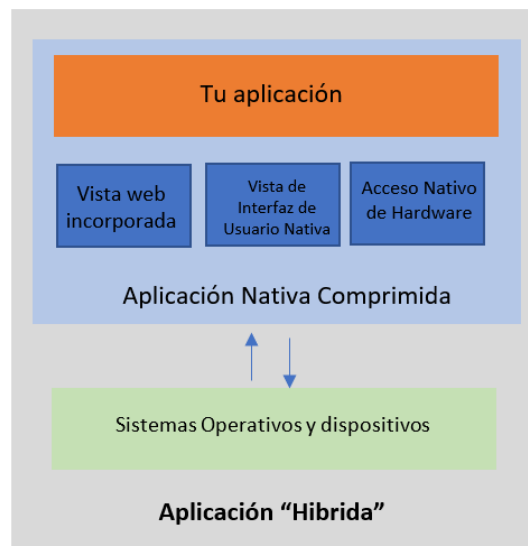


Figura 6.- Aplicaciones móviles

Fuente: bit.ly/2TVaTTQ

1.3.3 Web progresivas

Básicamente Progressive Web Apps (PWA) es un sitio web que utiliza tecnología web moderna pero que actúa y se siente como una aplicación móvil real. (Otemuyiwa, 2017) Este término lo acuñaron (A. Russell, 2015). El precursor más interesado en esta tecnología ha sido Google, asegurándose que las PWA's brinden una experiencia de aplicación nativa.

Mediante esta tecnología que permite mostrar la experiencia de una aplicación móvil a través de un navegador web. Se visualizarán de una manera óptima en el navegador web, sin que haya la necesidad de que sea instalada en un dispositivo. (Cuesta, 2018). Se muestra sin ninguna novedad en todo solo dispositivos móviles (Tablets, laptops) y ordenadores de mesa. (Frankston, 2018)

- **Características principales de las PWA**

Hay ciertas características que van a definir su existencia

- Responsive. - La interfaz de usuario debe acomodarse a todos los dispositivos escritorio, móvil, tablets, etc.
- Independencia de conectividad. - en áreas de conectividad baja deberá ser capaz de trabajar sin conexión.
- Instalable. - se puede añadir al home screen como un icono e iniciar cuando quiera utilizar la aplicación.

- Semejante a una app. - cuando se esté interactuando con la web app, no debe sentirse la diferencia como al momento de usar una aplicación nativa.
- Compartir. – fácil de compartir mediante un URL.

	NATIVA	HIBRIDA	WEB
Lenguaje	JAVA, C, .NET	HTML, CSS, JavaScript	HTML, CSS, JavaScript
Coste Desarrollo	X	—	✓
Interfaz Usuario	✓	✓	—
Rendimiento	✓	—	X
Multiplataforma	X	✓	✓
Tiempo de Desarrollo	X	—	✓
Tienda de Apps	✓	✓	—

Tabla 1.- Comparativa aplicaciones móviles nativas, híbridas y progresivas

Fuente: bit.ly/2CPOZOx

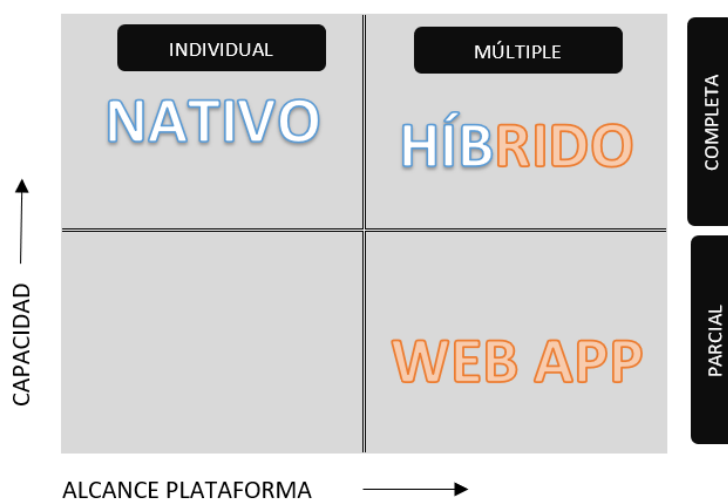


Figura 7.- Híbridas y progresivas

Fuente: bit.ly/2CP0ZOx

1.4 TIPOS DE CONEXIÓN A INTERNET

1.4.1 E

Enhanced data Rates for GSM Evolution, es un tipo de conexión que “permite alcanzar velocidades de hasta 384 kbps y recibir datos móviles pesados, como grandes archivos adjuntos de correo electrónico o navegar por páginas web complejas a gran velocidad” actuando como puente para las dos redes 2G y 3G a través de una actualización opcional de software realizada opcional.(Blasco, 2016)

1.4.2 GPRS

General Packet Radio Service versión mejorada del GSM, con la premisa de permanecer siempre y en cualquier lugar conectados, esta tecnología brinda “mensajería instantánea, los servicios de mensajes cortos (SMS) y multimedia (MMS) y de correo electrónico”. En velocidad de transferencia esta permite desde 56 a 114 kbps 30 mensajes por minuto a diferencia de la otra tecnología que permite entre 6 y 10 mensajes de texto.(Nouh et al., 2017)

1.4.3 3G

Esta tecnología se había implementado en Chile en el 2007 cuando Erasmo Rojas mencionó que “esta red móvil que en algún momento se pensó que era sólo para hablar, hoy puede ayudar a disminuir la brecha de información”.(Blasco, 2016)

Con mayor estabilidad y seguridad que la 2G, permite una transferencia de archivos mucho más rápida que una conexión normal, con una velocidad extensa que llega hasta los 2 Mbps lo que antes era algo imposible y lejano de concebir.

1.4.4 4G LTE

Long Term Evolution (LTE), mientras que en la mayoría de los países está establecida esta tecnología en otras todavía no se establece, según OpenSignal. Uruguay casi a la par de Singapur es el país donde el 84% del territorio posee esta tecnología.(Konovalov & Sychugov, 2018)

1.4.5 5G

Llamada la quinta generación, países pioneros (EE. UU, Japón, Corea del Sur, China) en esta tecnología prevén que para finales de este año sea el lanzamiento en dichos países. Para Latinoamérica se especula que su lanzamiento sea para el 2022.(Blasco, 2016)



Figura 8.- Red 5G

Fuente: (Wray, 2018)

Se dice que su velocidad será mucho más rápida (100 veces) a diferencia de tecnologías anteriores (4G, H+). Según, (Wray, 2018) “en comparación, 5G promete ofrecer velocidades superiores a 1 Gb / s (1000 Mbit / s), con muchas estimaciones que lo acercan a 10 Gb/s”. (Wray, 2018) Descarga y reproducción de juegos y películas será algo instantáneo sin buffering. De igual forma la realidad virtual y aumentada serán nuevas aplicaciones disponibles en esta tecnología.

1.4.6 Conexiones externas de los dispositivos

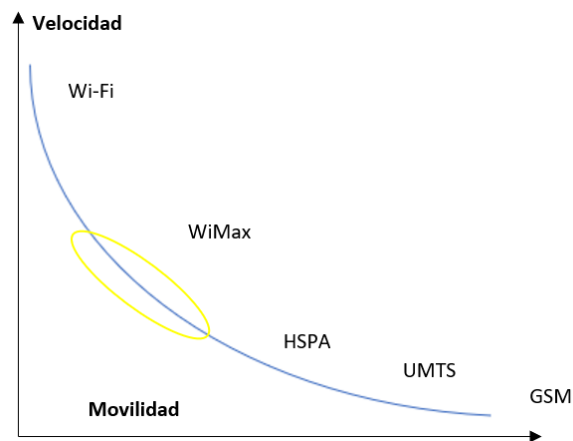


Figura 9.- Velocidad y Movilidad de sistemas inalámbricos

Fuente: bit.ly/2KM80UE

- **Wifi**

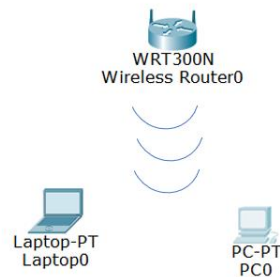


Figura 10.- Dispositivo envía señal inalámbrica

Fuente: bit.ly/2KOfyGq

Tecnología de red inalámbrica local, que permite conectarse entre dispositivos móviles y a internet a través un punto de red inalámbrica. De acuerdo a Alianza wi-fi es quien permite que se cumpla y certifique mediante el estándar 802.11 (BBVA & BBVA, 2015).

De acuerdo a este último el estándar que da a conocer las especificaciones (control de accesos a medios y la capa física), para brindar comunicación en redes inalámbricas de área local “en las frecuencias de 2.4,3.6,5 y 60 GHz, creados y mantenidos por el comité de estándares de IEEE LAN” (Kiernan, 2014)

- **Wimax**

Es un estándar que proporciona múltiples opciones de conexión inalámbrica de acuerdo con el estándar IEEE 802.16. este término se acuñó por el evento Wimax Forum realizado en el año 2011. Evento realizado para la interoperabilidad del estándar de acuerdo a (Walton, 2015) se describe a Wimax como “una tecnología basada en estándares que permite la entrega de acceso de banda ancha inalámbrica de última milla como alternativa al cable y DSL”.

Mediante varios dispositivos permite esta tecnología brindar ancho de banda móvil. Proporciona una línea de pago para suscripción de pago cable en digital para acceso de ancho de banda de última milla.

- **Lifi**

Tecnología inalámbrica que va a utilizar luz eléctrica de una casa, hogar para transmisión de datos y también es posible por luces ultravioletas e infrarrojas. Según, (Haas, 2013) es un “sistema de comunicaciones de luz visible capaz de transmitir datos a altas velocidades sobre espectro de luz visible”.

A diferencia del Wifi que utiliza frecuencia de radio para su transmisión esta utiliza luz eléctrica, que por cierto tiene demasiadas ventajas como mayor ancho de banda, trabajos en áreas susceptibles a interferencia.(Tsonev, Videv, & Haas, 2013). De igual forma usa los mismos protocolos que el wifi el 802.11, con la diferencia en su forma de recibir la señal.

1.5 TIPOS DE DISPOSITIVOS MÓVILES

Este término abarca un amplio rango de dispositivos de electrónicos de consumo, normalmente se entiende como un dispositivo que se puede conectar a internet, haciendo inclusión de las cámaras digitales y reproductores mp3 y mp4.(Wray, 2018)

1.5.1 Handheld PC

Conocidos comúnmente con clamshell¹ aparecieron antes de los PDAs. Permitía a los usuarios ejecutar aplicaciones financieras, científicas y de datos de forma local, además de su funcionalidad principal la calculadora. Su batería tenía un largo tiempo de vida debido a los chips de bajo consumo, de igual forma al no tener

¹ En ingles cubierta

partes móviles soportaba bien los golpes (Tilley, 2016). Con la aparición de los PDAs cada vez más potentes, pequeños y eficientes el mercado estaba bajo presión.

1.5.2 Teléfono móvil

Estos representan el punto de partida para llegar en primera instancia a los web-enabled phones y luego a los smartphones. Con un microprocesador que realiza cálculos a gran velocidad, gestiona los comandos y controla las señales de la estación base además de coordinar las demás funciones.(Feig & Feig, 2007)

Ventajas de este dispositivo ligero, transportable, económico; y desventajas poca potencia de proceso, memoria, capacidad de visualización limitada.

1.5.3 Personal digital assistant

Más conocido como PDA, combinan elementos de un ordenador, teléfono, fax, internet en un solo dispositivo; ofrecían básicamente calendarios, blocs de notas y agendas clásicas. A diferencia de los ordenadores portátiles la mayoría de estos tenían un lápiz óptico como dispositivo de entrada en lugar de un teclado; por lo tanto, se entiende que incorporaban características de reconocimiento de escritura a mano, algunos tenían comandos de voz mediante tecnologías de reconocimiento.(Feig & Feig, 2007)



Figura 11.- Figura de un PDA

Fuente: amzn.to/2ObVHic

Pantalla táctil con un procesador, memoria y un sistema operativo permitía la conexión directa con el ordenador de sobremesa lo que posibilitaba guardar datos y exportarlos a bases de datos o transferir e instalar nuevas aplicaciones al asistente.(Viken, 2015)

1.5.4 Web-enabled phone

Dispone de una pantalla limitada con un teclado típico de 12 botones para la entrada de datos, esta experiencia hacía de los teléfonos móviles una opción muy pobre para navegar por internet debido a la cantidad de datos que puede visualizarse es muy limitada.(David & Pozo, 2015)

1.5.5 Smartphones

Permiten guardar información e instalar programas además de usarse como un teléfono móvil en un único dispositivo, un smartphone se puede considerar como un teléfono móvil con las mismas funciones de un PDA integrado en el dispositivo. Otro atractivo de estos dispositivos es que los usuarios pueden ampliar las características del dispositivo descargando nuevas aplicaciones mediante la conexión inalámbrica.(Essay UK, 2016)

Los primeros sistemas operativos para smartphones más comunes fueron Symbian OS, Palm OS, Pocket PC Phone Edition y Microsoft Smartphone 2002; en la actualidad están reinando el mercado Android, iOS y Windows Mobile (Gallelgos, 2016). Tanto los fabricantes de dispositivos móviles como los operadores de telefonía han creado y están creando mercados donde los desarrolladores pueden subir sus aplicaciones para posteriores descargas. Los usuarios pueden bajarse las aplicaciones. Como exponentes de la generación moderna de smartphones, se pueden citar el iPhone 4, el iPhone 5 y el Samsung Galaxy II.

1.5.6 Tablet PC

Es un tipo de ordenador que tiene una pantalla con la que se puede interactuar directamente; durante muchos años estos dispositivos no han sido especialmente exitosos, ya que estaban pensados para trabajos de campo como una alternativa a los portátiles. Son ligeramente más pequeños que un portátil, los datos se ingresan mediante su pantalla táctil.(David & Pozo, 2015)

1.5.7 Tableta

Estos dispositivos han dejado de ser un futuro lejano para instalarse como una realidad presente en el mercado tecnológico, el iPad de Apple como uno de sus precursores, a partir de esto fueron los fabricantes de otras marcas sacando provecho para la creación de nuevas tabletas en poco tiempo.(Gallelgos, 2016) Apple con el iPad con su App Store, pantalla multi touch y la duración de la batería. RIM lanzo el BlackBerry Playbook dispone de su sistema operativo propio, pero también es posible ejecutar aplicaciones desarrollados para Android. Samsung con su Galaxy Tab tecnología 4G y 5G por defecto para los países para el cual este desarrollado este protocolo.

1.6 DESARROLLO DE APLICACIONES MÓVILES

Para el desarrollo de una aplicación móvil de calidad se necesita tener en cuenta tres herramientas principales: la arquitectura, lenguaje de desarrollo y la metodología con el que se va a utilizar.

1.6.1 Arquitectura

Se refiere a la estructura la forma y las tecnologías que se van a utilizar para la construcción y posterior deployment² del producto. Se toman en cuenta los casos funcionales y no funcionales, los cuales toman un rol muy importante todo para el correcto funcionamiento de la aplicación. (Kim, 2013)

- **Arquitectura cliente servidor**

Consiste en conectar distintos clientes en uno o más servidores, su modelo de aplicación es distribuida. Mediante peticiones del cliente al servidor envían paquetes de información, también conocidos como proveedores de recursos.

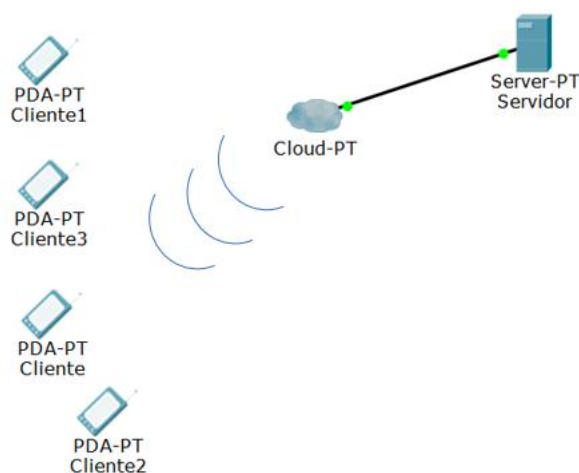


Figura 12.- Figura de un PDA

Fuente: amzn.to/2ObVHic

- **Cliente delgado**

Depende del servidor central para realizar sus tareas, todo el proceso se lo realiza en el servidor, y los clientes se los destina únicamente como terminales, que hacen de enviar y recibir datos, pero ningún proceso en sus dispositivos. (Valipour, Amirzafari, Maleki, & Daneshpour, 2009)

² Desplegar, lanzar la aplicación

- **Cliente inteligente**

A diferencia del cliente delgado aquí los terminales si pueden realizar el proceso, las consultas en el propio dispositivo móvil. Sin embargo, quienes llevan el almacenamiento son los propios servidores de bases de datos. Como ventajas se puede resaltar la reducción de transferencia de datos en la red. Pero su mayor desventaja es la sincronización entre base de datos y cliente. (Balalaie, Heydarnoori, & Jamshidi, 2016)

- **Cliente robusto**

Este cliente posee la mayor cantidad de procesos en su terminal y sus datos son transmitidos únicamente para almacenamiento, asumiendo la mayoría de la funcionalidad ellos mismo. El servidor se queda solo a carga del almacenamiento.

- **Arquitectura orientada a servicio**

Este diseño brinda servicios de forma autónoma que se compartan servicios diferentes componentes de la aplicación, mediante un protocolo de comunicación en la red. Un servicio se puede acceder de forma remota mediante una dirección URL o IP y actúa de forma independiente.



Figura 13.- Arquitectura Orientada a Servicios

Fuente: bit.ly/2zf9JNY

- **Arquitectura multicapa**

Es una arquitectura cliente servidor que se divide en capas y funcionan independientemente físicamente separadas son tres capas: de presentación, procesamiento y administración de datos. Beneficios de esta arquitectura al asegurarse tres capas independientes una de la otra, en caso de que algo no esté bien se dirige a la capa en donde se deba corregir y no volvemos a revisar toda la aplicación, creando de esta forma creando un desarrollo flexible y reutilizable. El concepto utilizado para una multicapa es el de tres capas en donde se tiene el modelo de negocio, la de datos y de presentación.(Summers, 2018)

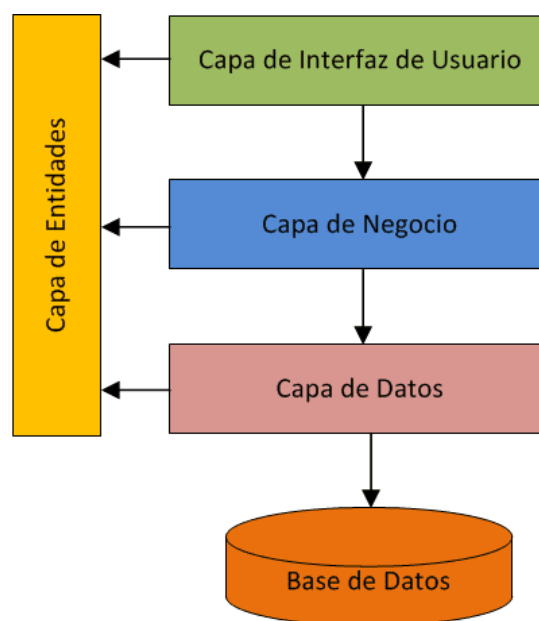


Figura 14.- Aplicación de tres niveles.

Fuente: bit.ly/2AyvzdQ

1.6.2 Lenguajes de desarrollo

Quienes van más de acuerdo con la tecnología y tipo de arquitectura que se vaya a elegir (híbrida, nativa o web), se da a conocer los siguientes lenguajes de programación para el desarrollo de aplicaciones móviles.(Evans, 2017)

- **Objective-C**

Fue introducido por el propio Apple para el desarrollo de aplicaciones en MacOS³ y iOS con la finalidad que sean aplicaciones robustas y escalables. Este permite mayor aprovechamiento del hardware que posee se logra poseer mayor visualización precisamente en gráficas de entrada y salida (Cox & Novobilski, 2017).

³ Sistema operativo para computadores Apple

En 2014(K. Russell, 2014) la compañía decidió lanzar una versión nueva y mejorada llamada Swift con la atención que todas las aplicaciones sean aún más eficientes y robustas.

- **Swift**

Lenguaje de programación compilado, desarrollado por Apple para MacOS, iOS⁴ y Linux. La última versión mejorada de Objective-C para desarrollo de aplicaciones móviles en iOS. Por su parte la empresa dueña de este lenguaje anima a que todos los desarrolladores de Objective-C hagan uso de esta versión mejorada para una programación completa. Recomiendan que es más segura, eliminado de errores y paradigmas que se presentaban en Objective-C.(Apple, 2015)

- **C#**

C# es para Windows Phone lo que Objective-C o Swift para iOS. Es el lenguaje de programación con orientación a objetos perfecto para el desarrollo de aplicaciones móviles en Windows Phone.(Drayton, Albahari, & Neward, 2015) Desarrollado netamente por Microsoft en versionamiento de .NET y aprobada por la ISO 23270:2016. El IDE que en la actualidad da soporte en Visual Studio en sus última versión.(Petzold, 2012)

- **Java**

Lenguaje de desarrollo orientados a objetos preferido cuando se trata para Android. Fue desarrollado por Sun Microsystem y adquirida posteriormente por la ahora Oracle (Binstock, 2015). Con código útil para reutilizar y actualizar el software. Java está considerando también enfocarse al desarrollo de aplicaciones para iOS. Al tener una máquina virtual (JVM⁵), quieren permitir ejecutar una vez y ejecutar en cualquier lugar. Su ejecución y desarrollo en la actualidad es multiplataforma lo que permite aún más rápida su escalabilidad y por ende el uso de este lenguaje de programación.(Gosling, Joy, Steele, Guy L., & Bracha, 2015)

⁴ Sistema operativo para dispositivos móviles Apple

⁵ Java Virtual Machine

- **Kotlin**

Lenguaje orientado a objetos estático para JVM y JavaScript. Admite interoperabilidad y la inteligibilidad. Este proyecto salió a la luz por la necesidad de mejorar la experiencia de codificación de una manera efectiva y práctica. Por lo que es un lenguaje mixto, además es compatible con Java y Objective-C , lo que permite que el aprendizaje de este idioma fluido (Redchets, 2017). Con una sintaxis renovada Kotlin facilita su uso a diferencia de otros lenguajes. Asimismo, con este lenguaje funciona del lado del servidor, para desarrollo web y desarrollo de escritorio.(Krill, 2015)

- **JavaScript**

Lenguaje de programación muy similar al de Java incluso en su forma de escribir, pero son dos diferentes lenguajes de programación (Krill, 2015). JavaScript por su lado está enfocado hacia el desarrollo interactivo de páginas web, de lado de los servidores con NodeJS⁶ y de igual forma trabaja con desarrollo de aplicaciones móviles híbridas juntamente con HTML5⁷ y CSS3⁸.(Haverbeke, 2014)

- **TypeScript**

Es un lenguaje de programación de código abierto desarrollado por Microsoft. Influenciado y reconfigurado por lenguajes de programación como Java, JavaScript y C#. Diseñado para el desarrollo de aplicaciones grandes. Al ser una super configuración de JavaScript su código fuente es compatible para construir y ejecutarse con JavaScript. (Rosenwasser, 2019) Puede ser usado en JavaScript para el desarrollo tanto del lado del cliente como del servidor.

1.6.3 Frameworks de desarrollo

- **Ionic**

Es un framework⁹ de código abierto para el desarrollo de aplicaciones móviles híbridas con tecnologías como CSS¹⁰, HTML5¹¹ y Sass¹². Creado en Dfrity Co. por Max

⁶ Ambiente de ejecución para JavaScript

⁷ Lenguaje para desarrollo de páginas web

⁸ Lenguaje usado en presentación de un documento html

⁹ Entorno de software universal que proporciona funcionalidad como parte del propio software

¹⁰ Hojas de estilo en cascada. Cascade Style Sheets

¹¹ Lenguaje de marcas para la creación de páginas web

¹² Lenguaje de para hojas de estilos en cascada

Lynch, Ben Sperry y Adam Bradley. La version original fue lanzada en el año 2013 y desarrollado sobre Angular JS y Apache Cordova. Las aplicaciones son desarrolladas en estas tecnologías web y para luego distribuirse a través de tiendas de aplicaciones como App Store en iOS, Store de Windows y Play Store para Android.(Lynch, 2016) Ionic utiliza complementos de Cordova para obtener acceso al hardware (cámara, GPS, linterna, etc) del teléfono móvil. Ionic permite crear e implementar aplicaciones mediante wrapping¹³ con la herramienta de Cordova con una simple línea de comando 'ionic'.

- **Xamarin**

Framework para la creación de aplicaciones móviles cross-plataform¹⁴. Permite a los desarrolladores compartir 90 por ciento del código en la mayoría de las plataformas. Está basada en el stack¹⁵ de la tecnología de Microsoft, contando con una comunidad de desarrolladores de aproximadamente 1.4 millones. (Peppers, 2014) Fue creada por los desarrolladores quienes habían creado anteriormente Mono, una plataforma de código abierto basada en el framework de .NET, liderado por Miguel de Icaza. La compañía como tal se fundó en el año de 2011.(Friedman, 2018) Xamarin utiliza un único lenguaje C#, para la creación de aplicaciones móviles, a diferencia de otros frameworks este ya viene compilado de forma nativa lo que hace que se mas fácil su operación en aplicaciones de alto rendimiento como si fuesen totalmente nativas.

- **React Native**

Es una version nativa de la librería de JavaScript, para la creación de aplicaciones móviles que son difíciles de diferenciar de las que son desarrolladas de forma nativa usando Java u Objective-C. Utiliza los mismos bloques de construcción de la interfaz de usuario que utilizan las aplicaciones normales de iOS y Android, pero el utilizando React y JavaScript (Mehul, 2018). Desarrollado por Jordan Walke un ingeniero en Facebook. Después de muchos meses de desarrollo lanzaron la primera version en el 2015.(Occhino, 2016)

¹³ Empaquetar todo el proyecto y mostrar como una aplicación móvil

¹⁴ Desarrollo para varias plataformas (Android, iOS, Windows y Web)

¹⁵ Conjunto de tecnologías para el desarrollo de un software

1.6.4 Metodologías

Lo mismo que en el desarrollo de software tradicional, el desarrollo de aplicaciones móviles toman los mismo problemas con planificación estrategias, requerimientos y todo lo que conlleva el ciclo de vida del desarrollo de software.(Stapic, Mijac, & Strahonja, 2016)

- **Modelo cascada**

Es un diseño secuencial ligeramente para ciertas áreas de diseño de la ingeniería. En el ciclo de vida de esta metodología la secuencia de desarrollo va hacia abajo a través de las bases que se detalla.(Vesela & Krbecek, 2016)

Este modelo se desarrolló cuando se dieron cuenta que, en procesos de fabricación y construcción, el coste los cambios de diseño de retroalimentación o revisión. Por lo que tenía que ser una norma ya establecida para que los procesos se realicen una sola vez sin posibilidad a revisión. (Stapic et al., 2016)

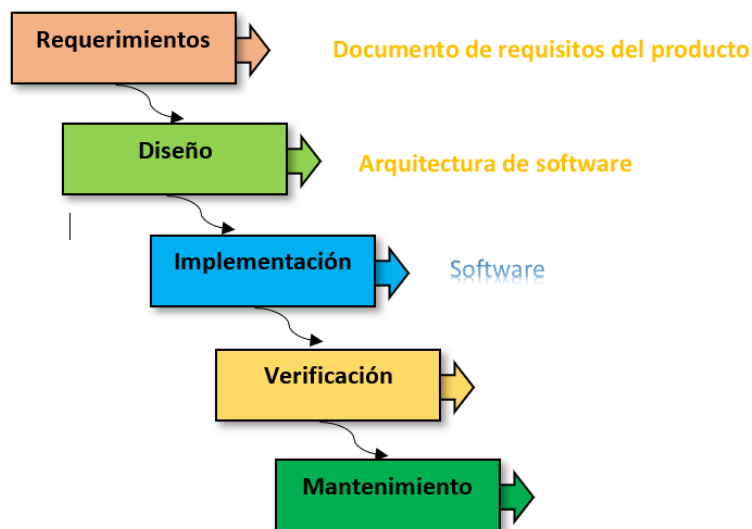


Figura 15.- Ciclo de vida del software en modelo cascada

Fuente: bit.ly/2IXwb4I

- **Desarrollo rápido de aplicaciones**

Es un método que permite el desarrollo que se caracteriza por la construcción de prototipos y el uso de utilidades CASE. Permite también a abarcar usabilidad, utilidad y rapidez de ejecución. Se enfatiza más rápido en el trabajo de software, sobre la planificación estricta y el registro de requisitos. Quiere decir

manos a la obra en el proyecto y sigamos sin tantos rodeos, y las pruebas son muchas. Al ser un desarrollo inmediato, se les resta un poco de importancia a la documentación.(Anderson, 2017)

A continuación, se presenta las fases de desarrollo que presenta este modelo

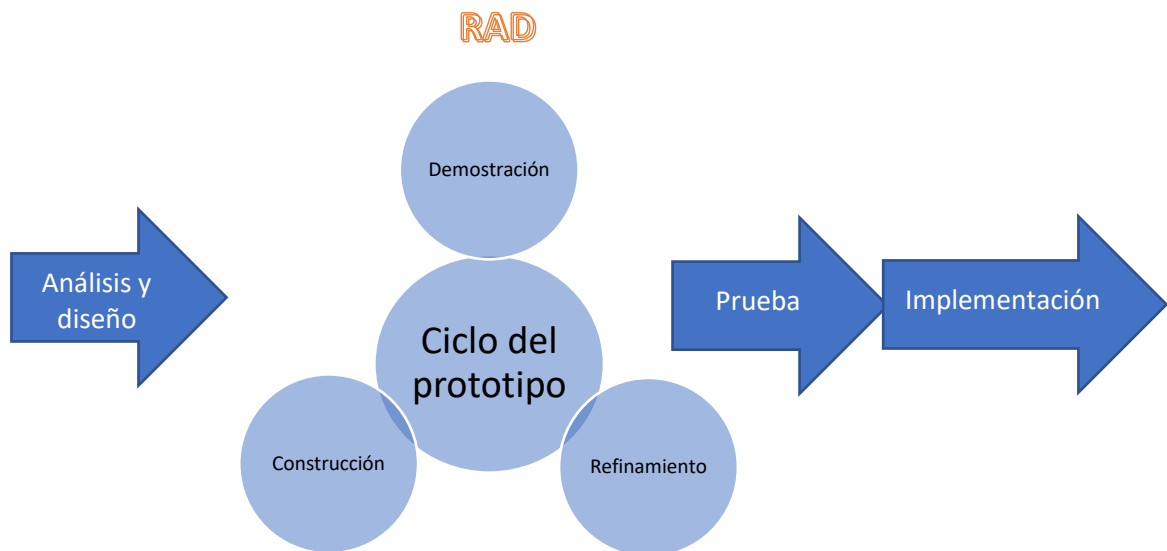


Figura 16.- Fases de desarrollo de la metodología RAD

Fuente: bit.ly/2KPDnhd

- **Metodologías ágiles**

Son métodos de ingeniería basados en desarrollo incremental e iterativo. Según la necesidad del proyecto los requerimientos solucionan y evolucionan a través del ciclo de vida de desarrollo del software. (Cockburn, 2015) De esta manera se organiza a los equipos de trabajo quienes están inmersos dentro del proyecto, en un proceso de plazos cortos. Se caracteriza por obtener la flexibilidad e inmediatez del proyecto pudiendo iterar de una manera muy fácil y ahorrar tiempo y costes.(Martínez, 2014)

Entre las metodologías ágiles más destacados están:

- **Agile Unified Process (AUP)**

Esta versión ágil del RUP¹⁶. Este aplica técnicas ágiles tales como: desarrollo dirigido por pruebas (TDD)¹⁷, modelamiento ágil (AM)¹⁸ y reestructuración de la

¹⁶ Metodología de desarrollo Rational Unified Process

¹⁷ Test Driven Development

¹⁸ Agile Modeling

base de datos para mejorar el rendimiento. Usando tecnologías ágiles este demuestra una forma más fácil de desarrollo mediante conceptos que aún se mantiene en RUP.(Chin, 2014)

El ciclo de vida se parece mucho a RUP(Christou, Ponis, & Palaiologou, 2013):

- Inception. – el equipo identifica: el alcance inicial del proyecto, la arquitectura potencial y su un porcentaje de dinero para el inicio el proyecto cláusulas que se hayan puesto de acuerdo en el contrato.
- Elaboration. - se propone la arquitectura y el equipo establece la factibilidad del sistema.
- Construction. – se inicia la construcción del proyecto y se queda en encuentros periódicos con las partes interesadas para definir las prioridades que son muy altas.
- Transition. – El objetivo es validar y desplegar el sistema ya para un ambiente de producción.

AUP se maneja por siete principios:

- Model. - identificar el problema que se da por parte del cliente y buscar una solución viable, siempre poniendo en contexto el tipo de negocio y la organización del cliente.
- Implementation. – transformar el modelo planteado en código ejecutable y desarrollar pruebas particularmente pruebas individuales.
- Test. – asegurar la calidad mediante pruebas esto incluye hallar defecto que pueden estar en su diseño como en el código fuente y verificar que se cumplan todos los requerimientos.
- Deployment. – se va a planificar la entrega del sistema y ejecutar el plan de instalación del sistema hacia usuarios finales.
- Configuration management. – hacer un correcto seguimiento al sistema en el tiempo y control del versionamiento.
- Project management. – es la directiva que se encarga de todas las actividades del proyecto. Manejar riesgos, dirigir personas con las tareas y dead-lines¹⁹, asegurándose que hayan sido entregados a tiempo y dentro del presupuesto.

¹⁹ Fecha limite

- Environment. – ayuda adicional asegurándose que el equipo tenga una orientación y las herramientas adecuadas. (hardware y software)

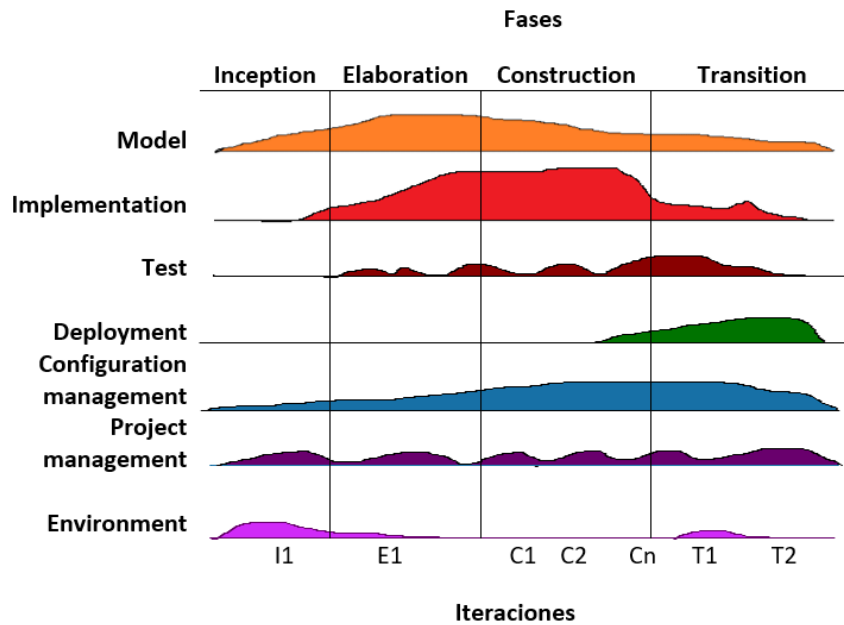


Figura 17.-Ciclo de vida del AUP

Fuente: bit.ly/2SA3lke

- **Crystal Clear (CC)**

Es una metodología ágil derivada de la rama de metodología Cristal. CC²⁰ se caracteriza por delimitar el número de participantes en el equipo de trabajo de hasta máximo 6 personas y la complejidad que el proyecto posee. Es utilizado para desarrollo de proyectos pequeños.(Chin, 2014)

²⁰ Crystal Clear

		Clear	Yellow	Orange	Red	Maroon
Críticidad del proyecto	Life(L)	L6	L20	L40	L80	L200
	Essential Money(E)	E6	E20	E40	E80	E200
	Discretionary money (D)	D6	D20	D40	D80	D200
	Comfort(C)	C6	C20	C40	C80	C200
		1-6	7-20	21-40	41-80	81-200
		Número de personas en el proyecto				

Figura 18.- Metodología Crystal

Fuente: bit.ly/2zeWsl5

Está diseñado para brindar resultados rápidos y efectivos mediante una metodología liviana, y de igual forma priorizada para proyectos seguros. (Cockburn, 2014) En esta metodología CC utilizan métodos. Las tres primeras son obligatorias para toda la familia cristal, y para CC solo las tres son obligatorias siendo las otras opcionales dependiendo como se haga uso de la metodología. (Dilamani, 2014)

- Entrega frecuente. - para metodologías ágiles las entregas rápidas son una principal característica. Para asegurarse de estar avanzando correctamente en cada iteración se debe realizar reuniones para planificar y asegurarse de que todo avance correctamente para la próxima iteración; de igual forma se medirá la velocidad y el buen desarrollo del equipo.
- Mejora planificada. - la naturaleza de desarrollo de software tiende a ser demasiado cambiante durante sus requerimientos. El equipo está presto a dar a conocer cuál es su situación actual. Permite hacer esos ajustes. Solo para hacer un poco de diferenciación entre Scrum y Crystal Clear, hablando de este último el equipo tiene siempre una reunión antes de una iteración para discutir los problemas si es que existieran cualquiera que fuese a reflejar la situación, reuniones que no tome demasiado tiempo. (Lowe, 2016)
- Comunicación gradual. - una de las propiedades más importantes en la que se enfoca esta metodología es en la comunicación entre cada uno de los integrantes. Por lo cual es muy recomendable que cada uno de los equipos tengan una comunicación que fluya hasta el fondo de la audiencia de los miembros del equipo.

- Seguridad personal. – cada miembro del equipo debe estar en la capacidad de dar a conocer sus ideas sea o no una mala idea, a través de la confianza porque mientras más en confianza tengamos, más fácil será la tarea de compartir ideas.
- Ambiente técnico con pruebas automatizadas. – herramientas de software que aún no tienen un lugar específico en todos los equipos.(Dilamani, 2014)

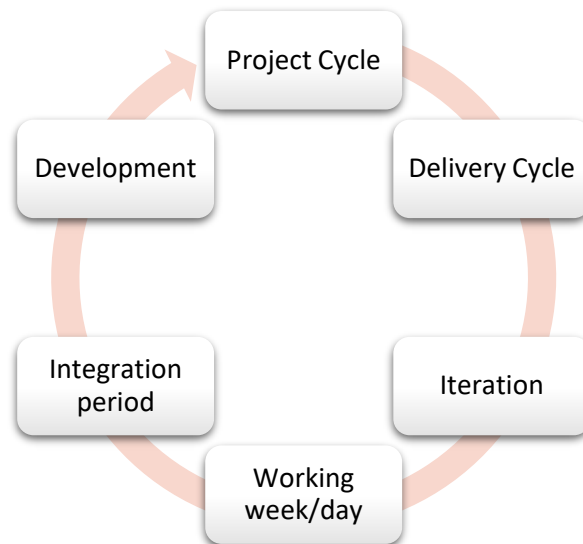


Figura 19.- Ciclo de vida de CC

CICLO DE VIDA	DEFINICIÓN
Project Cycle	El mismo proyecto es una que continúa a otro proyecto y se repite el ciclo.
Delivery Cycle	Consiste en verificar el plan de software, para poner el número de iteraciones que se va a realizar.
Iteration	Planificación iterativa Actividades diarias "Ritual de finalización" del proyecto
Working week/day	Elegir entre si realizar reuniones diarias o semanales dependiendo del formato del proyecto y del equipo.
Integration period-development	Integración y pruebas del sistema. Algunas veces los equipos en el proceso continúan las pruebas sin detener el desarrollo.

Development	La esencia, la parte central del trabajo del programador. Un equipo desarrolla una pequeña tarea, brinda solución y revisa que su configuración quede bien hecha en todo el sistema.
--------------------	--

Tabla 2.- Definición de ciclo de vida Crystal Clear

Fuente: (Cockburn, 2014)

- **Kanban**

Esta metodología nació como un sistema de para la producción de automóviles de Toyota en Japón. Consiste en una organización de trabajo sin iteraciones, con énfasis en la entrega del trabajo a tiempo, logrando simplificar el tiempo y evitando los errores por falta de información.(Stabile, 2015) Con Kanban la gestión de priorizar las tareas de acuerdo de urgencia.

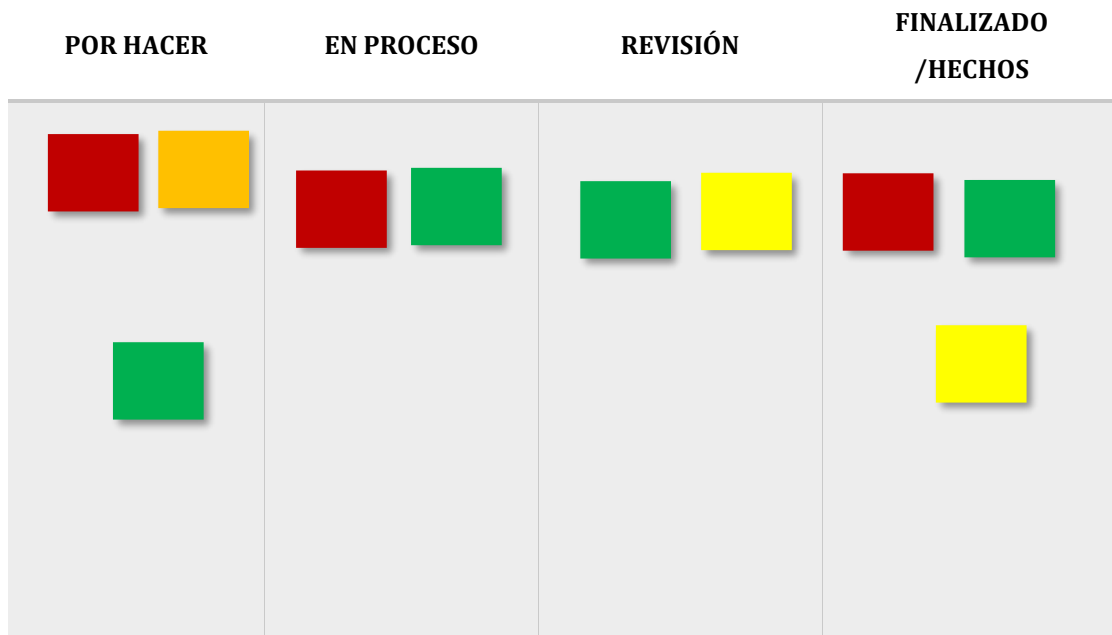


Figura 20.- Kanban Desk

Fuente: bit.ly/2CQZFuL

A inicios de siglo XXI actores clave de la industria del software se dieron cuenta que se podía hacer uso de esta metodología para cambiar positivamente las forma que se entregaban los productos y servicios. Enfocando a un mayor enfoque en la eficiencia y aprovechando los avances en la tecnología informática. Kanban

dejo de trabajar con la industria automotriz para aplicarse cada vez más a otros sectores industriales y complejos.(Wakode, Raut, & Talmale, 2015)

Kanban tiene cuatro principales núcleos, con el único fin de hacer las cosas y principios as importantes.

- Inicia con lo que tienes. - permite continuar con el flujo de trabajo con sistemas en desarrollo sin interrumpir lo que se esté haciendo, destacando los problemas que deberán ser abordados ayudando a evaluar y planificar cambios para que la implementación sea los menos disruptiva posible. Hace que Kanban sea fácil de implementar in cualquier organización como si no hubiese ningún cambio que hacer al momento de iniciar.(Wakode et al., 2015)
- Estar en constante evolución. – esta metodología hace hincapié en hacer pequeños incrementales cambios al proceso, y lo que menos se quiere es hacer cambios extremos por que pueden crear un ambiente de incertidumbre en el proyecto.
- Respeta los procesos, roles y responsabilidades actuales. – está diseñado para promover y alentar a cambios incrementales y lógicos sin provocar miedo al cambio en sí mismo. Kanban no prohíbe el cambio, pero tampoco es camisa de fuerza para no hacer al cambio.
- Incentiva el liderazgo en tu equipo. – esto es una premisa dentro de la metodología. Es importante que cada integrante del equipo tiene que estar siempre con una mentalidad de mejora continua para lograr obtener un rendimiento óptimo.

PRÁCTICAS	DEFINICIÓN
Visualizar el Workflow	Primero tener en claro los requerimientos que se del producto entregable. Visualizar todas las tareas que se vaya a realizar en un pizarrón con tarjetas representando en cada columna un paso en tu flujo de trabajo.
Acortar el trabajo en progreso	El enfoque siempre debe ser uno, cambiar a la mitad casi siempre perjudica el proceso. Una principal función de Kanban es asegurarse el número de tareas que sean realizable. Si no hay límites de trabajo en curso, eso no es Kanban.
Hacer el proceso más específico	Se trata de administrar el flujo de trabajo, la administración mas no al equipo de trabajo, enfocándose siempre en administra los procesos de trabajo y entender cómo hacer que el trabajo sea más rápido.
Feedback constante	Esta metodología apoya a las reuniones regulares son necesarias para la transferencia de conocimiento. Se llevarán a cabo en frente del tablero donde cada miembro contara ante los demás lo que hizo el día anterior y lo que va a realizar hoy. La duración promedio debe ser de 10-15 minutos dependiendo del tamaño del equipo y los temas a tratarse.
Mejorar la colaboración	Compartiendo la visión de un futuro mejor y la comprensión colectiva de los problemas que se deben superar es la mejor manera de lograr una mejora continua y un cambio sostenible dentro de la organización. Únicamente el equipo que entiende este concepto tiene más probabilidades de desarrollar una comprensión compartida del problema y por lo tanto sugerir paso para corregir el problema.

Tabla 3.- Prácticas de Kanban

Fuente:(Ferrao & Dias Canedo, 2015)

- **Extreme Programming (XP)**

Metodología de desarrollo en la ingeniería de software expuesta por Ken Beck (Wellington, 2015) con el objetivo de encontrar maneras de escribir software de alta calidad rápido y siendo fácil de ser entregables para los clientes con los cambios que sean necesarios. El más favorable de las metodologías ágiles para el desarrollo. Por lo que hace énfasis en la adaptabilidad en vez de previsibilidad que quiere decir esto, utiliza menos documentación y se enfoca mucho más en el código.(Sharma & Hasteer, 2016a) La adopción de esta metodología está catalogada

como una de las mejores, que se aplica de diferente forma en todo el ciclo de vida del software.(Borralho, Ferreira, Vieira, Queluz, & Rodrigues, 2017)

VALORES Y PRINCIPIOS	DEFINICIÓN
Communication	Es sumamente esencial ya que diariamente estamos trabajando juntos desde los requerimientos hasta la codificación, para crear la mejor solución ante el problema planteado.
Simplicity	Escribir un código simple y limpio es lo que pretende cualquier programador dándole valor al producto y ahorrando tiempo y esfuerzo dentro del equipo.
Feedback	Periódicamente el equipo hace revisión del producto, recibe feedback anotando con mucho cuidado los cambios que se vayan a hacer y así continúa mejorando. Explicaremos el proyecto que mejor se adapte al proceso que ya tiene y no es viceversa.
Respect	Dentro del equipo deberá existir respeto entre cada uno de los miembros. Cada persona tiene asignado una parte del proyecto que contribuye a un objetivo común.
Courage	Es necesario que todo el equipo objetivamente evalúe sus propios resultados sin ninguna excusa y siempre estar dispuestos a aceptar los cambios necesarios.

Tabla 4.- Valores y principios de la metodología XP

Fuente: (Sharma & Hasteer, 2016b)

Roles

- Tracker. - pregunta a cada uno de los programadores de cómo esta, escuchar con mucha cautela y tomar acción si al parecer se algo no se entiende y está perdiendo de vista el objetivo principal.
- Customer. – escribe historias y especifica pruebas funcionales. Tiene la autoridad de decidir lo que va o no en las historias de usuarios.
- Programmer. – estima lo que son historias y define las tareas de ingeniería. Estima le tiempo que va a tomar implementarse cada una de las tareas.
- Tester. – implementa y despliega pruebas funcionales, cada uno de los resultados son medidos gráficamente asegurándose que los integrantes del equipo sepan cuando los resultados de las pruebas disminuyen.

- Coach. – se aseguran de agendar las reuniones para diferentes actividades que tiene planeado el equipo registrando resultados de la reunión para informes futuros.(Sharma & Hasteer, 2016b)

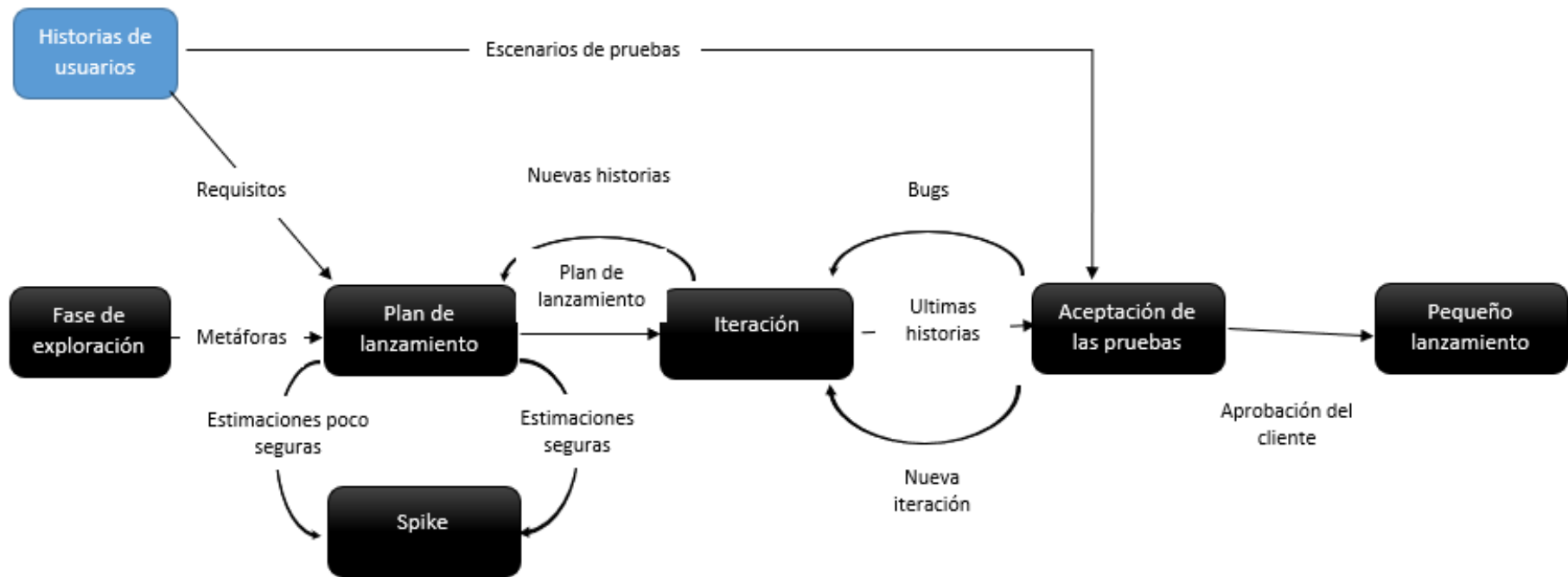


Figura 21.- Ciclo de vida de Extreme Programming (XP)

Fuente: bit.ly/2K12Lh5

1.7 GNOSEOLOGÍA DE LAS APLICACIONES MÓVILES Y LA MOVILIDAD

PERIODO	DESCRIPCIÓN
1960 - 1980	Aunque la movilidad compartida es nueva, el concepto no lo es. Este se remonta a la segunda guerra mundial cuando existían los “clubes de auto compartido”. Cuando querían conservar el caucho para la guerra, las autoridades incentivaban el viaje compartido. Posteriormente en la crisis del petróleo fue muy recomendado debido al embargo del petróleo y sus consecuencia la crisis energética con el fin de preservar el petróleo disponible.(BELZ & LEE, 2013)
1960	En Europa iniciaron los primeros programas de bike-sharing ²¹ . Las bicicletas se dejaban en lugares específicos por toda la ciudad, cualquiera podía coger y empezar a moverse hasta su destino.(Shaheen, Martin, & Cohen Apoorva Musunuri Abhinav Bhattacharyya, 2016)
1990	Internet abría nuevas posibilidades para las personas para obtener información acerca del servicio de tránsito. Las agencias de tránsito empezaron a hacer sus propios mapas disponibles en línea.(Dutzik, Madsen, & Baxandall, 2013)
1996 Onstart	Proporciona comunicaciones basadas en suscripciones, seguridad del vehículo, servicios de emergencia, llamadas, manos libres, etc.(Department of Transportation & Highway Administration, 2016)

²¹ Bicicletas que puedan utilizar varias personas

1998	En los Estados Unidos hizo su incursión en el carsharing ²² cerca de una década después emergió en Europa como tecnologías emergentes como internet lo usuarios reservaban sus autos diariamente o por horas en línea y de forma remota.(Shaheen et al., 2016)
2000 Zipcar	Proporciona reservas de automóviles a sus miembros, facturables por minuto, hora o día. Es posible que los miembros tengan que pagar una cuota de membresía mensual o anual además de los cargos por reservaciones de automóviles.(Shaheen et al., 2016)
2000 Gas Buddy	Que proporciona precios de estaciones de servicio cercanas a partir de datos enviados por el usuario, así como a través de asociaciones con otras compañías y directamente de los operadores de la estación.(Shaheen et al., 2016)
2002 Philly CarShare	Tiene autos disponibles para uso las 24 horas del día a través de la ciudad de Filadelfia y la región. (Dutzik et al., 2013)
2005 Google Maps como compañía	El verdadero potencial de la tecnología que ayudó en la navegación del transporte público fue cuando Google lanzó su primera aplicación con el mapa de tránsito en línea.(Dutzik et al., 2013)
2006 Free Map (Waze)	Proporciona información de navegación paso a paso y tiempos de viaje y detalles de ruta enviados por el usuario, mientras que descarga información dependiente de la ubicación a través de una red de telefonía móvil.(Department of Transportation & Highway Administration, 2016)

²² Compartir el viaje en un solo automóvil

<p>2008 Google Maps aplicación móvil</p>	<p>Para conducir, transporte público, carsharing, bicicletas compartidas, andar en bicicleta y caminar en más de 30 ciudades en todo el mundo.(Department of Transportation & Highway Administration, 2016)</p>
<p>2009 Uber</p>	<p>Es una compañía de redes de transporte con sede en San Francisco, California. Uber ofrece servicios que incluyen viajes compartidos entre pares, servicios de transporte, transporte de alimentos y un sistema para compartir bicicletas.(Bhuiyan, 2018)</p>
<p>2010 Globe Sherpa</p>	<p>Pagos móviles y tickets para tránsito, estacionamiento, eventos y destinos locales.(Siuhi & Mwakalonge, 2016)</p>
<p>2010</p>	<p>Un nuevo modelo emerge en los estados unidos. Carsharing de uno a uno usando el servicio de internet buscando personas interesados en prestar sus automóviles en renta.(Department of Transportation & Highway Administration, 2016)</p>
<p>2011 Citymapper</p>	<p>Para conducir, transporte público, carsharing, bicicletas compartidas, andar en bicicleta y caminar en más de 30 ciudades en todo el mundo.(Department of Transportation & Highway Administration, 2016)</p>
<p>2011, May Spinlister</p>	<p>Para compartir equipos deportivos al aire libre, específicamente bicicletas, tablas de surf, esquís y tablas de snowboard.(Siuhi & Mwakalonge, 2016)</p>

2012, June Lyft	Es una empresa de redes de transporte ofreciendo servicio de automóviles, scooters y el sistema de bicicleta compartida. Es la segunda empresa más grande de transporte con el 30% del mercado en los Estados Unidos.(O'Brien, 2018)
2012 Transit App	Diseñado para agregar y mapear datos de tránsito público en tiempo real, funcional en más de 125 áreas metropolitanas de todo el mundo.(Department of Transportation & Highway Administration, 2016)
2012 Amtrack	Se lanzó el programa nacional de venta de entradas. Entrega de boletos comprados en línea por correo electrónico. Sin embargo, han adquirido los servicios más nuevos que están tomando una forma fácil de usar con aplicaciones móviles.(Dutzik et al., 2013)
2013 RideScout	Tiene la capacidad de sincronizar calendarios personales con la aplicación de viaje para encontrar atracciones y eventos.(Shaheen et al., 2016)
2013 Nextbus	Fue uno de los primeros servicios en proveer información de tránsito para los pasajeros. Mediante su página web, aplicación móvil, mensajes de texto o alertas con llamadas telefónicas se publicaba información de cuando su bus o tren iba a llegar a la estación.(Gannes, 2013)
2014 Bridj	Permite al usuario reservar un servicio de transporte entre el hogar y el trabajo durante las horas de viaje a través de una aplicación móvil.(Johnston, 2014)

2015 hasta la actualidad	Ya en la actualidad por la concientización sobre el problema que el tráfico genera, las ciudades en los últimos han buscado nuevos modelos y sistemas que ayuden a hacer uso más eficiente del transporte como el uso de auto compartido.(Aratani, 2016)
-----------------------------	--

Tabla 5.- Gnoseología de las aplicaciones móviles en la movilidad

Fuente: Propia

1.8 ESTÁNDAR Y BUENAS PRÁCTICAS

1.8.1 Estándar para la calidad de software ISO 25010

- **Calidad de software**

Este concepto es el que verifica que tan bien cumple o se ajusta de acuerdo con un diseño en función de requisitos funcionales. De esta manera tiene competitividad en el mercado estructurado bajo un marco o un estándar. El cumplimiento de los requisitos no funcionales se refleja en el cumplimiento con la calidad de software, que respaldan la entrega de requisitos funcionales.(Galín, 2018)

- **Norma ISO 25010**

Esta norma está diseñada con el objetivo de minimizar errores humanos que puedan afectar la entrega y la concepción del proyecto. Errores que cambien las prioridades del proyecto. Estas prioridades y convertirlas en valores que se puedan medir que se adhieran a los metas del proyecto. (Birla & Johansson, 2015)

De acuerdo con el modelo de calidad que presenta la norma, se da a conocer las siguientes características:

Característica	Descripción	Sub característica	Descripción
Adecuación Funcionalidad	Son aquellas que satisfacen las necesidades presentadas en el proyecto. Idoneidad, exactitud, interoperabilidad, seguridad y cumplimiento de funcionalidad.(Miguel, Mauricio, & Rodríguez, 2014)		
Fiabilidad	Características las cuales brindan de poder mantener su operabilidad y rendimiento durante un periodo de tiempo determinado. Madurez, tolerancia a fallos cumplimiento de confiabilidad.(ISO, 2011)		
Usabilidad	Si el producto final es intuitivo y de fácil manejo para el usuario. Un conjunto de estos estará a cargo de la evaluación de esta ya sea establecidos o implícitos en el proyecto. Comprensibilidad, operabilidad, atractivo, cumplimiento de la usabilidad.(ISO, 2011)		
Eficiencia de desempeño	Esta característica mide el rendimiento y la cantidad de recursos utilizados para el desarrollo del producto, dentro de las condiciones establecidas. Utilización de recursos, comportamiento del tiempo, cumplimiento de eficiencia.(ISO, 2011)	Comportamiento temporal	Tiempos de respuesta y procesamiento y el rendimiento de un sistema cuando lleva a cabo sus funciones bajo

condiciones determinadas en relación con un benchmark²³ establecido.

Utilización de recursos	Cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función
-------------------------	--

Capacidad	Grado en que los límites máximos de un parámetro de un producto o software cumplen con los requisitos.
-----------	--

Mantenibilidad Esfuerzo mínimo que se deberá realizar al momento de crear incrementar y escalar el producto, logrando que el impacto sea en lo mínimo. Posibilidad de cambiar, escalabilidad estabilidad, cumplimiento de la capacidad de cambiar.(Birla & Johansson, 2015)

²³ Tabla utilizada para medir el rendimiento. Indicador utilizado como herramientas para comparación y evaluar el rendimiento.

Portabilidad	Características por cumplirse para que el software se transferible en distintos entornos. Adaptabilidad, instalabilidad, coexistencia y conformidad de portabilidad.
Compatibilidad	Es la capacidad de compartir el mismo entorno de hardware o software entre dos o más sistemas que necesitará compartirse información o llevar a cabo sus funciones requeridas.
Seguridad	La información debe estar segura y lo damos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos.

Tabla 6.- Características de la norma ISO/IEC 25010

Fuente: (ISO, 2011)

- **Indicadores**

Se utilizan para predecir la eficacia del comportamiento del producto de software durante la prueba o de funcionamiento. Se deben tener establecidas las condiciones necesarias, como de la configuración tanto del hardware como del software en un entorno de referencia.(Caro, Fuentes, & Soto, 2013)

NOMBRE DE MÉTRICA	TIEMPO DE RESPUESTA
Propósito de las métricas	Medir cual es tiempo estimado para completa una tarea específica.
Método de aplicación	<p>Evaluar la eficiencia del sistema operativo y las llamadas del sistema de aplicación.</p> <p>Estimar el tiempo de respuesta en base a esto.</p> <p>Los siguiente pueden ser medidos:</p> <ul style="list-style-type: none"> - Todo o partes de las especificaciones del diseño - Prueba de transacción completa o ruta - Prueba completa de módulos o partes del producto de software. - Producto de software completa durante la fase de prueba.
Medición, formula y cálculos de elemento de datos	$X = \text{tiempo (calculado o simulado)}$
Interpretación del valor medido	Cuanto más corto mejor
Tipo de escala métrica	
Tipo de medida	$X = \text{tiempo}$
Fuente de entrada de medición	<p>Sistema operativo conocido</p> <p>Tiempo estimado en llamadas al sistema</p>
ISO/IEC 12207 SLCP	Validación
Referencia	Revisión conjunta
Público objetivo	<p>Desarrolladores</p> <p>Solicitantes</p>

Tabla 7.- Métricas para evaluación en la característica Eficiencia de Desempeño

Fuente: (Caro et al., 2013)

1.8.2 Scrum como marco de trabajo

- **Scrum**

Metodología ágil de desarrollo para la gestión de un proceso. Se caracteriza por su preparación para equipos de seis a nueve personas, con acciones que van de iteraciones llamadas timeboxed. También llamados sprints que se tratan de procesos a realizarse y que se pueden revisar en menos de 30 días, generalmente dos semanas.(Maximini, 2015)

Scrum²⁴ se caracteriza por tener un equipo de trabajo auto organizado. Por lo que no existe un líder scrum quien decidirá que hacer o de qué forma realizar dicha tarea. Existen roles quienes están a cargo de que el proceso se realice dentro de este marco de desarrollo.(Bohmer, Hugger, & Lindemann, 2017) Esta el scrum master es quien entrena al equipo ayuda a los miembros del equipo a utilizar para desempeñar en un nivel más alto.

- **Scrum roles**

Product owner²⁵. – es una persona con visión autoridad; responsable de la comunicación y la visión del equipo. Muchas veces tiene que luchar con todas las dificultades de encontrar un equilibrio correcto de entre la necesidad de administrar dentro de la organización y responder a todas las preguntas del equipo.(Mundra, Misra, & Dhawale, 2013)

Scrum master²⁶. – Es un facilitador para el product owner y el team. Scrum Master no gestiona el equipo, elimina todo impedimento que exista para lograr que el equipo cumpla con los objetivos manera ágil. De esta forma se asegura que el tema siga siendo productivo y creativo. De igual forma trabajar, asesorar al product owner sobre como maximizar la ganancia para el equipo.

Team²⁷. – el equipo es totalmente autogestionado y responsable de la organización, con siete miembros completamente dedicados idealmente en una sala protegida de distracciones externas. Es un equipo multidisciplinario conformado por ingenieros

²⁴ Metodología ágil de trabajo multipropósito

²⁵ Dueño del producto

²⁶ Facilitador

²⁷ Equipo encargado de desarrollar en proyecto

de software, arquitectos, programadores, analistas expertos de control de calidad, testers y diseñadores de UI²⁸.

- **Eventos Scrum**

Estos eventos regulan y minimizan la necesidad de tener reuniones no definidas, todos los eventos son una caja de tiempo. Una vez que el sprint²⁹ inicia no se puede alargar o acorta. Los eventos restantes pueden finalizar siempre y cuando se logre el propósito del evento, asegurando que se invierta una cantidad adecuada de tiempo sin desperdiciar en el proceso.(Srivastava, Bhardwaj, & Saraswat, 2017)

- Sprint³⁰. - es un periodo de tiempo de un mes o menos durante el cual una historia de usuario se ha completado. Los sprint tienen duraciones constantes a lo largo del desarrollo, un nuevo sprint comienza inmediatamente después de haber finalizado uno anterior.(Karam, 2017b)
- Sprint planning³¹. - es toda la planificación del sprint. Este trabajo es elaborado por el trabajo colaborativo de todo el equipo. Tiene una duración de ocho horas para un sprint de un mes. el Scrum Master garantiza que el evento se lleve a cabo y que los asistentes entiendan el propósito.
- Daily scrum³². - es un evento de 15 minutos en el equipo de desarrollo se coincide a las actividades y se crea un plan para las siguientes 24 horas. Como se describe se desarrolla cada día. Esto sirve para planificar el avance del próximo trabajo del sprint. Se lleva a cabo a la misma hora y lugar todos los días para reducir la complejidad.
- Sprint review³³. - una revisión se lleva a cabo al final de cada sprint para revisar el incremento y adaptar el product backlog³⁴ si es necesario. Durante la revisión el equipo y las partes interesadas colaboran sobre lo que se hizo en el sprint. Es una reunión informal no una reunión como un sprint planning destinada a obtener comentarios y fomentar la colaboración. El scrum

²⁸ Interfaz de Usuario. **User Interface**

²⁹ Es un período de tiempo de un mes o menos durante se realiza un proceso

³⁰ Carrera, periodo de tiempo de trabajo

³¹ Planificación del trabajo

³² Reunión diaria

³³ Revisión del sprint

³⁴ Lista ordenada de todo lo necesario en el desarrollo de una historia de usuario

master garantiza que el evento se lleve a cabo y que los asistentes entiendan el propósito.(Karam, 2017a)

- Sprint retrospective³⁵. – es una oportunidad para que el equipo se inspeccione a sí mismo y cree un plan para que se difunda mejoras para el próximo sprint. Esta ocurre después de la revisión de sprint y antes de la planificación del siguiente sprint. Esta es una oportunidad para el equipo mejore y todos los miembros deben estar presentes.

- **Artefactos scrum**

Representa el valor al trabajo que provee de transparencia y oportunidades de inspección y adaptación.

- Product backlog³⁶. – es una lista ordenada de todo lo que necesita el producto lo que se sabe que necesita el producto. Aquí se almacena los requisitos para cualquier cambio que se realice le producto. Evolucionan a medida que el producto también lo hace, por lo que cambia constantemente para identificar lo que es apropiado para el producto.
- Sprint backlog³⁷. – es una previsión del equipo de desarrollo sobre que funcionalidad habrá en la próxima iteración. Identifica todo lo que el trabajo que el equipo identifica como necesario para el cumplir el objetivo del sprint. De igual forma se cataloga como plan con suficientes detalles para que los cambios en curso puedan entender el daily scrum. Es una imagen altamente visible y en tiempo real del trabajo que el equipo de desarrollo planea durante el sprint perteneciente únicamente al equipo de desarrollo.(Mundra et al., 2013)
- Increment³⁸. – es la suma de todos los elementos de la cartera de productos completados durante el sprint y el valor de los incrementales de los sprints anteriores. Al final el sprint debe estar “completado” lo que significa que debe estar en condiciones de uso. Se entiende por “completado” cuando todo el equipo necesita una evaluación del trabajo y el incremento del producto todos deben estar de acuerdo en que se finalizó para dar por hecho como “completado”.

³⁵ Retroalimentación junto al equipo

³⁶ Lista ordenada de todo lo necesario en el desarrollo de una historia de usuario

³⁷ Planificación de las tareas pendientes

³⁸ Todas las tareas que se dan por completados

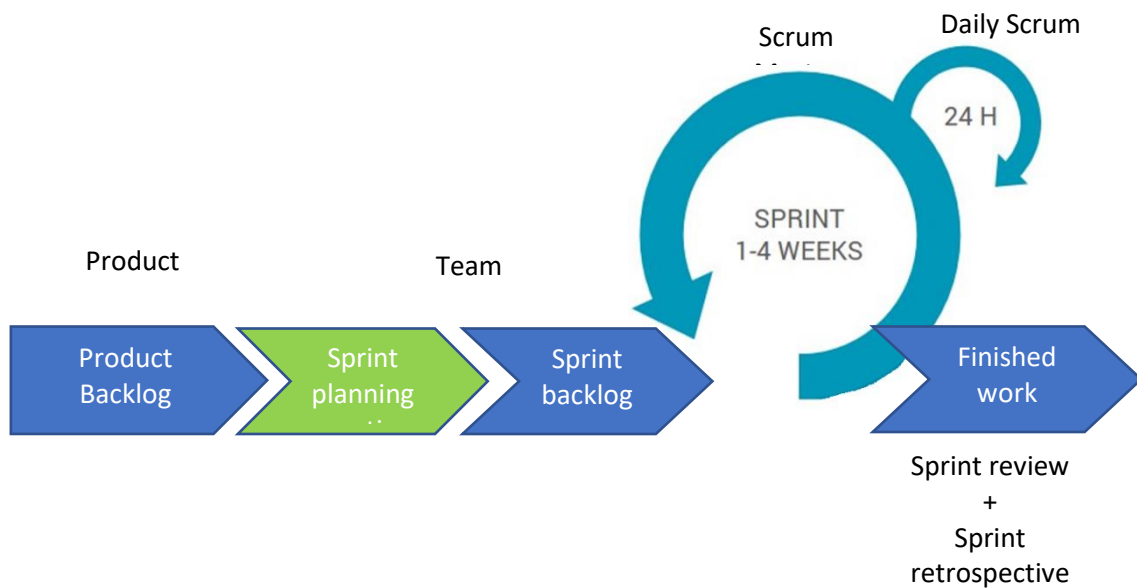


Figura 22.- Ciclo de vida de la metodología Scrum

Fuente: bit.ly/2DivJ4X

1.9 IONIC COMO FRAMEWORK DE DESARROLLO

Es un framework³⁹ de código abierto para el desarrollo de aplicaciones móviles híbridas con tecnologías como CSS⁴⁰, HTML5⁴¹ y Sass⁴². Creado en Dfrity Co. por Max Lynch, Ben Sperry y Adam Bradley. La versión original fue lanzada en el año 2013 y desarrollado sobre Angular JS y Apache Cordova. Las aplicaciones son desarrolladas en estas tecnologías web y para luego distribuirse a través de tiendas de aplicaciones como AppStore en iOS, Store de Windows y Play Store para Android.(Lynch, 2016)

Ionic utiliza complementos de Cordova para obtener acceso al hardware (cámara, GPS, linterna, etc) del teléfono móvil. Ionic permite crear e implementar aplicaciones mediante wrapping⁴³ con la herramienta de Cordova con una simple línea de comando 'ionic'.

Aplicación multiplataforma, creada a partir de una aplicación web simple, pero permite generar una aplicación nativa con acceso a todas las funcionalidades

³⁹ Entorno de software universal que proporciona funcionalidad como parte del propio software

⁴⁰ Hojas de estilo en cascada. **Cascade Style Sheets**

⁴¹ Lenguaje de marcas para la creación de páginas web

⁴² Lenguaje de para hojas de estilos en cascada

⁴³ Empaquetar todo el proyecto y mostrar como una aplicación móvil

específicas del teléfono. Estas aplicaciones multiplataforma no son puramente nativas y no están completamente basadas en web. Mediante una API⁴⁴ nativa del dispositivo la aplicación tiene acceso al teléfono.(Yusuf, 2016) Al combinar la parte nativa del teléfono con un elemento web nos da una solución de aplicación universal. Lo más relevante de esto y conveniente tanto para los propietarios de los negocios como para los desarrolladores es que una vez terminado el código podemos ejecutar en múltiples plataformas.

- **Ventajas**

Este framework se basa en tecnologías muy reconocidas lo que permite que la curva de aprendizaje sea mínima, el equipo de desarrollo tampoco tiene que conocer todas las por menores relacionadas con cada plataforma móvil.(Yusuf, 2016)

Al ser un marco de desarrollo de código abierto todos los programadores (Yusuf, 2016) pueden unirse a la comunidad de desarrolladores de Ionic, mediante un foro para compartir y obtener ayuda, en caso de cualquier problema se puede contar con la asistencia especial para las empresas.

- **Desventajas**

En cuanto al rendimiento una aplicación híbrida siempre va a estar en desventaja a diferencia de una netamente nativa, sin embargo, la brecha de rendimiento es casi notable para un usuario promedio. Pero si queremos realizar material gráfico como desarrollo para videojuegos o 3D la diferencia será muy notable.

⁴⁴ Protocolo de comunicación y herramientas para construcción software. **Application Programming Interface**

CAPÍTULO II

DESARROLLO

2.1. PROCESO DE COMPARTIR UN VIAJE EN AUTOMÓVIL

Proceso de viaje por autocompartido
Caso beneficiario

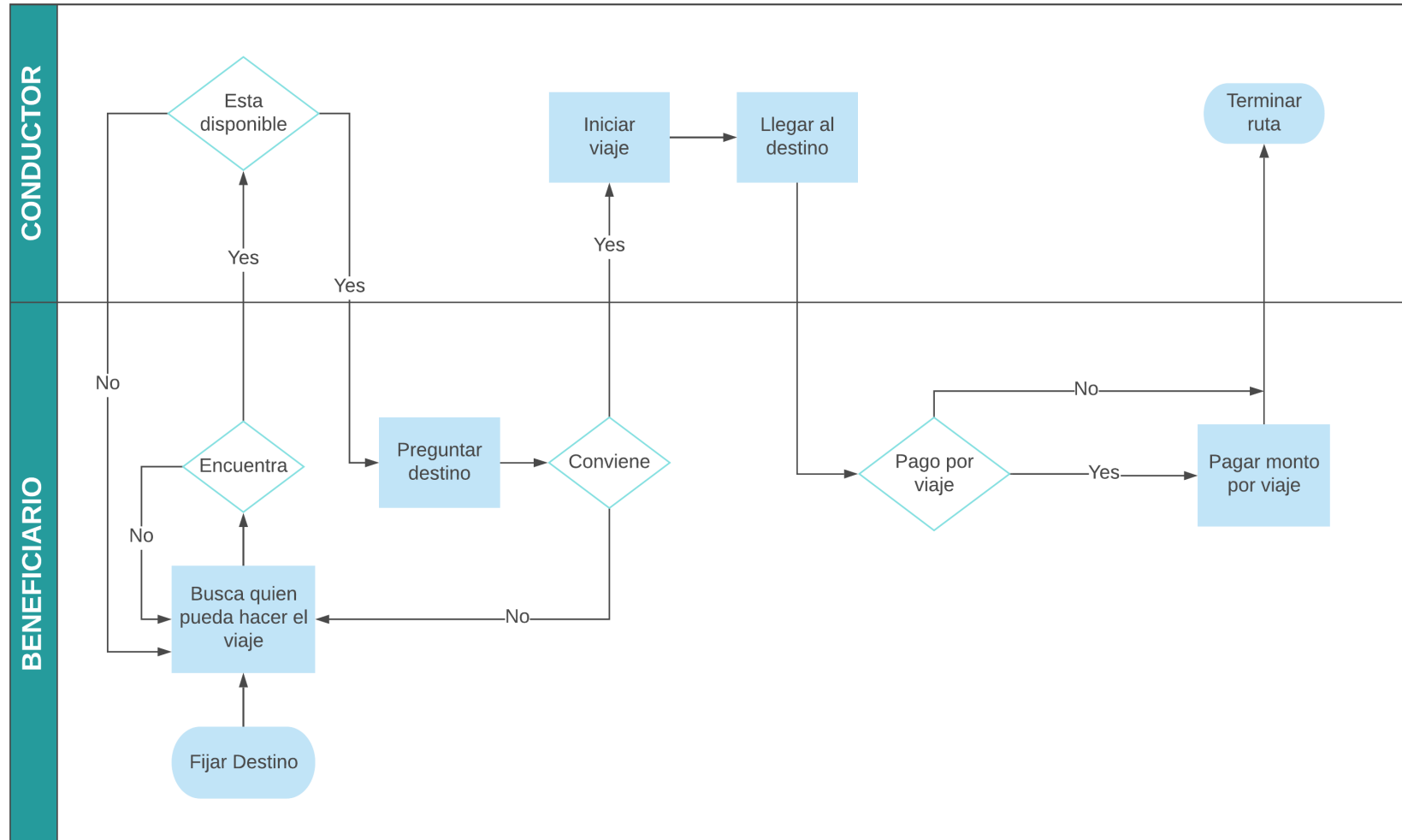


Figura 23.- Proceso de auto compartido – caso beneficiario

Proceso de viaje por autocompartido
Caso conductor

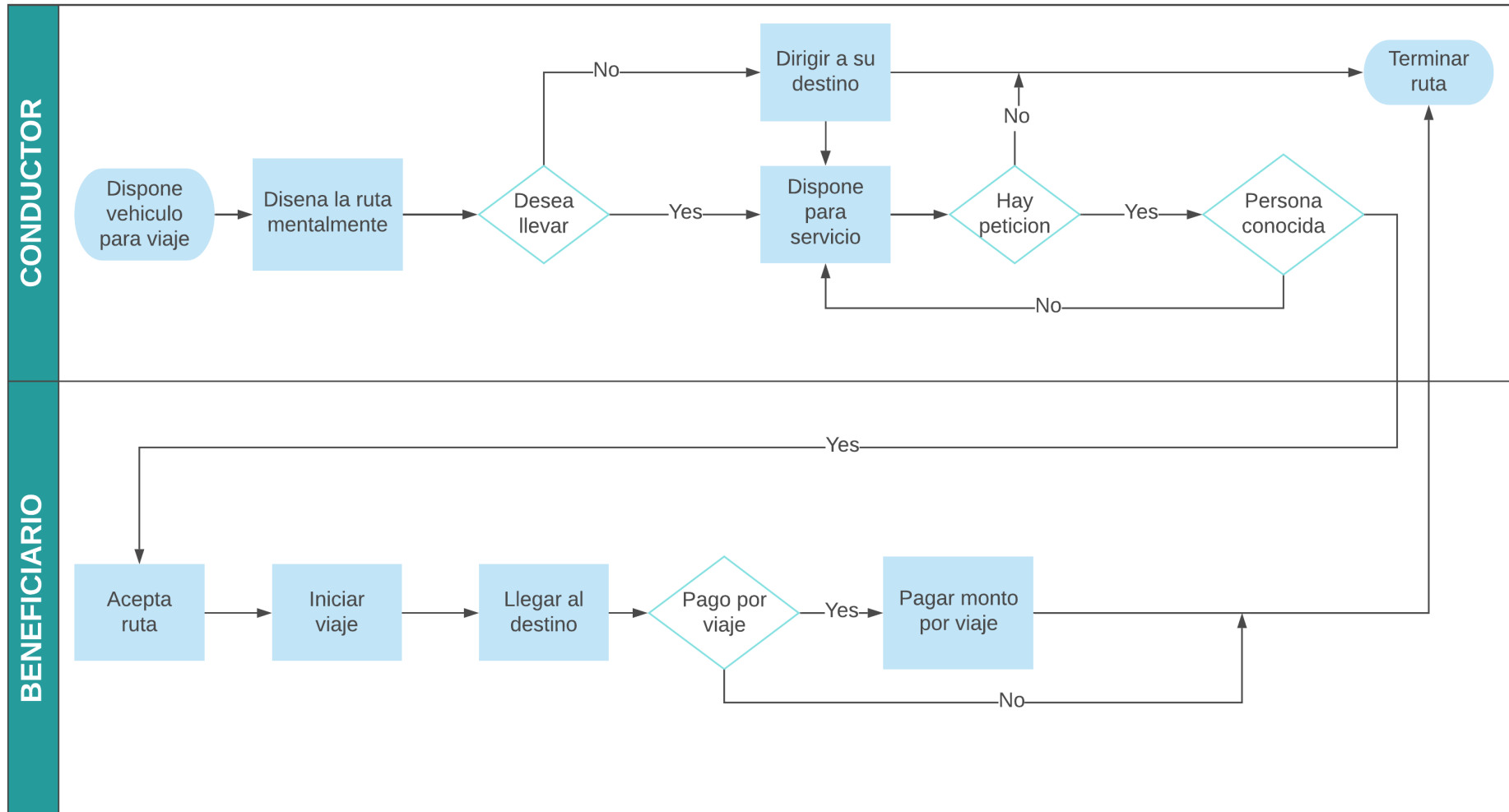


Figura 24.- Proceso de auto compartido – caso conductor

2.2. PLANEACIÓN DEL PROYECTO

Para el desarrollo de la aplicación móvil se tomó en cuenta el proceso de auto compartido - caso conductor, como se detalla en la *figura 26*. Este proceso permite al equipo de desarrollo codificar y evaluar la realidad del modelo de negocio a ser implementado, considerando tiempos y costos dentro del proyecto.

2.2.1. Definición de requerimientos

Las historias de usuario se han obtenido en base a requisitos necesarios para completar el módulo de creación y visualización de ruta.

Estimación	Valoración
0	Es la puntuación para estimar historias que ya están hechas y necesitan un refinamiento, que se las puede realizar en ese instante. 0 que son fáciles de realizar, donde el trabajo implique menos de 1 hora.
1/2	4 horas
1	8 horas
2	Entre 1 y 2 días
3	Entre 2 y 4 días
5	Entre 3 y 5 días
8	Entre 5 y 8 días
13	2 semanas
20	Entre 2 y 4 semanas
40	1 mes
100	Esta valoración se utiliza cuando la historia de usuario es muy grande y requiere de más tiempo para su desarrollo.
? o ∞	Este tipo de valoración se usa cuando no se tiene los conocimientos suficientes para resolver la historia de usuario.

Tabla 8.- Valoración para las historias de usuario

HISTORIA DE USUARIO

HU101

Usuario: SN

Nombre: Elaboración requerimientos

Prioridad: Alta

Riesgo: Alta

Estimación: 8

Descripción: Levantamiento de requisitos en reunión con los stakeholders del producto en el Departamento de Tecnologías. Necesito una base de datos y arquitectura bajo las siguientes normas, para que sea acorde a la arquitectura vigente de la universidad.

Pruebas de Aceptación:

- Levantamientos de requisitos firmados por el Product owner.
 - Las tablas de la base de datos deben tener metadatos compatibles y legibles y una norma de escritura (MOV_) que sea utilizado en todas las tablas de movilidad.
 - El despliegue de la aplicación se debe realizar para las dos plataformas tanto en Android como en iOS.
-

Tabla 9.- Historia de usuario H101

HISTORIA DE USUARIO

HU102

Usuario: Conductor/Pasajero

Nombre: Diseño y prototipado

Prioridad: Alta

Riesgo: Alta

Estimación: 13

Descripción: Necesito visualizar una vista preliminar de la aplicación móvil para disponer si se necesita cambios o no.

Pruebas de Aceptación:

- Las guías y estilos de UI (colorimetría, tipografía, iconografía, navegabilidad) deben estar relacionadas con la identidad de la universidad o de la institución.

Tabla 10.- Historia de usuario H102

HISTORIA DE USUARIO

HU103

Usuario: Conductor

Nombre: Gestión de creación de rutas

Prioridad: Alta

Riesgo: Alta

Estimación: 13

Descripción: Como conductor necesito crear mi ruta de viaje, para poner mi viaje a disponibilidad a otras personas. Toda la transacción y gestión de datos deberá estar asegurada.

Pruebas de Aceptación:

- Antes de elegir el destino deseado por el conductor, deberá elegir a la universidad como origen y viceversa.
- El origen, destino ruta y punto de encuentro se visualizará en Google Maps

Tabla 11.- Historia de usuario H103

HISTORIA DE USUARIO

H104

Usuario: Conductor/Pasajero

Nombre: Gestión de Mapas

Prioridad: Alta

Riesgo: Alta

Estimación: 8

Descripción: Como pasajero necesito visualizar todas las reservas sin ningún parámetro, para seleccionar la ruta que más me convenga se visualizarán los

siguientes datos: origen, destino y hora de salida, fotografía del conductor, nombres y apellidos del conductor, número de asientos disponibles.

Pruebas de Aceptación:

- El origen y destino deben estar en una letra grande.

Tabla 12.- Historia de usuario H104

HISTORIA DE USUARIO

H105

Usuario: Conductor

Nombre: Gestión de imágenes

Prioridad: Alta

Riesgo: Alta

Estimación: 13

Descripción: Como pasajero necesito seleccionar un viaje disponible, para iniciar el viaje.

Pruebas de Aceptación:

- Cada uno de los diferentes viajes publicados deberán estar ordenados en orden cronológico, el ultimo se posicionará en la parte de arriba y así sucesivamente.

Tabla 13.- Historia de usuario H105

HISTORIA DE USUARIO

H106

Usuario: Pasajero

Nombre: Gestión de reserva de rutas

Prioridad: Alta

Riesgo: Alta

Estimación: 20

Descripción: Como conductor necesito aceptar o rechazar solicitud de viaje, para tener la elección de llevar o no a una determinada persona.

Pruebas de Aceptación:

- Debe haber un botón de aceptar y rechazar en cada uno de los solicitantes en un viaje determinado.
-

Tabla 14.- Historia de usuario H106

HISTORIA DE USUARIO

H107

Usuario: Conductor/Pasajero

Nombre: Visualización de rutas

Prioridad: Alta

Riesgo: Alta

Estimación: 8

Descripción: Como pasajero necesito visualizar la ruta y el punto de encuentro en un mapa además de la información del conductor, para añadir seguridad a la plataforma y tener un registro de todo el recorrido.

Pruebas de Aceptación:

- Los mapas se visualizarán con Google Maps por preferencia del cliente.
 - La información del conductor deberá ser la fotografía, nombres, apellidos y el automóvil en el que va a realizar el viaje.
 - Además, la información académica del conductor como: ocupación, facultad, carrera y el número de teléfono móvil.
 - La información del automóvil será la imagen del automóvil, placas, marca y modelo.
-

Tabla 15.- Historia de usuario H107

HISTORIA DE USUARIO		
H108	Usuario: Pasajero	
Nombre: Gestión de usuarios		
Prioridad: Alta	Riesgo: Alta	Estimación: 13
<p>Descripción: Como pasajero necesito visualizar la ruta y el punto de encuentro en un mapa además de la información del conductor, para añadir seguridad a la plataforma y tener un registro de todo el recorrido.</p>		
<p>Pruebas de Aceptación:</p> <ul style="list-style-type: none"> • Los mapas se visualizarán con Google Maps por preferencia del cliente. • La información del conductor deberá ser la fotografía, nombres, apellidos y el automóvil en el que va a realizar el viaje. • Además, la información académica del conductor como: ocupación, facultad, carrera y el número de teléfono móvil. • La información del automóvil será la imagen del automóvil, placas, marca y modelo. 		

Tabla 16.- Historia de usuario H108

2.2.2. Definición del product backlog

De acuerdo con cada una de las historias de usuarios que se han levantado en los requerimientos esto es una vista general del product backlog. Ver tabla 15.

Prioridad	Id	Historia	Estimación
Alta	H101	Elaboración requerimientos	8
Alta	H102	Diseño y prototipado	13
Alta	H103	Gestión de creación de rutas	20

Alta	H104	Gestión de Mapas	8
Alta	H105	Gestión de imágenes	13
Media	H106	Gestión de reserva de rutas	20
Alta	H107	Visualización de rutas	3
Alta	H108	Gestión de usuarios	13

Tabla 17.- Product Backlog

2.2.3. Definición de roles del proyecto

Se han declarado los siguientes roles conforme a la metodología Scrum, para el desarrollo de la aplicación móvil y seguimiento.

Persona	Descripción	Rol
Ing. Juan Carlos García	Director del Departamento de Desarrollo Tecnológico e Informático (DDTI) de la UTN	Propietario del Producto (Product Owner)
Ing. Alex Guevara	Director de la propuesta de investigación y docente de la Carrera de Sistemas Computacionales de la UTN	Jefe de Proyecto (Scrum Master)
Bertil Tandayamo	Tesista y desarrollador de software	Equipo de desarrollo (Development Team)
Ing. Lenin Chávez	Analista de Sistemas de la DDTI UTN	Equipo de análisis del sistema (Development Team)

Ing. Irving López	Analista de Sistemas de la DDTI UTN	Equipo de análisis y testing del sistema (Development Team)
-------------------	--	---

Tabla 18.- Roles del proyecto

El desarrollo de la aplicación móvil se realizó de forma incremental mediante la metodología Scrum. Esta metodología trabaja con Sprints. Son lapsos de tiempo donde se deben realizar determinadas tareas que van de acuerdo con los requerimientos del desarrollo de software obtenidas de las historias de usuario.

2.2.4. Planificación de Sprints

Se mostrará el desarrollo en todas las iteraciones hasta lograr obtener el producto completo. El resultado de la ejecución de los sprint se muestra a continuación. El desarrollo de cada Sprint tiene una duración de cuatro semanas y de 100 horas cada uno.

Sprint 1

H101 Elaboración requerimientos

H102 Diseño y prototipado

Historia de Usuario	Actividad	Tarea	No. Horas
		Levantamiento de información de requisitos funcionales	20
H101	Elaboración requerimientos	Instalación de herramientas de la arquitectura designada	10

		Planificación de requisitos no funcionales	20
		UI de conductor	25
H102	Diseño y prototipado	UI de pasajero	25
		TOTAL	100

Tabla 19.- Sprint 1

Sprint 2

H103 Gestión de creación de rutas

Historia de Usuario	Actividad	Tarea	No. Horas
		Creación de rutas	20
		Eliminación de rutas	20
H103	Gestión de creación de rutas	Actualización de rutas	20
		Reporte de rutas	20
		Función de aceptar o rechazar rutas	20
		TOTAL	100

Tabla 20.- Sprint 2

Sprint 3

H104 Gestión de Mapas

H105 Gestión de imágenes

Historia de Usuario	Actividad	Tarea	No. Horas
H104	Gestión de Mapas	Creación de mapa para punto de encuentro	25
		Función para visualizar el trayecto	25
H105	Gestión de imágenes	Función para añadir imagen mediante cámara	8
		Guardar imágenes de automóvil	8
		Eliminar imágenes de automóvil	8
		Actualizar imágenes de automóvil	8
		Lectura imágenes de automóvil	8
		Función para añadir imagen mediante galería	10
TOTAL			100

Tabla 21.- Sprint 3

Sprint 4

H106 Gestión de reserva de rutas

Historia de Usuario	Actividad	Tarea	No. Horas
H106	Gestión de reserva de rutas	Creación de rutas	25
		Eliminación de rutas	25
		Actualización de rutas	25
		Reporte de rutas	25
		TOTAL	100

Tabla 22.- Sprint 4

Sprint 5

H107 Visualización de rutas

H108 Gestión de usuario

Historia de Usuario	Actividad	Tarea	No. Horas
H107	Visualización de rutas	Elaborar un listado de las rutas creadas por el conductor	30
		Elaborar un listado de las rutas disponibles	30
H108	Gestión de usuarios	Acoplamiento de módulo de login	40

Tabla 23.- Sprint 5

2.2.5. Desarrollo de los Sprints

Sprint 1

H101 Elaboración requerimientos

Se realiza la instalación de las herramientas que se da a conocer en la siguiente arquitectura Fig. 27, de igual forma se diseñó el modelo entidad relación de la base de datos, Fig. 28.

Arquitectura del sistema

Desde el año 2007 el Departamento de Tecnologías de la universidad ha implementado una arquitectura tecnológica basada en Oracle⁴⁵. Desde la base de datos pasando por el middleware⁴⁶ y hasta el entorno de desarrollo de software. Ionic fue seleccionado como framework de desarrollo para proyectos de aplicaciones móviles, por lo tanto, siguiendo la arquitectura se decidió continuar con la misma herramienta para el desarrollo de este proyecto de tesis; en un futuro permitirá a que la mantenibilidad y despliegue sean más fáciles de implementar. Revisar anexo 1 para más información de la arquitectura.

⁴⁵ Empresa multinacional de servicios tecnológicos

⁴⁶ Software que proporciona servicios de aplicaciones de software para comunicación de hardware y software

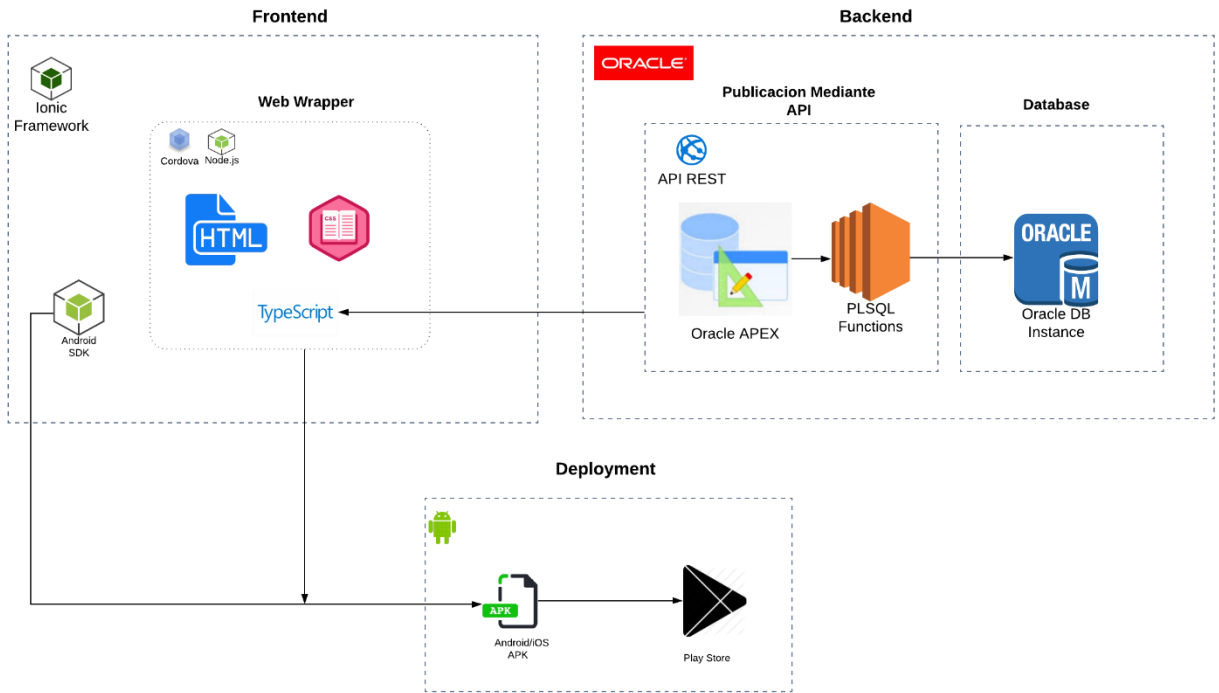


Figura 25.- Arquitectura del sistema

Diagrama entidad relación de la base de datos

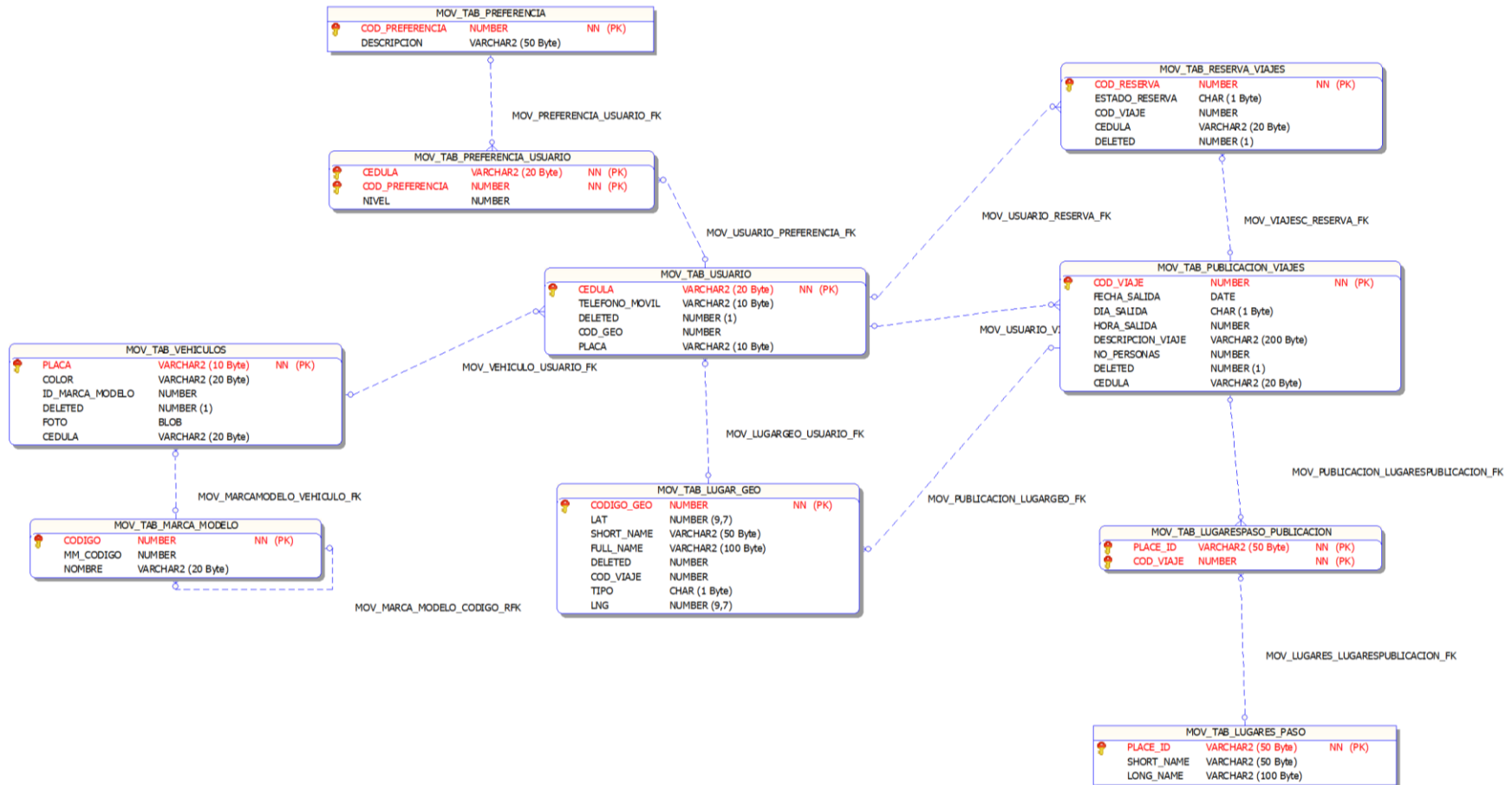


Figura 26.- Diagrama entidad relación base de datos



Kubix UTN

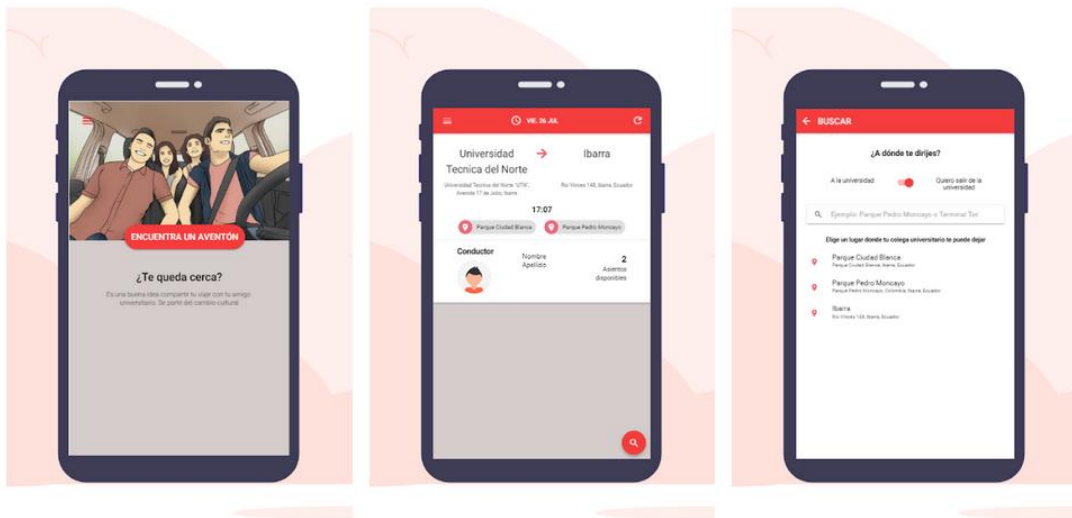
Lenin Xavier Chávez G. Educación

Para todos

- Esta aplicación está en desarrollo y puede ser inestable.
- Esta aplicación es compatible con tu dispositivo.

Añadir a la lista de deseos

Instalar



Kubix es una opción de movilidad mediante vehículo compartido hecho a la medida para el uso de la comunidad universitaria de la Universidad Técnica del Norte.

Figura 27.- Publicación de aplicación en Play Store

La aplicación se desplego únicamente para dispositivos Android. El despliegue para dispositivo iOS no se pudo concretar por dos razones:

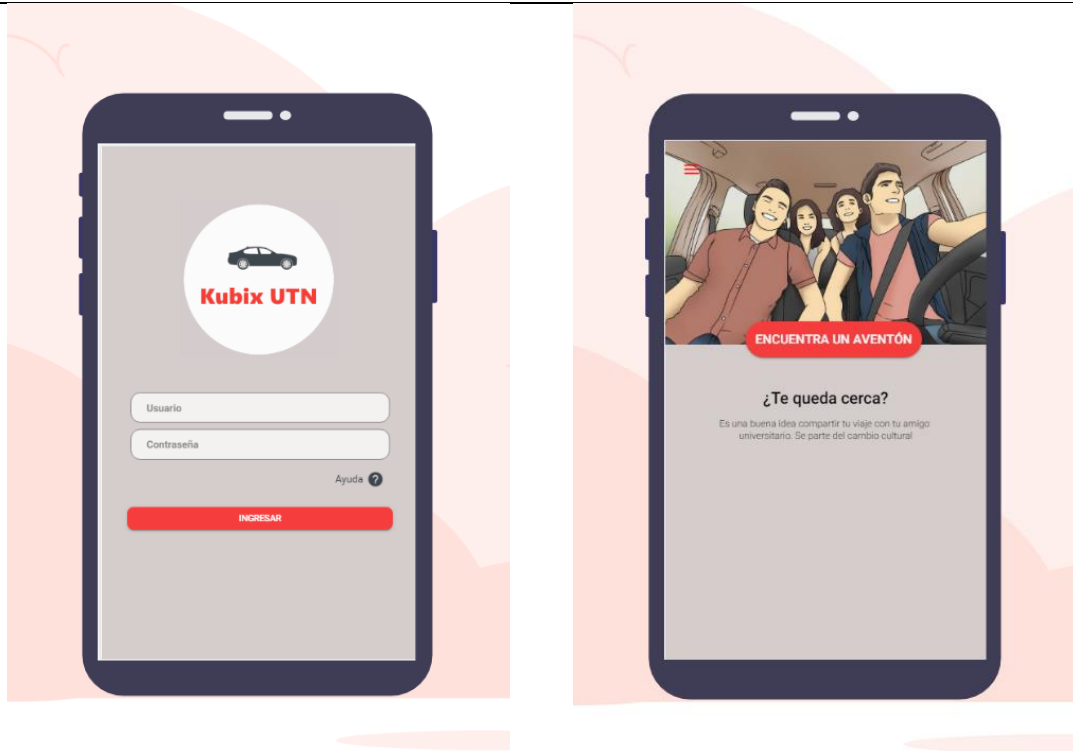
- Razón equipamiento (hardware). - para hacer desarrollo y posteriormente pruebas era necesario contar con tecnología netamente de Apple tanto en smartphone como en computadora con sistema operativo MacOS.
- Razón desarrollo. - era necesario el desarrollo desde una forma inicial para este sistema operativo ya que la interfaz y algunos plugin no coinciden tanto como para iOS como para Android.

Dicho esto, el tema de arquitectura y utilización de recursos del API están preparados para utilizar en el desarrollo netamente para este sistema operativo.

H102 Diseño y prototipado

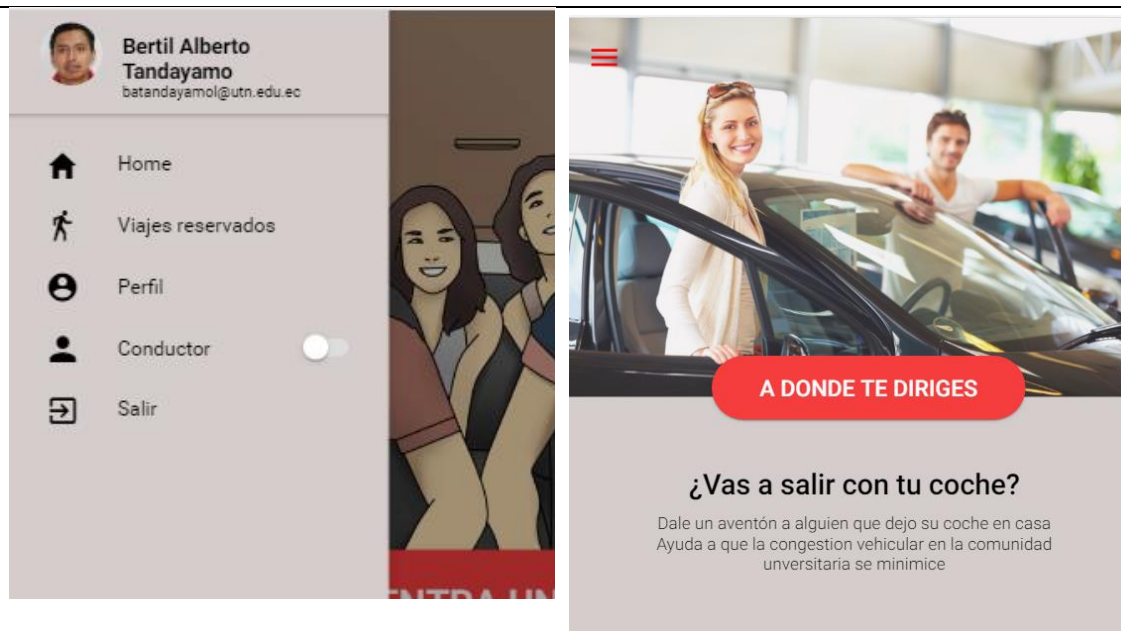
Para la fase de diseño y prototipado de la aplicación móvil se consideró trabajar con la herramienta de Adobe XD por que en esta historia de usuario se especifica la utilización de la identidad institucional tales como: colorimetría, tipografía, iconografía, navegabilidad.

Diseño de aplicación móvil



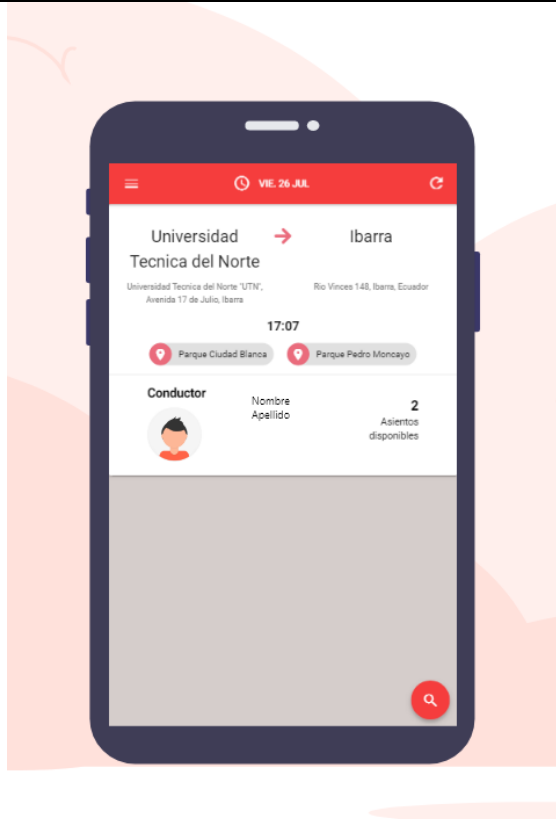
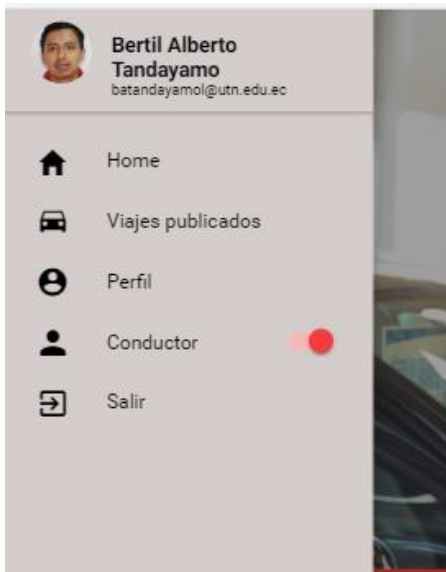
Inicio de sesión para la aplicación

Vista del home en modo pasajero. Esta vista será por defecto al momento de iniciar la aplicación.



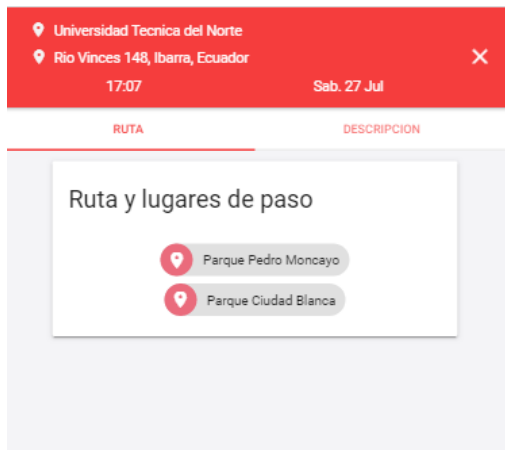
Menú lateral modo pasajero. Desde aquí puede cambiarse a modo conductor.

Vista home modo conductor



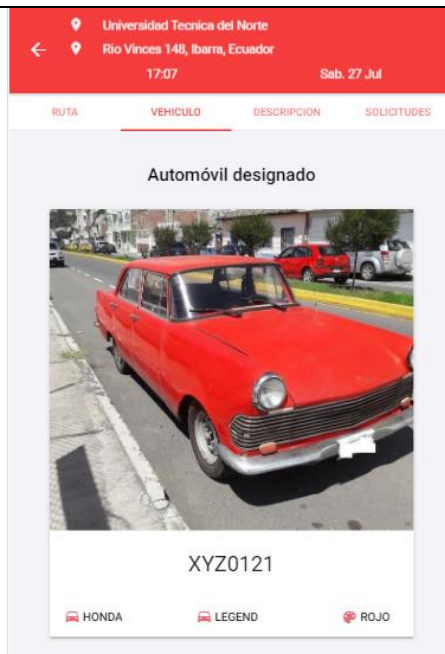
Menú lateral modo conductor. Se observa que esta activado el modo conductor.

Viajes disponibles en un día específico. Vista pasajero.



Vista a detalle del viaje seleccionado

Vista de viajes publicados. Vista conductor



Vista detallada de un viaje publicado

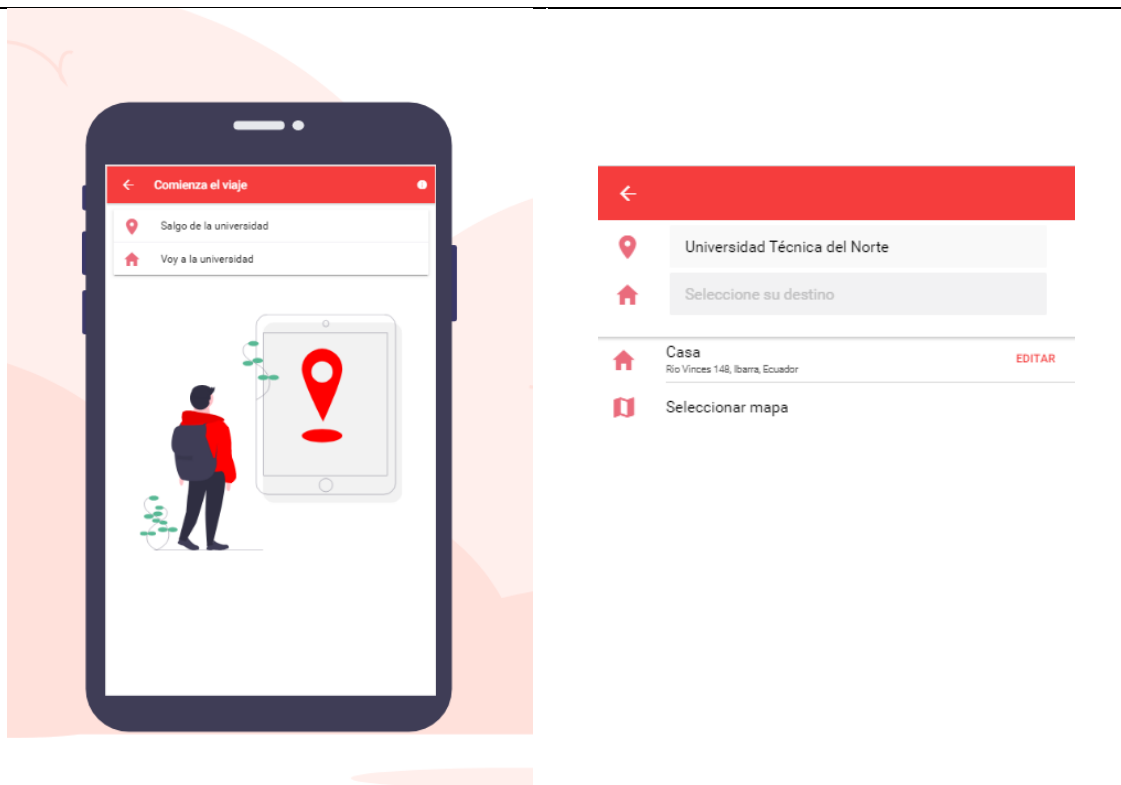
Tabla 24.- Diseño y prototipado de aplicación

Sprint 2

H103 Gestión de creación de rutas

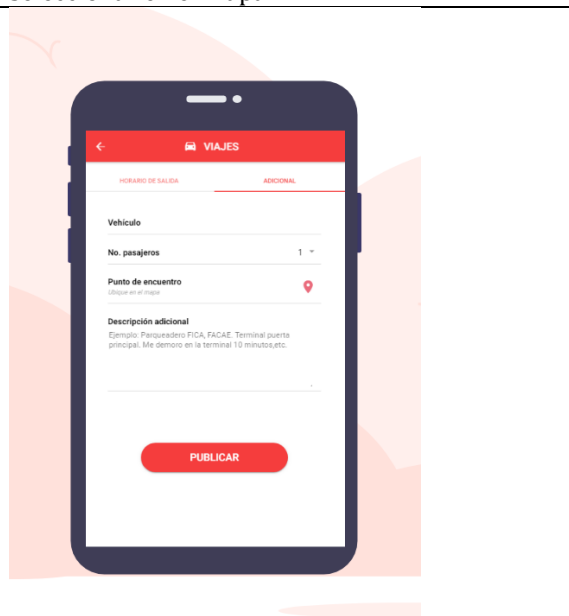
El usuario conductor realizará la gestión de rutas este módulo es el modelo de negocio para el servicio brindado a la comunidad universitaria.

Gestión de creación de rutas



El conductor selecciona como origen o destino la universidad

Elección de destino un lugar guardado con reciente, la casa o un lugar distinto que se puede seleccionar en el mapa.



Elección de origen ya sea la ubicación actual, la casa o un lugar distinto que se puede seleccionar en el mapa.

Detalles adicionales para la creación de la ruta. Y se publica la ruta

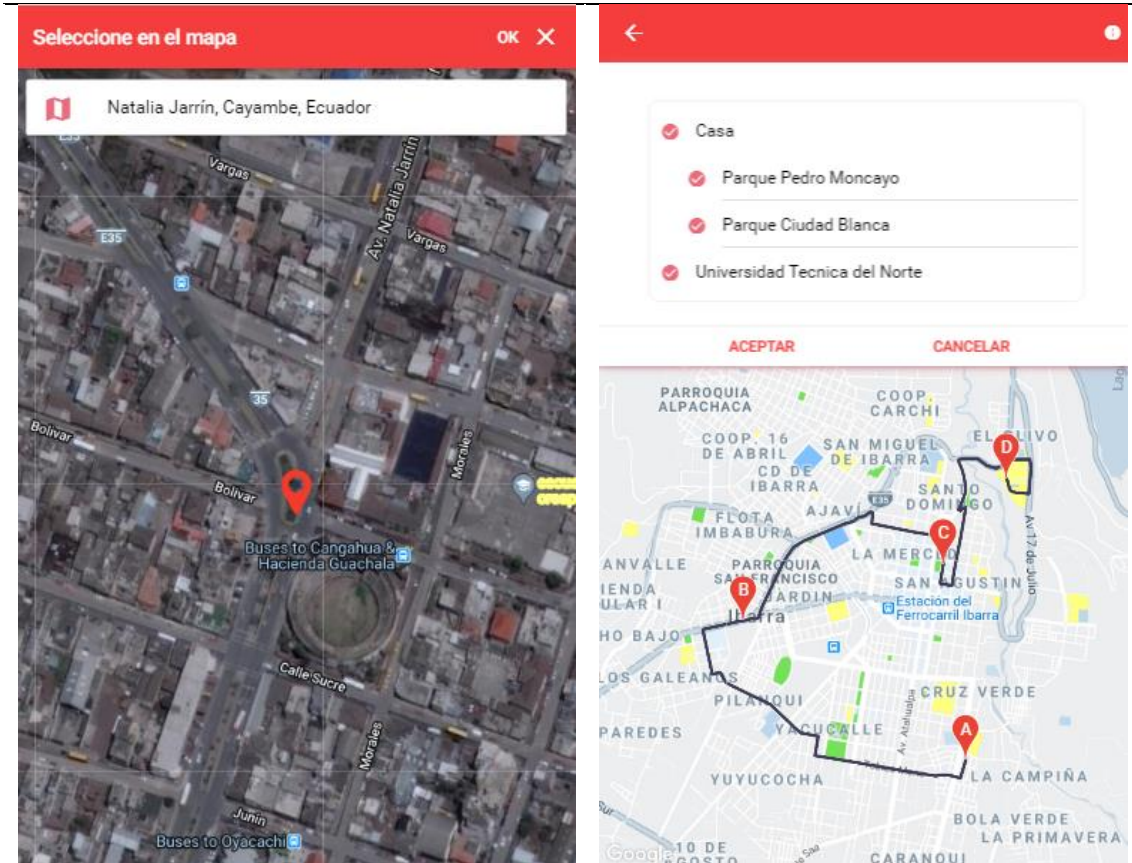
Tabla 25.- Gestión de creación de rutas

Sprint 3

H104 Gestión de Mapas

Para el Desarrollo de gestión de mapas se utilizó el API de Google Maps que ofrece en su capa gratuita. Específicamente se utilizó el Google API JavaScript.

Gestión de mapas



Selección de punto fijo como por ejemplo un punto de encuentro, un origen o un destino. Con muestra de dirección exacta. Esta opción está disponible solo para conductores para la creación de la ruta

Ruta creada por el conductor. Esta ruta se muestra en el mapa de acuerdo con el origen y destino que el conductor haya elegido, teniendo en cuenta siempre que el origen o el destino siempre será el campus universitario (UTN).

Tabla 26.- Gestión de mapas

Gestión de imágenes



	
<p>Visualización de imágenes de automóvil</p>	<p>Actualizar o ingresar fotografía del automóvil mediante cámara o galería</p>

Tabla 27.- Gestión de imágenes

Sprint 4

H106 Gestión de reserva de rutas

Gestión de reserva de rutas

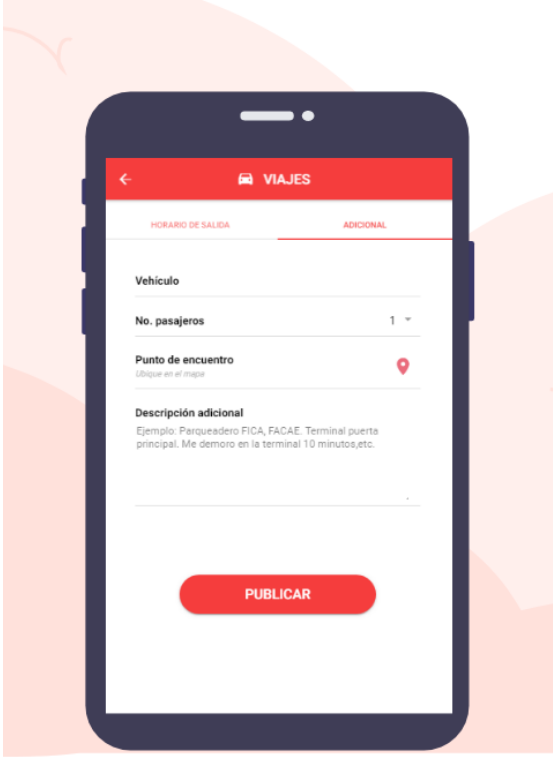

	
<p>Creación y publicación de rutas, usuario conductor.</p>	<p>Reporte y eliminación de rutas publicadas</p>

Tabla 28.- Gestión de reserva de rutas

Sprint 5

H107 Visualización de rutas

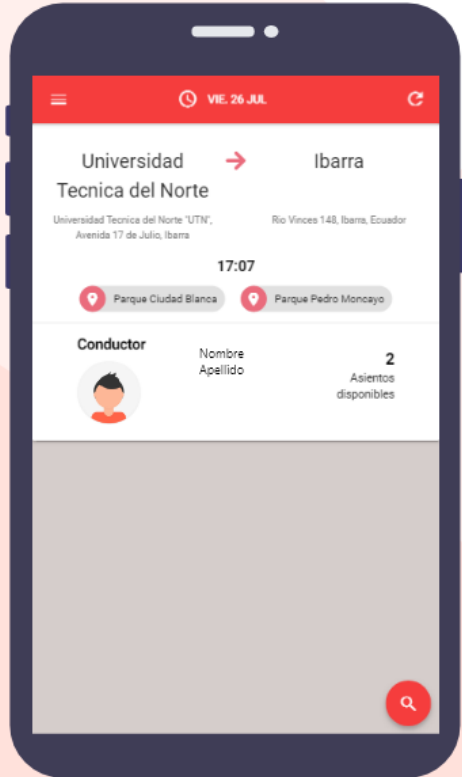
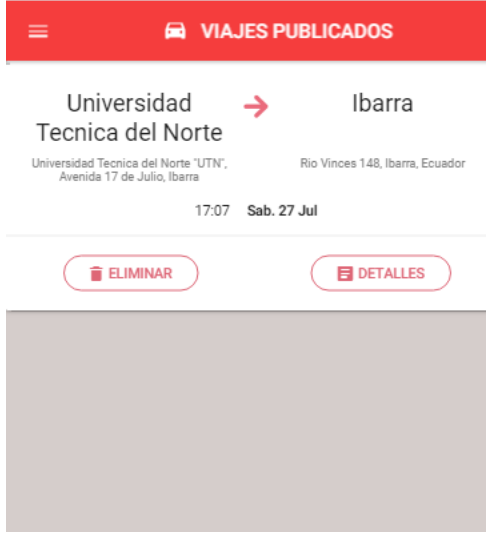
Visualización de rutas	
	
Visualización de las rutas que están disponibles para el día actual.	Rutas publicadas por el conductor

Tabla 29.- Visualización de rutas

Gestión de usuario



Gestión y acoplamiento del módulo de login con el módulo ingreso de la universidad

Gestión de sesiones en un dispositivo móvil para el ingreso y salida de la aplicación.

Tabla 30.- Gestión de usuario

CAPÍTULO III

VALIDACIÓN DE RESULTADOS

Para validar el presente proyecto se estableció evaluar la característica de eficiencia de desempeño de la norma ISO IEC 25010 tomando en cuenta las tres subcaracterísticas (comportamiento temporal, utilización de recursos y capacidad de procesamiento) se aplicó una evaluación de stress con uso y aplicación de Apache JMeter⁴⁷ como herramienta de evaluación, con el objetivo de comprobar que la aplicación móvil y los servicios alojados en el servidor Web y de aplicaciones tengan un alto performance en cuanto a la cantidad de peticiones y respuesta del servicio desplegado, simulando un ambiente de producción de altas prestaciones. Para ello, se realizó una muestra calculada de un total de 295.128 peticiones de tipo http⁴⁸ evaluando la gestión de rutas de la aplicación en ejecución y la gestión de mapas, divididos en cuatro diferentes muestras, cada una con tres ejecuciones de las pruebas controladas dentro de la aplicación. Mediante la herramienta de análisis estadístico SPSS⁴⁹ se realizó un análisis de distribución de datos y normalización de estos a través del test estadístico de Shapiro-Wilk⁵⁰ que verificó si el comportamiento de los datos obtenidos en la evaluación de stress, determinando si la distribución de los datos es normal.

3.1 PRUEBAS

Para aplicar la herramienta óptima que se adapte a las subcaracterísticas a ser ejecutadas dentro de la evaluación de stress se analizaron varias alternativas (Apache JMeter, Apache WorkBench, BlazeMeter, Neotys y Targus).

Después de un análisis de cada herramienta con sus opciones, se definió por su naturaleza y robustez trabajar con Apache JMeter como herramienta de evaluación, ya que permite medir en gran parte el performance del despliegue de una aplicación móvil en producción; además por sus características favorables como (interfaz gráfica amigable, soporte para diferentes extensiones de archivos para importar y

⁴⁷ Es una aplicación de escritorio de Apache utilizado para teste de carga y para analizar y medir el desempeño de un software.

⁴⁸ Del inglés Hypertext Transfer Protocol o Protocolo de Transferencia de Hipertexto HTTP.

⁴⁹ SPSS del inglés Statistical Package for the Social Sciences o Paquete Estadísticos para las Ciencias Sociales. Es un software que permite realizar cálculos estadísticos.

⁵⁰ Test de normalidad para frecuencias estadísticas, creado por Samuel Sanford Shapiro y Martin Wilk

exportar datos), y como valor agregado por ser una herramienta de código y uso libre.

Para la elaboración de la muestra de mínimos y máximos, se tomó en cuenta el número de automóviles que constan como activos de acuerdo con el sistema de control de ingreso de tarjetas RFID al campus, información generada por el Dirección de Desarrollo Tecnológico e Informático de la UTN.

Se detalla a continuación los siguientes datos del parque vehicular registrados con fuente de la DDTI UTN:

Descripción	Cantidad	Número de viajes mínimos	Número de viajes máximos	Número de pasajeros mínimos	Número de pasajeros máximo
Automóviles	529	2	4	2	5
Motocicletas	99	2	4	2	2

Tabla 31.- Posibles combinaciones de conjuntos de datos

El número de viajes mínimos se especificó teniendo en cuenta el ingreso en la mañana y la salida al final de la jornada. En caso del número máximo se tomó en cuenta la entrada y salida en la hora de almuerzo, este último caso es opcional, no necesariamente van a ser los 4 viajes.

En los automóviles para el caso de número de pasajeros mínimo se consideró (dos personas) un pasajero y adicional el conductor, y número máximo de pasajeros (cinco personas) cuatro pasajeros y adicional el conductor.

En las motocicletas se especificó como número mínimo y máximo de pasajeros (2 personas) un pasajero y adicional el conductor.

Una vez realizado las posibles combinaciones para cada uno de ellos se obtuvo 16 posibles combinaciones.

Se realizó tres iteraciones o sets de tres (S_n), ver tabla 33, y al final se muestra en la tabla 32 la combinación con el número total de procesos que se estableció parametrizar para la ejecución de la prueba. Con una muestra de un total de 295.128

divididos en cuatro conjuntos e iteraciones de tres en cada ejecución. Ver tabla 31 para identificar a las variables para la obtención de la cantidad de procesos:

Cp= Cantidad de procesos

Cv= Cantidad vehículo (tanto para automóviles como para motos)

Vmin = Número mínimo de viajes

Vmax = Número máximo de viajes

Pmin = Número mínimo de pasajeros

Pmax = Número máximo de pasajeros

$$Cp = Cv * Vmin * Vmax * Pmin * Pmax$$

Ecuación 1.- Fórmula para obtener la cantidad de procesos

Conjunto	No. proceso	Cant. de procesos (Cp)	Iteraciones	Total de procesos
C1	1	2512	3	7536
	2	2908	3	8724
	3	3570	3	10710
	4	3966	3	11898
C2	5	4628	3	13884
	6	4628	3	13884
	7	5024	3	15072
	8	5024	3	15072
C3	9	5686	3	17058
	10	6082	3	18246
	11	6744	3	20232
	12	7140	3	21420
C4	13	8860	3	26580
	14	9256	3	27768
	15	10976	3	32928
	16	11372	3	34116
TOTAL				295128

Tabla 32.- Iteraciones para pruebas

Una vez que se obtuvieron las combinaciones posibles se realizó las pruebas correspondientes a la evaluación de stress de la ejecución de la aplicación móvil en producción, ver anexo 3. Como resultado de la ejecución Apache JMeter se obtiene varios indicadores de medición entre los cuales se seleccionó el campo success (tareas ejecutadas), esta variable a su vez tiene dos estados *true* y *false* dando a conocer las transacciones exitosas (*true*) y no exitosas (*false*).

La variable da a conocer la efectividad de la tarea o proceso realizado, de esta forma se logró comprobar la eficiencia y el rendimiento de la aplicación.

Se elaboró una matriz para la ubicación de todos los valores tanto de *true* como de *false* y al final se obtuvo una media, Ver ecuación 2 de los tres sets realizados para esa cantidad de procesos.

$$\bar{xR} = \frac{Sn_1 + Sn_2 + \dots + Sn_n}{\# \text{ Muestra}}$$

Ecuación 2.- Definición de ecuación de medias

$$\bar{xR} = \frac{S1 + S2 + S3}{\text{Total de S}}$$

Ecuación 3.- Calculo de medias

Conjunto 1	S1		S2		S3		x̄R	
	true	false	true	false	true	false	true	false
2512	82896	5024	82418	5502	82323	5597	74328	4558
	(94.29%)	(5.71%)	(93.74%)	(6.26%)	(93.63%)	(6.37%)	(93.89%)	(6.11%)
2908	93552	8228	95959	5821	95964	5816	95158	6622
	(91.92%)	(8.08%)	(94.28%)	(5.72%)	(94.29%)	(5.71%)	(93.49%)	(6.51%)
3570	117801	7149	117808	7142	117802	7148	117804	7146
	(94.28%)	(5.72%)	(94.28%)	(5.72%)	(94.28%)	(5.72%)	(94.28%)	(5.72%)
3966	130860	7950	130731	8079	130856	7954	130816	7994
	(94.27%)	(5.73%)	(94.18%)	(5.82%)	(94.27%)	(5.73%)	(94.24%)	(5.76%)
x̄R							104527	6580
							(93.98%)	(6.03%)

Tabla 33.- Conjunto 1. Número de estados (*true/false*) en los procesos

Conjunto 2	S1		S2		S3		$\bar{x}R$		
	true	false	true	false	true	false	true	false	
4628	152603	9377	152653	9327	152607	9373	149468	12497	
	(88.40%)	(11.60%)	(94.24%)	(5.76%)	(94.21%)	(5.79%)	(92.28%)	(7.72%)	
4628	143145	18790	152653	9327	152607	9373	149468	12497	
	(88.40%)	(11.60%)	(94.24%)	(5.76%)	(94.21%)	(5.79%)	(92.28%)	(7.72%)	
5024	165792	10048	165774	10066	165668	10172	165745	10095	
	(94.29%)	(5.71%)	(94.28%)	(5.72%)	(94.22%)	(5.78%)	(94.26%)	(5.74%)	
5024	165792	10048	165774	10066	165668	10172	165745	10095	
	(94.29%)	(5.71%)	(94.28%)	(5.72%)	(94.22%)	(5.78%)	(94.26%)	(5.74%)	
							$\bar{x}R$	157607	11296
								(93.27%)	(6.73%)

Tabla 34.- Conjunto 2. Número de estados (true/false) en los procesos

Conjunto 3	S1		S2		S3		$\bar{x}R$		
	true	false	true	false	true	false	true	false	
5686	187457	11523	187638	11372	186602	11276	187232	11390	
	(94.21%)	(5.79%)	(94.29%)	(5.71%)	(94.30%)	(5.70%)	(94.27%)	(5.73%)	
6082	200733	12169	199768	13102	196036	12096	198846	12456	
	(94.28%)	(5.72%)	(93.85%)	(6.15%)	(94.19%)	(5.81%)	(94.11%)	(5.89%)	
6744	220515	14264	220963	13561	220639	13813	220706	13879	
	(93.92%)	(6.08%)	(94.22%)	(5.78%)	(94.11%)	(5.89%)	(94.08%)	(5.92%)	
7140	235606	14294	234326	14559	234866	14327	234933	14393	
	(94.28%)	(5.72%)	(94.15%)	(5.85%)	(94.25%)	(5.75%)	(94.23%)	(5.77%)	
							$\bar{x}R$	210429	13030
								(94.17%)	(5.83%)

Tabla 35.- Conjunto 3. Número de estados (true/false) en los procesos

Conjunto 4	S1		S2		S3		x̄R		
	true	false	true	false	true	false	true	false	
8860	292380 (94.29%)	17720 (5.71%)	289479 (93.35%)	20621 (6.65%)	290830 (93.85%)	19071 (6.15%)	290896 (93.83%)	19137 (6.17%)	
9256	195095 (72.93%)	72422 (27.07%)	296161 (93.64%)	20120 (6.36%)	245528 (84.17%)	46171 (15.83%)	245595 (83.58%)	46238 (16.42%)	
10976	355571 (92.99%)	26785 (7.01%)	320115 (87.70%)	44884 (12.30%)	337743 (90.43%)	35735 (9.57%)	337810 (90.38%)	35801 (9.62%)	
11372	136269 (94.23%)	8341 (5.77%)	371684 (94.12%)	23222 (5.88%)	374833 (94.22%)	23007 (5.78%)	294262 (94.19%)	18190 (5.81%)	
							x̄R	292141 (90.50%)	29842 (9.51%)

Tabla 36.- Conjunto 4. Número de estados (true/false) en los procesos

Conjunto	xR	
	true	false
C1	104527 93.98%	6580 6.03%
C2	157607 93.27%	11296 6.73%
C3	210429 94.17%	13030 5.83%
C4	292141 90.50%	29842 9.51%
xR Total	191176 93%	15187 7%

Tabla 37.- Media de todos los conjuntos

Se obtuvo un **93% de eficiencia** en la ejecución de sus procesos de la aplicación con una diferencia de **7% de no éxito**. Para comprobar que el resultado sea el correcto se procedió a analizar los resultados en la herramienta de SPSS.

3.2 ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

Definidas las variables de medición se procedió a especificar el proceso de comprobación de concurrencia para la aplicación móvil especificando el estado de ejecución true o false comparado con la ISO/IEC 25010 en las características de eficiencia de desempeño. Por lo tanto, se procedió a realizar un análisis de medias totales \bar{x} . De esta manera se pudo evidenciar que los tiempos no se vean alterados de una manera abrupta cuando existan un incremento periódico o disparado de peticiones al servidor por parte de carga de usuarios múltiples.

Para la distribución de datos se eligió Shapiro-Wilk, este test permitió determinar si la distribución de la media en el tiempo es normal o no, mediante la herramienta estadística SPSS de IBM, se procedió a describir un resumen estadístico descriptivo de cada uno de los conjuntos analizados. Ver tabla 38

Descriptives

		Statistic	Std. Error	
xSuccessTrueC1	Mean	104526.500	12473.7992	
	95% Confidence Interval for Mean	Lower Bound	64829.304	
		Upper Bound	144223.696	
	5% Trimmed Mean	104743.667		
	Median	106481.000		
	Variance	622382670.333		
	Std. Deviation	24947.5985		
	Minimum	74328.0		
	Maximum	130816.0		
	Range	56488.0		
	Interquartile Range	48027.5		
	Skewness	-.337	1.014	
	Kurtosis	-1.899	2.619	
xSuccessFalseC1	Mean	6580.000	730.8671	
	95% Confidence Interval for Mean	Lower Bound	4254.055	
		Upper Bound	8905.945	
	5% Trimmed Mean	6613.778		
	Median	6884.000		
	Variance	2136666.667		
	Std. Deviation	1461.7341		
Minimum	4558.0			

	Maximum		7994.0	
	Range		3436.0	
	Interquartile Range		2708.0	
	Skewness		-1.122	1.014
	Kurtosis		1.699	2.619
xSuccessTrueC2	Mean		78803.250	45618.2339
	95% Confidence Interval for Mean	Lower Bound Upper Bound	-66374.330 223980.830	
	5% Trimmed Mean		78351.111	
	Median		74734.000	
	Variance		8324093068.9 17	
	Std. Deviation		91236.4679	
	Minimum		.0	
	Maximum		165745.0	
	Range		165745.0	
	Interquartile Range		161675.8	
	Skewness		.027	1.014
	Kurtosis		-5.841	2.619
xSuccessFalseC2	Mean		5648.000	3297.5296
	95% Confidence Interval for Mean	Lower Bound Upper Bound	-4846.211 16142.211	
	5% Trimmed Mean		5581.278	
	Median		5047.500	
	Variance		43494806.000	
	Std. Deviation		6595.0592	
	Minimum		.0	
	Maximum		12497.0	
	Range		12497.0	
	Interquartile Range		11896.5	
	Skewness		.114	1.014
	Kurtosis		-5.348	2.619
xSuccessTrueC3	Mean		210429.250	10717.3125
	95% Confidence Interval for Mean	Lower Bound Upper Bound	176321.978 244536.522	
	5% Trimmed Mean		210356.667	
	Median		209776.000	
	Variance		459443147.58 3	
	Std. Deviation		21434.6250	
	Minimum		187232.0	
	Maximum		234933.0	

	Range		47701.0	
	Interquartile Range		41240.8	
	Skewness		.119	1.014
	Kurtosis		-2.774	2.619
xSuccessFalseC3	Mean		13029.500	682.9864
	95% Confidence Interval for Mean	Lower Bound Upper Bound	10855.932 15203.068	
	5% Trimmed Mean		13044.833	
	Median		13167.500	
	Variance		1865881.667	
	Std. Deviation		1365.9728	
	Minimum		11390.0	
	Maximum		14393.0	
	Range		3003.0	
	Interquartile Range		2608.0	
	Skewness		-.379	1.014
	Kurtosis		-2.671	2.619
xSuccessTrueC4	Mean		292140.750	18837.5431
	95% Confidence Interval for Mean	Lower Bound Upper Bound	232191.281 352090.219	
	5% Trimmed Mean		292189.444	
	Median		292579.000	
	Variance		1419412114.2 50	
	Std. Deviation		37675.0861	
	Minimum		245595.0	
	Maximum		337810.0	
	Range		92215.0	
	Interquartile Range		70002.8	
	Skewness		-.070	1.014
	Kurtosis		1.463	2.619
xSuccessFalseC4	Mean		29841.500	6798.9253
	95% Confidence Interval for Mean	Lower Bound Upper Bound	8204.285 51478.715	
	5% Trimmed Mean		29577.889	
	Median		27469.000	
	Variance		184901541.66 7	
	Std. Deviation		13597.8506	
	Minimum		18190.0	
	Maximum		46238.0	
	Range		28048.0	

Interquartile Range	25202.0	
Skewness	.480	1.014
Kurtosis	-3.253	2.619

Tabla 38.- Descripción de datos procesados

Dado que el conjunto de datos posee una cantidad menor a 50 datos se eligió basar el estudio en el test de Shapiro-Wilk, donde se planteó dos hipótesis, una nula (H_0) y una alternativa (H_a), con un nivel de significancia (α) de un 95% para el nivel de confianza que se estableció obtener.

H_0 : $\bar{x} \sim N$ la serie de \bar{x}_R de C_n es normal

H_a : $\bar{x} \sim N$ la serie de \bar{x}_R de C_n no es normal

$\alpha = 0,05$

El software SPSS generó un valor de significancia (α_{SW}) para cada uno de los conjuntos evaluados., si el valor de α_{SW} es mayor al valor de α se acepta la hipótesis nula (H_0); caso contrario si el valor de significancia es menor se acepta la hipótesis alternativa (H_a). Se entiende como la respuesta tiene una normalidad en los datos si la hipótesis es nula, en cambio si la hipótesis es alterna se debe hacer un reajuste de los datos.

Tests of Normality						
	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
xSuccessTrueC1	.203	4	.	.971	4	.850
xSuccessFalseC1	.261	4	.	.935	4	.626
xSuccessTrueC2	.306	4	.	.771	4	.060
xSuccessFalseC2	.304	4	.	.808	4	.117
xSuccessTrueC3	.206	4	.	.962	4	.793
xSuccessFalseC3	.233	4	.	.945	4	.688
xSuccessTrueC4	.237	4	.	.960	4	.779
xSuccessFalseC4	.284	4	.	.876	4	.323

a. Lilliefors Significance Correction

Figura 28.- Test de normalidad. Validación de datos

$\alpha_{SW} = 0.527$

Una vez realizado el test se obtuvo el siguiente resultado, el valor de significancia (α_{SW}) es mayor al de α , concluyendo que la distribución de estos conjuntos es normal, y al ir aumentando el número de peticiones la tendencia sigue siendo

normal, de igual forma, proporciona los gráficos de todo el test de distribución que se realizó mediante el análisis de Shapiro-Wilk como se muestra en la Figura 29, 30, 31, 32, 33, 34, 35 y 36 .

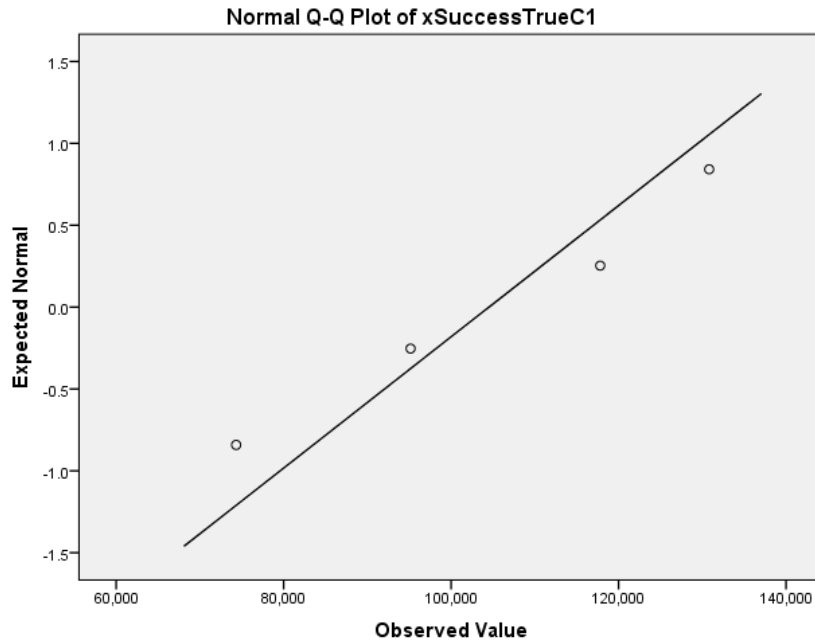


Figura 29.- Gráfica de normalidad. True C1

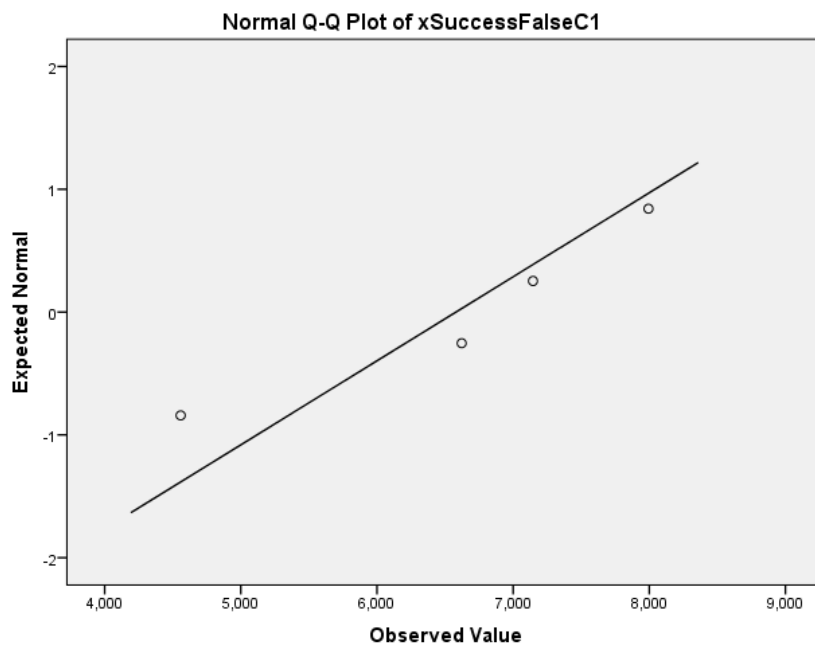


Figura 30.- Gráfica de normalidad. False C1

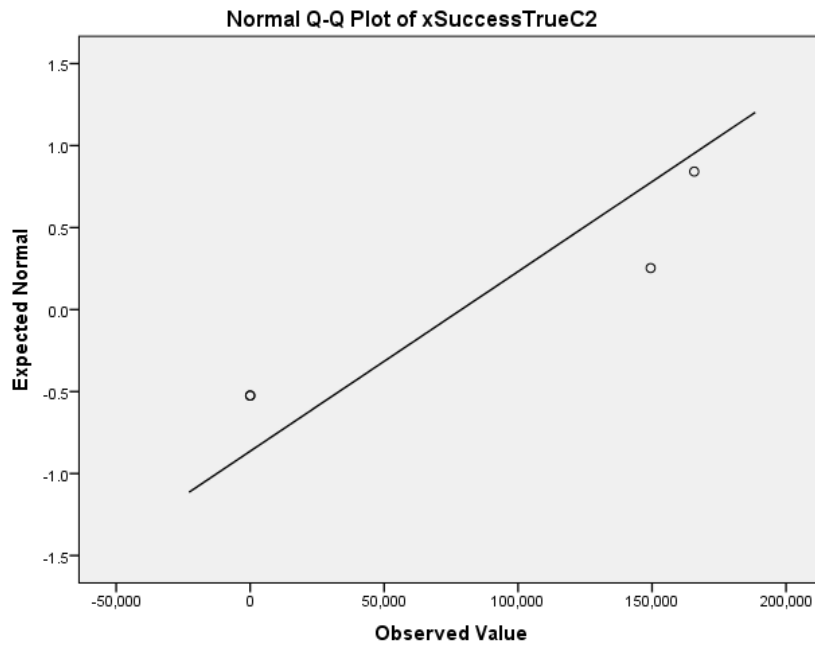


Figura 31.- Gráfica de normalidad. True C2

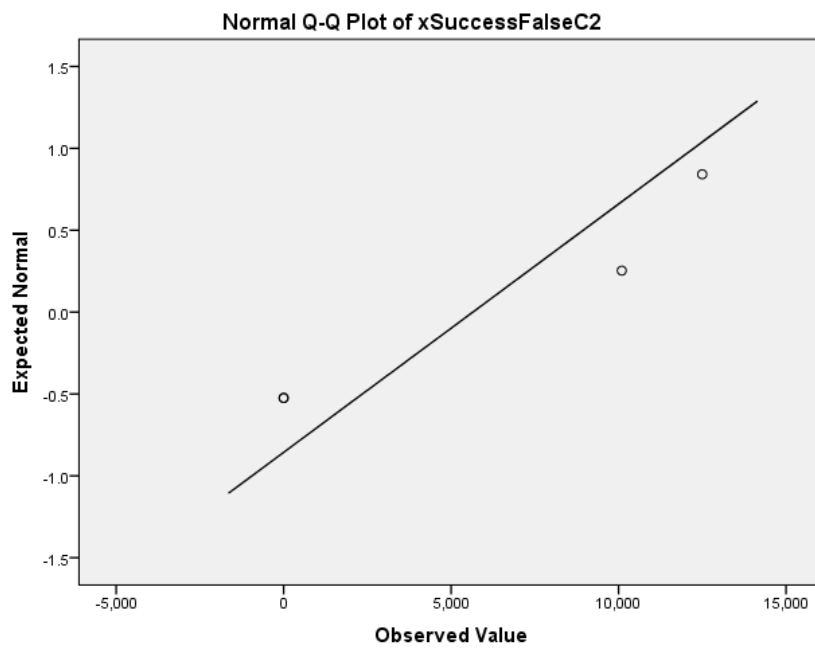


Figura 32.- Gráfica de normalidad. False C2

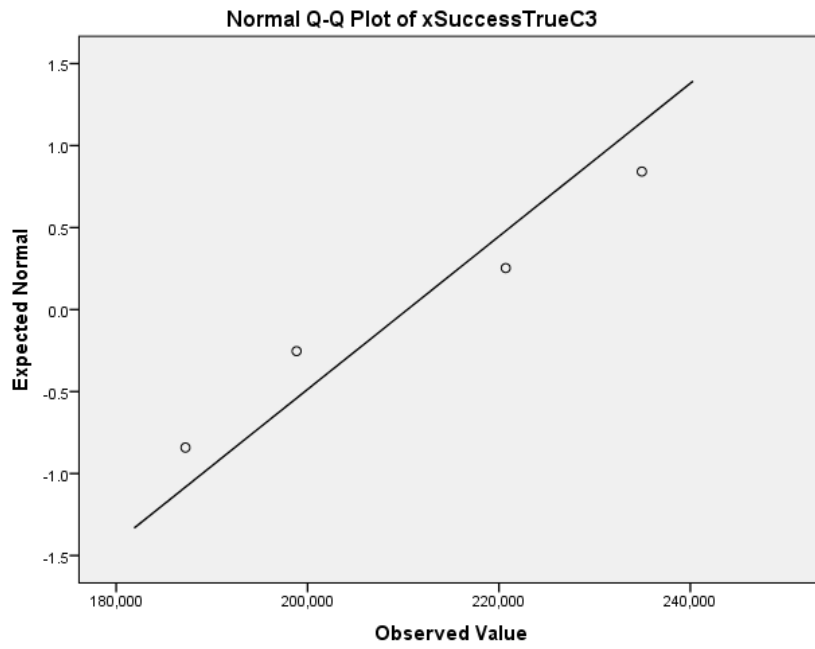


Figura 33.- Gráfica de normalidad. True C3

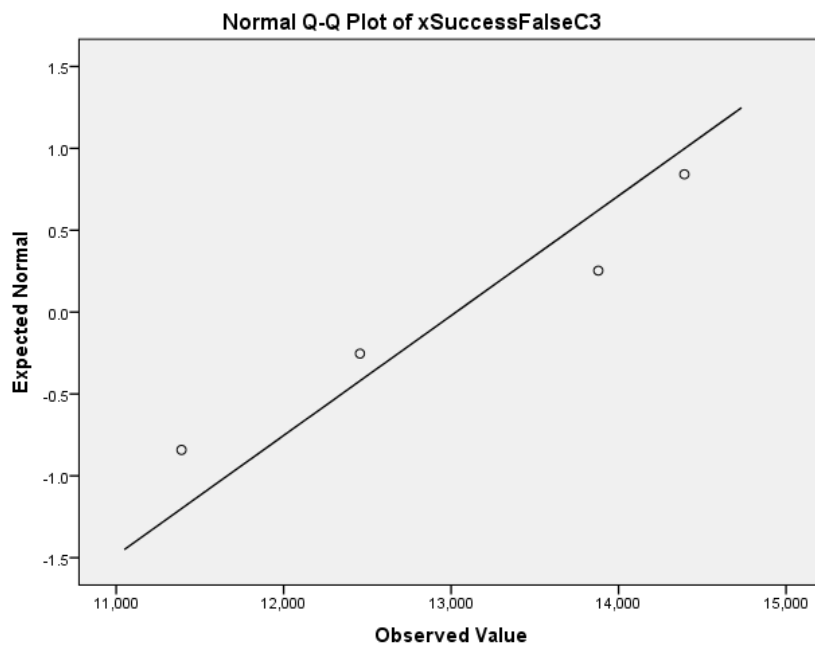


Figura 34.- Gráfica de normalidad. False C3

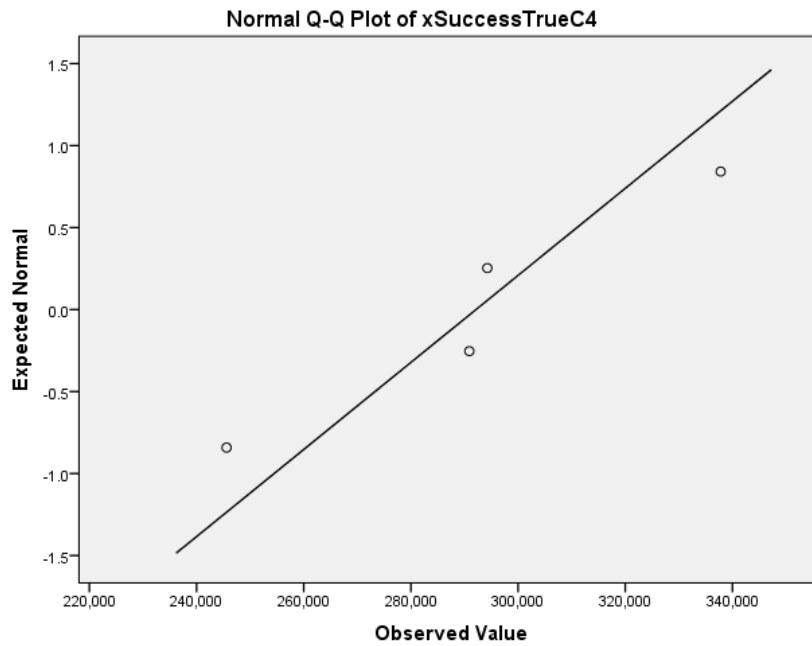


Figura 35.- Gráfica de normalidad. True C4

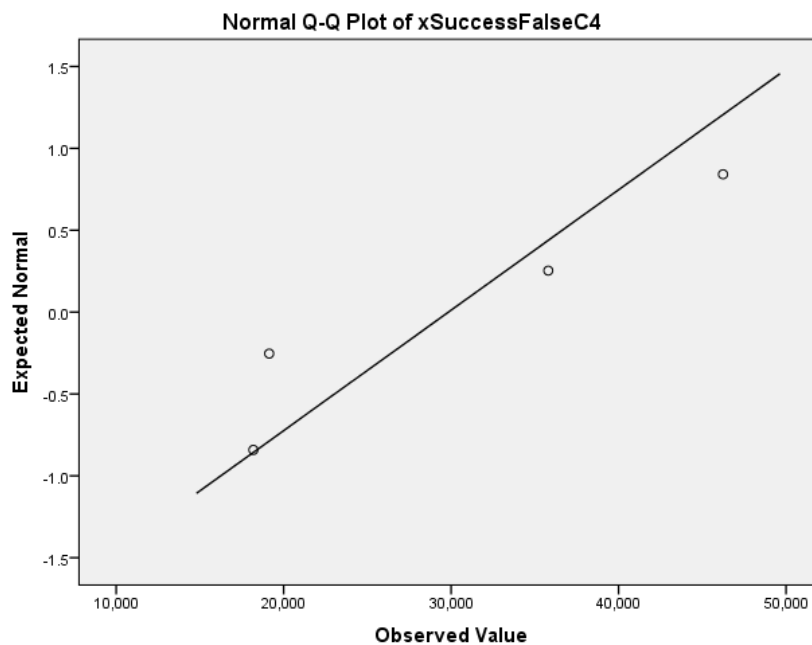


Figura 36.- Gráfica de normalidad. False C4

3.3 ANÁLISIS DE IMPACTOS

Con la implementación de este proyecto se logró fortalecer el proceso de movilidad en la comunidad universitaria mediante una aplicación móvil desarrollada con estándares de calidad en donde se ha comprobado su eficiencia y funcionamiento. Fundamentada principalmente en los Objetivos de Desarrollo Sostenible (ODS) planteado por la Organización de Naciones Unidas (ONU), en la

cual manifiesta una ciudad inteligente donde la movilidad forma parte de este concepto con la ayuda de la tecnología beneficiar a toda una comunidad y transformar la movilidad hacia una que sea sustentable y sostenible.

Cabe recalcar que la universidad es pionera en este tipo de proyectos que van más allá de lo académico, el nombre de la UTN en referencia a urbanismo y movilidad marca una ruta base para futuros proyectos enmarcados en esta temática.

CONCLUSIONES

- La API de Google Maps, mediante la visualización geográfica tanto de la ruta como de la ubicación, brinda una mejor experiencia para el usuario.
- Scrum como marco de trabajo agile permite organizar las actividades de forma más efectiva y eficiente, sobre aquellas tareas que se tenían que realizar de forma manual para el desarrollo de software.
- La herramienta de Oracle Apex RESTful permitió agilizar el trabajo de elaboración de servicios Web para posteriormente consumir en la aplicación móvil.
- El desarrollo de un software evaluado por la característica de Efectividad de Desempeño de la norma ISO/IEC 25010, permite entregar un producto de calidad, mediante métricas y resultados efectivos.
- El tema de movilidad sustentable e inteligente en el campus universitario es un tópico con poca importancia, hace falta estrategias para generar una cultura Smart para que el cambio se vea reflejado en gran proporción.

RECOMENDACIONES

- Comparar los dos procesos tanto del caso conductor, ver imagen 25 como del caso beneficiario, ver imagen 26, para la aceptación del proceso que más resulte eficiente en el proceso de auto compartido.
- Para el desarrollo de aplicaciones híbridas, es preferible utilizar el api de Google Maps JavaScript. Especialmente en Ionic el SDK nativo no está completo y tiene ciertos errores.
- Es necesario seguir añadiendo características a la aplicación móvil tanto en la lógica de negocio como en la usabilidad y experiencia de usuario. De igual forma todas las características añadidas deberán ser medibles tanto el performance como la carga de usuarios.
- Para que el porcentaje de funcionalidad y efectividad incremente, se debe hacer pruebas más específicas tomando en cuenta variables tales como conectividad de red (LAN, Wifi, Móvil), lugar donde se encuentra (con respecto a la universidad).
- Debe existir un proceso de movilidad que sea validado por expertos en movilidad y transporte, para que de esta forma sea ejecutada de una forma más efectiva.
- Es necesario continuar con la ejecución y mantenimiento de este proyecto por parte de la universidad, con la intención de proyectar el interés en un cambio de una movilidad más inteligente y sustentable con el planeta.

GLOSARIO DE TÉRMINOS

API: (Application Programming Interface) Protocolo de comunicación y herramientas para construcción software. (Pardo & Pardo, 2019)

Backlog: Lista ordenada de todo lo necesario en el desarrollo de una historia de usuario. Un listado de todas las funcionalidades que un determinado sistema va a poseer.

Buffering: pre-cargar los datos en memoria. Generalmente utilizado en reproducción de audio y video por internet.

CSV: Comma-separated Values. Es un tipo archivo que está delimitado por comas. Este tipo de archivo únicamente almacena texto plano.

Deployment: es todas las actividades que hacen que un sistema de software esté disponible para su uso. El proceso de implementación general consiste en varias actividades interrelacionadas con posibles transiciones entre ellas.(Lasa Gómez et al., 2017)

Framework: Entorno de software universal que proporciona funcionalidad como parte del propio software.(Lasa Gómez et al., 2017)

IP: Internet Protocol, Protocolo de Internet para la comunicación con las diferentes host o máquinas ya sea en una red local u otro tipo de redes.

ISO/IEC: El Comité Conjunto Técnico ISO/IEC 1 (JTC 1) fue creado en 1987 para "desarrollar, mantener, promover y facilitar los estándares relacionados con la Tecnología de la Información".(Lasa Gómez et al., 2017)

Node JS: es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación ECMAScript, asíncrono.(Lasa Gómez et al., 2017)

Scrum: Metodología ágil de trabajo multipropósito. 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 proyectos.(Lasa Gómez, Alvarez García, & Heras del Dedo, 2017)

SDK: es generalmente un conjunto de herramientas de desarrollo de software que permite a un desarrollador de software crear una aplicación informática para un sistema concreto.(Sanz & López, 2014)

UI: es el medio con que el usuario puede comunicarse con una máquina, equipo, computadora o dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo(Fernández, 2018)

Wrapping: es el proceso de enmarcar los datos principales o un programa que configura otro programa para que pueda ejecutarse con éxito.(Rouse, 2014)

BIBLIOGRAFÍA

- Anderson, K. (2017). What is Rapid Application Development and When Should You Use It? Retrieved May 23, 2018, from <https://blog.capterra.com/what-is-rapid-application-development/>
- Apple. (2015). Apple Open Sources Swift in Latest Pitch to the Enterprise. *The Wall Street Journal Blogs*. Retrieved from <https://blogs.wsj.com/cio/2015/12/03/apple-open-sources-swift-in-latest-pitch-to-the-enterprise/>
- Aratani, L. (2016). New carpooling app matches riders and drivers — for free - The Washington Post. Retrieved April 2, 2019, from https://www.washingtonpost.com/news/dr-gridlock/wp/2016/09/26/new-carpooling-app-matches-riders-and-drivers-for-free/?utm_term=.853c1785805f
- Balalaie, A., Heydarnoori, A., & Jamshidi, P. (2016). Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture. *IEEE Software*, 33(3), 42–52. <https://doi.org/10.1109/MS.2016.64>
- BBVA, F., & BBVA, F. (2015). “wifi” como término español. Retrieved from <http://www.fundeu.es/recomendacion/wifi-como-termino-espanol-545/>
- BELZ, N. P., & LEE, B. H. Y. (2013). *Composition of Vehicle Occupancy for Journey-To-Work Trips*. LA, US. Retrieved from <http://www.uvm.edu/~transctr/research/grad/12-0489.pdf>
- Bhuiyan, J. (2018). Uber’s first diversity report under CEO Khosrowshahi: Still white, male - Recode. Retrieved April 14, 2019, from <https://www.recode.net/2018/4/25/17275010/uber-executive-diversity-black-latinx>
- Binstock, A. (2015). *Java’s 20 Years of Innovation*. Forbes. Retrieved from <https://www.forbes.com/sites/oracle/2015/05/20/javas-20-years-of-innovation/>
- Birla, S., & Johansson, M. (2015). Quality Requirements for Software-dependent Safety-critical Systems History, current status, and future needs. Retrieved from <https://pdfs.semanticscholar.org/5b49/2c54560642a6b5ed100fec94f50e75b34704.pdf>

- Blasco, L. (2016). Cuáles son las diferencias entre E, GPRS, 3G, 4G, 5G y esas otras redes a las que se conecta tu celular (y cómo te afectan tu conexión a internet) - BBC Mundo. Retrieved May 5, 2018, from <http://www.bbc.com/mundo/noticias-37247130>
- Bohmer, A. I., Hugger, P., & Lindemann, U. (2017). Scrum within hardware development insights of the application of scrum for the development of a passive exoskeleton. In *2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC)* (pp. 790–798). IEEE. <https://doi.org/10.1109/ICE.2017.8279965>
- Borrvalho, R., Ferreira, S., Vieira, P., Queluz, M. P., & Rodrigues, A. (2017). A novel localization C# framework for RAN optimization using extreme programming methodologies. In *2017 20th International Symposium on Wireless Personal Multimedia Communications (WPMC)* (pp. 516–521). IEEE. <https://doi.org/10.1109/WPMC.2017.8301867>
- Cardona, A. (2016). El transporte terrestre y la historia de la humanidad | Sertrans. Retrieved April 2, 2019, from <https://www.sertrans.es/trasporte-terrestre/el-transporte-terrestre-la-historia-de-la-humanidad/>
- Caro, A., Fuentes, A., & Soto, M. A. (2013). Desarrollando sistemas de información centrados en la calidad de datos. *Ingeniare. Revista Chilena de Ingeniería*, *21*(1), 54–69. <https://doi.org/10.4067/S0718-33052013000100006>
- Chin, G. (2014). *Agile Project Management: How to Succeed in the Face of Changing Project Requirements*. AMACOM.
- Christou, I., Ponis, S., & Palaiologou, E. (2013). Using the Agile Unified Process in Banking. *IEEE Software*, *27*(3), 72–79. <https://doi.org/10.1109/MS.2009.156>
- Cockburn, A. (2014). *Crystal clear a human-powered methodology for small teams*.
- Cockburn, A. (2015). *Agile Software Development*. Highsmith Series.
- Corwin, S., Vitale, J., Kelly, E., & Cathles, E. (2015). *El futuro de la movilidad*. Madrid, ES. Retrieved from [https://www2.deloitte.com/content/dam/Deloitte/ec/Documents/manufacturing/DUP_Future_of_Mobility_ES_WEB_\(00000002\).pdf](https://www2.deloitte.com/content/dam/Deloitte/ec/Documents/manufacturing/DUP_Future_of_Mobility_ES_WEB_(00000002).pdf)
- Cox, B. J., & Novobilski, A. J. (2017). *Object-oriented programming : an evolutionary approach*. Addison-Wesley Pub. Co.

- Cuesta, L. (2018). Aplicaciones Web Progresivas - La solución móvil más completa. Retrieved May 17, 2018, from <https://www.tu-app.net/blog/aplicaciones-web-progresivas/>
- David, J., & Pozo, M. (2015). *Introducción a los dispositivos móviles*. (Editorial UOC, Ed.). Barcelona: Editorial UOC.
- Deloach, S. B., & Tiemann, T. K. (2012). *DEPARTMENT OF ECONOMICS WORKING PAPER SERIES Not driving alone: Commuting in the Twenty-first century*. Retrieved from <http://org.elon.edu/econ/WPS/wp2010-01.pdf>
- Department of Transportation, U., & Highway Administration, F. (2016). *Smartphone Applications To Influence Travel Choices Practices and Policies*. Washington. Retrieved from <https://ops.fhwa.dot.gov/Publications/fhwahop16023/fhwahop16023.pdf>
- Dilamani, M. T. (2014). A short review on Crystal Clear methodology and its advantages over scrum, the popular software process model, 4.
- Drayton, P., Albahari, B., & Neward, T. (2015). *C# language pocket reference*. O'Reilly.
- Dredge, S. (2013). Mobile apps revenues tipped to reach \$26bn in 2013 | Technology | The Guardian. Retrieved April 3, 2019, from <https://www.theguardian.com/technology/appsblog/2013/sep/19/gartner-mobile-apps-revenues-report>
- Dutzik, T., Madsen, T., & Baxandall, P. (2013). *A New Way to Go: The Transportation Apps and Vehicle-Sharing Tools that Are Giving More Americans the Freedom to Drive Less*. Retrieved from www.frontiergroup.org.
- Essay UK. (2016). Evolution of Mobile Device Technology. Retrieved May 1, 2018, from <https://www.ukessays.com/essays/computer-science/evolution-mobile-device-technology-9171.php>
- Evans, K. (2017). 5 Programming Languages For Mobile App Development. Retrieved May 19, 2018, from <https://dev.to/kennethevans107/5-programming-languages-for-mobile-app-development--2ec5>
- Feig, N., & Feig, N. (2007). *Mobile Payments: Look to Korea*. banktech.com. Retrieved from http://www.banktech.com/blog/archives/2007/06/mobile_payments.html

- Fernández, P. E. C. (2018). *Usabilidad web: teoría y uso*. RA-MA Editorial.
- Ferrao, S. E. R., & Dias Canedo, E. (2015). A study of the applicability of an agile methodology scrum allied to the Kanban method. In *2015 10th Iberian Conference on Information Systems and Technologies (CISTI)* (pp. 1–6). IEEE. <https://doi.org/10.1109/CISTI.2015.7170382>
- Frankston, B. (2018). Progressive Web Apps [Bits Versus Electrons]. *IEEE Consumer Electronics Magazine*, 7(2), 106–117. <https://doi.org/10.1109/MCE.2017.2776463>
- Friedman, N. (2018). Pros and Cons of Xamarin vs Native Mobile Development. Retrieved April 6, 2019, from <https://www.altexsoft.com/blog/mobile/pros-and-cons-of-xamarin-vs-native/>
- Galín, D. (2018). Software Quality Factors (Attributes). In *Software Quality: Concepts and Practice* (pp. 23–44). Hoboken, NJ, USA: John Wiley & Sons, Inc. <https://doi.org/10.1002/9781119134527.ch2>
- Gallelgo, J. (2016). Línea del Tiempo- Evolución de los dispositivos móviles. Retrieved May 1, 2018, from <https://www.slideshare.net/JocelyneGallegos/linea-del-tiempo-evolucion-de-los-dispositivos-mviles>
- Gannes, L. (2013). NextBus Acquired by Cubic for \$20.6M - Liz Gannes - News - AllThingsD. Retrieved April 14, 2019, from <http://allthingsd.com/20130127/nextbus-acquired-by-cubic-for-20-6m/?refcat=news>
- González Verdugo, D. (2016). Desarrollo de aplicaciones móviles híbridas con el framework Ionic. Retrieved May 16, 2018, from <https://solidgargroup.com/desarrollo-de-apps-hibridas-con-ionic?lang=es>
- Gosling, J., Joy, B., Steele, Guy L., J., & Bracha, G. (2015). *The Java language specification* (3rd ed.). Addison-Wesley. Retrieved from <http://java.sun.com/docs/books/jls/index.html>
- Haas, H. (2013). High-speed wireless networking using visible light. *SPIE Newsroom*. <https://doi.org/10.1117/2.1201304.004773>
- Haverbeke, M. (2014). *Eloquent Javascript : a modern introduction to programming*. No Starch Press.

- ISO. (2011). *ISO/IEC 25010:2011 : Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models*. Retrieved from http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnnumber=35733
- Johnston, K. (2014). Pop-up bus service Bridj to launch test runs June 2 - The Boston Globe. Retrieved April 14, 2019, from <https://www.bostonglobe.com/business/2014/05/13/pop-bus-service-bridj-launch-test-runs-june/tOGnFPBvXISyzma28F71NK/story.html>
- Karam, L. M. (2017a). Scrum vs. Kanban. Retrieved November 13, 2018, from <https://dzone.com/articles/scrum-vs-kanban-what-is-the-difference>
- Karam, L. M. (2017b). What Are Scrum Sprints? Retrieved November 13, 2018, from <https://dzone.com/articles/a-complete-scrum-sprint-explanation>
- Kiernan, B. (2014). IEEE Approves IEEE 802.16m – Advanced Mobile Broadband Wireless Standard. *IEEE*. Retrieved from <http://standards.ieee.org/news/2011/80216m.html>
- Kim, H.-K. (2013). Architecture for Adaptive Mobile Applications. *International Journal of Bio-Science and Bio-Technology*, 5(5), 197–210. <https://doi.org/10.14257/ijbsbt.2013.5.5.21>
- Konovalov, D. A. L. A. L., & Sychugov, S. G. (2018). Planar antenna array for 3G and 4G wireless communication. In *2018 Moscow Workshop on Electronic and Networking Technologies (MWENT)* (pp. 1–4). IEEE. <https://doi.org/10.1109/MWENT.2018.8337279>
- Kortum, K., Schönduwe, R., Stolte, B., & Bock, B. (2016). Free-Floating Carsharing: City-Specific Growth Rates and Success Factors. *Transportation Research Procedia*, 19, 328–340. <https://doi.org/10.1016/J.TRPRO.2016.12.092>
- Krill, P. (2015). JetBrains readies JVM language Kotlin. *Infoworld.Com*. Retrieved from <https://www.infoworld.com/d/application-development/jetbrains-readies-jvm-based-language-167875>
- Lasa Gómez, C., Alvarez García, A., & Heras del Dedo, R. de las. (2017). *Métodos ágiles Scrum, Kanban, Lean*. Anaya Multimedia.
- Lowe, D. (2016). What is Crystal Clear? *Scrum&Kanbam*. Retrieved from

- <https://scrumandkanban.co.uk/what-is-crystal-clear/>
- Lynch, M. (2016). How 2015 Went for Ionic | The Ionic Blog. Retrieved April 6, 2019, from <https://blog.ionicframework.com/how-2015-went-for-ionic/>
- Martínez, E. (2014). Las Metodologías Ágiles: qué son y cuáles son sus ventajas. Retrieved May 24, 2018, from <https://www.iebschool.com/blog/que-es-agile-agile-scrum/>
- Martinez, G. (2017). Desarrollo de aplicaciones móviles híbridas ¿Me conviene? - Northware. Retrieved May 16, 2018, from <http://www.northware.mx/desarrollo-de-aplicaciones-moviles-hibridas/>
- Martinez, R. (2015). Arquitectura para la Implementación de Sistemas Móviles basados en servicios de Geolocalización y Crowdsourcing. Retrieved from <http://imgbiblio.vaneduc.edu.ar/fulltext/files/TC119363.pdf>
- Maximini, D. (2015). *The scrum culture : introducing agile methods in organizations*. Cham: Springer. Retrieved from <https://books.google.com/books?id=ShojBgAAQBAJ>
- Mehul, R. (2018). React Native — Is it Really the Future of Mobile App Development? Retrieved April 6, 2019, from <https://hackernoon.com/react-native-is-it-really-the-future-of-mobile-app-development-31cb2c531747>
- Miguel, J. P., Mauricio, D., & Rodríguez, G. (2014). A Review of Software Quality Models for the Evaluation of Software Products. *International Journal of Software Engineering & Applications (IJSEA)*, 5(6). <https://doi.org/10.5121/ijsea.2014.5603>
- Mundra, A., Misra, S., & Dhawale, C. A. (2013). Practical Scrum-Scrum Team: Way to Produce Successful and Quality Software. In *2013 13th International Conference on Computational Science and Its Applications* (pp. 119–123). IEEE. <https://doi.org/10.1109/ICCSA.2013.25>
- Nouh, D. M., Abdelaziz, B., Faissal, E., Anass, L., Nazha, C., & Jamal, B. (2017). A M2M communication based on GPRS and IEC61850 for smart substations remote control. In *2017 International Conference on Electrical and Information Technologies (ICEIT)* (pp. 1–5). IEEE. <https://doi.org/10.1109/EITech.2017.8255276>
- O'Brien, S. A. (2018). Cars, bikes, scooters: Uber and Lyft battle it out. Retrieved April 14, 2019, from <https://money.cnn.com/2018/07/03/technology/uber->

lyft-bikes-scooters/index.html

- Occhino, T. (2016). React Native: Bringing modern web techniques to mobile - Facebook Code. Retrieved April 6, 2019, from <https://code.fb.com/android/react-native-bringing-modern-web-techniques-to-mobile/>
- Otemuyiwa, P. (2017). Introduction to Progressive Web Apps (Offline First) - DZone Mobile. Retrieved October 30, 2018, from <https://dzone.com/articles/introduction-to-progressive-web-apps-offline-first>
- Pardo, D., & Pardo, D. (2019). ¿Para qué sirve una API? Solventa por fin esta duda de primerizo. *Pandora FMS*. Retrieved from <https://blog.pandorafms.org/es/para-que-sirve-una-api/>
- Peppers, J. (2014). *Xamarin Cross-platform Application Development*. Olton, UNITED KINGDOM: Packt Publishing Ltd. Retrieved from <http://ebookcentral.proquest.com/lib/utn/detail.action?docID=1582945>
- Petzold, C. (2012). *Programming Microsoft Windows with C#*. Microsoft Press.
- RAE. (2019). Definición de transporte - Qué es, Significado y Concepto. Retrieved April 2, 2019, from <https://definicion.de/transporte/>
- Redchets, A. (2017). Java vs. Kotlin: ¿Cuál es la mejor opción para el desarrollo de aplicaciones de Android? | Clearbridge Mobile. Retrieved May 19, 2018, from <https://clearbridgemobile.com/java-vs-kotlin-which-is-the-better-option-for-android-app-development/>
- Rosenwasser, D. (2019). Announcing TypeScript 3.4 RC | TypeScript. Retrieved April 3, 2019, from <https://devblogs.microsoft.com/typescript/announcing-typescript-3-4-rc/>
- Rouse, M. (2014). What is wrapper? - Definition from WhatIs.com. Retrieved July 28, 2019, from <https://searchmicroservices.techtarget.com/definition/wrapper>
- Rouse, M. (2016). ¿Qué es Desarrollo de aplicaciones móviles híbridas? - Definición en WhatIs.com. *TechTarget*. Retrieved from <https://searchdatacenter.techtarget.com/es/definicion/Desarrollo-de-aplicaciones-moviles-hibridas>

- Russell, A. (2015). Progressive Web Apps: Escaping Tabs Without Losing Our Soul – Infrequently Noted. Retrieved October 30, 2018, from <https://infrequently.org/2015/06/progressive-apps-escaping-tabs-without-losing-our-soul/>
- Russell, K. (2014). *Swift, Apple's New Programming Language, Has Been In Development For Nearly Four Years | TechCrunch*. EE.UU. Retrieved from <https://techcrunch.com/2014/06/04/apples-new-programming-language-has-been-in-development-for-nearly-four-years/>
- Sanz, R. L., & López, R. G. (2014). *Introducción a la movilidad: 4G/LTE y el desarrollo de aplicaciones Android*. (Dextra Editorial, Ed.). Barcelona, ES: Dextra Editorial.
- Shaheen, S., Martin, E., & Cohen Apoorva Musunuri Abhinav Bhattacharyya, A. (2016). *Mobile Apps and Transportation: A Review of Smartphone Apps and a Study of User Response to Multimodal Traveler Information*. Berkley. Retrieved from http://innovativemobility.org/wp-content/uploads/MobileApps_UCCONNECT-1.pdf
- Sharma, P., & Hasteer, N. (2016a). Analysis of linear sequential and extreme programming development methodology for a gaming application. In *International Conference on Communication and Signal Processing, ICCSP 2016* (pp. 1916–1920). IEEE. <https://doi.org/10.1109/ICCSP.2016.7754505>
- Sharma, P., & Hasteer, N. (2016b). Analysis of linear sequential and extreme programming development methodology for a gaming application. *2016 International Conference on Communication and Signal Processing (ICCSP)*, 1916–1920. <https://doi.org/10.1109/ICCSP.2016.7754505>
- Siuhi, S., & Mwakalonge, J. (2016). Opportunities and challenges of smart mobile applications in transportation. *Journal of Traffic and Transportation Engineering (English Edition)*, 3(6), 582–592. <https://doi.org/10.1016/j.jtte.2016.11.001>
- Srivastava, A., Bhardwaj, S., & Saraswat, S. (2017). SCRUM model for agile methodology. In *2017 International Conference on Computing, Communication and Automation (ICCCA)* (pp. 864–869). IEEE. <https://doi.org/10.1109/CCAA.2017.8229928>
- Stabile, A. G. (2015). Desarrollo ágil de software: Crystal Clear | Folder IT. Retrieved May 25, 2018, from <https://folderit.net/itech/es/desarrollo-agil->

de-software-crystal-clear-es/

- Stapic, Z., Mijac, M., & Strahonja, V. (2016). Methodologies for development of mobile applications. In *2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* (pp. 688–692). IEEE. <https://doi.org/10.1109/MIPRO.2016.7522228>
- Summers, M. (2018). Overview of mobile app development architecture. *Magora*. Retrieved from <https://magora-systems.com/mobile-app-development-architecture/>
- Tilley, C. (2016). *The History of Windows CE: Windows CE 1*. HPC:Factor. Retrieved from <http://www.hpcfactor.com/qmlink/?linkID=20>
- Torres, M. M. (2013). *TRANSPORTE, OPERADORES, REDES* (Papel). Madrid, ES. Retrieved from <https://ebookcentral.proquest.com/lib/utnortesp/reader.action?docID=3229404&query=transporte>
- Tsonev, D., Videv, S., & Haas, H. (2013). Light fidelity (Li-Fi): towards all-optical networking. In B. B. Dingel & K. Tsukamoto (Eds.), *Proc. SPIE* (Vol. 9007, p. 900702). <https://doi.org/10.1117/12.2044649>
- UPV. (2012). Movilidad y transporte : Cátedra Municipios Sostenibles : UPV. Retrieved April 2, 2019, from <http://www.upv.es/contenidos/CAMUNISO/info/820042normalc.html>
- Valipour, M. H., Amirzafari, B., Maleki, K. N., & Daneshpour, N. (2009). A brief survey of software architecture concepts and service oriented architecture. In *2009 2nd IEEE International Conference on Computer Science and Information Technology* (pp. 34–38). IEEE. <https://doi.org/10.1109/ICCSIT.2009.5235004>
- Vesela, V., & Krbecek, M. (2016). Development methodologies of mobile applications. In *2016 International Conference on Interactive Mobile Communication, Technologies and Learning (IMCL)* (pp. 3–4). IEEE. <https://doi.org/10.1109/IMCTL.2016.7753760>
- Viken, A. (2015). PDA - handheld computer. *Britannica.Com*. Retrieved from <https://www.britannica.com/technology/PDA>
- Wakode, R. B., Raut, L. P., & Talmale, P. (2015). Overview on Kanban Methodology and its Implementation Overview on Kanban Methodology and its

- Implementation. *Ijsrd*, 3(October), 2518–2521.
- Walton, M. (2015). Is “Wi-Fi on steroids” really the next big thing? *CNN*. Retrieved from http://articles.cnn.com/2005-10-17/tech/wireless.wimax_1_wi-fi-georgia-s-new-media-institute-scott-shamp?_s=PM:TECH
- Wauters, R. (2014). Ride-sharing in Europe: the race between BlaBlaCar and Carpooling.com. Retrieved April 2, 2019, from <https://tech.eu/features/481/ride-sharing-europe-carpooling-blablacar/>
- Wellington, C. A. (2015). Managing a Project Course Using Extreme Programming. In *Proceedings Frontiers in Education 35th Annual Conference* (pp. T3G-1-T3G-5). IEEE. <https://doi.org/10.1109/FIE.2005.1611948>
- Wray, S. (2018). What Is 5G and when does it arrive? Retrieved May 10, 2018, from <https://5g.co.uk/guides/what-is-5g/>
- Yusuf, S. (2016). *Ionic framework by example : build amazing cross-platform mobile apps with Ionic, the HTML5 framework that makes modern mobile application development simple*. Birmingham.

ANEXOS

ANEXO 1. IMPLEMENTACIÓN Y DESPLIEGUE DE ARQUITECTURA DE SOFTWARE

1.1 INSTALACIÓN DE AMBIENTES DE TRABAJO

Se descarga e instala el IDE Visual Studio Code para el manejo de los directorios del proyecto en Ionic. Desde la página principal de VS Code.

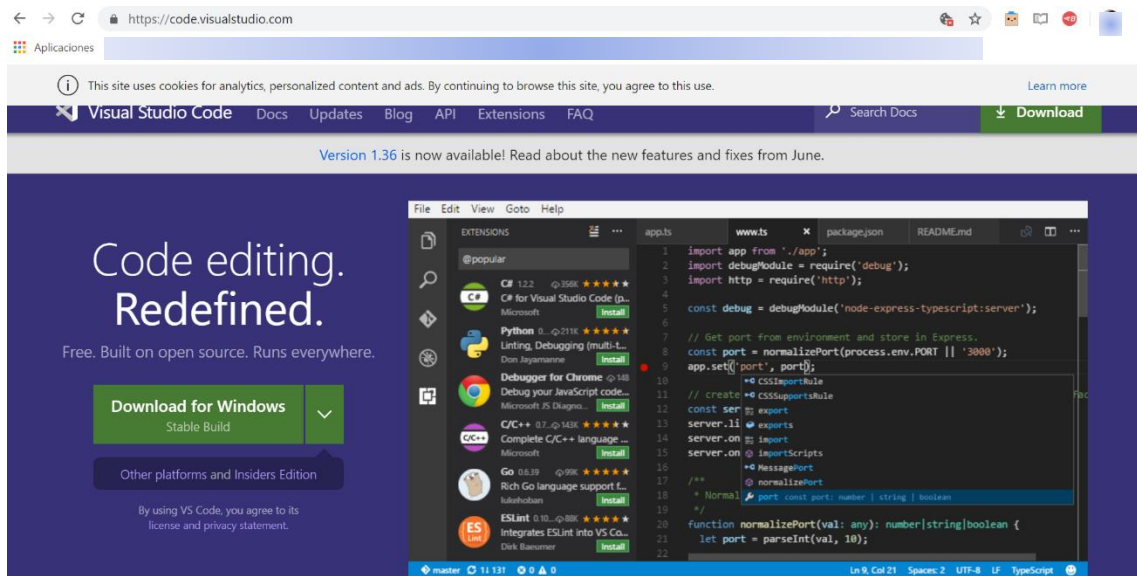


Figura 37.- Descarga de Visual Studio Code

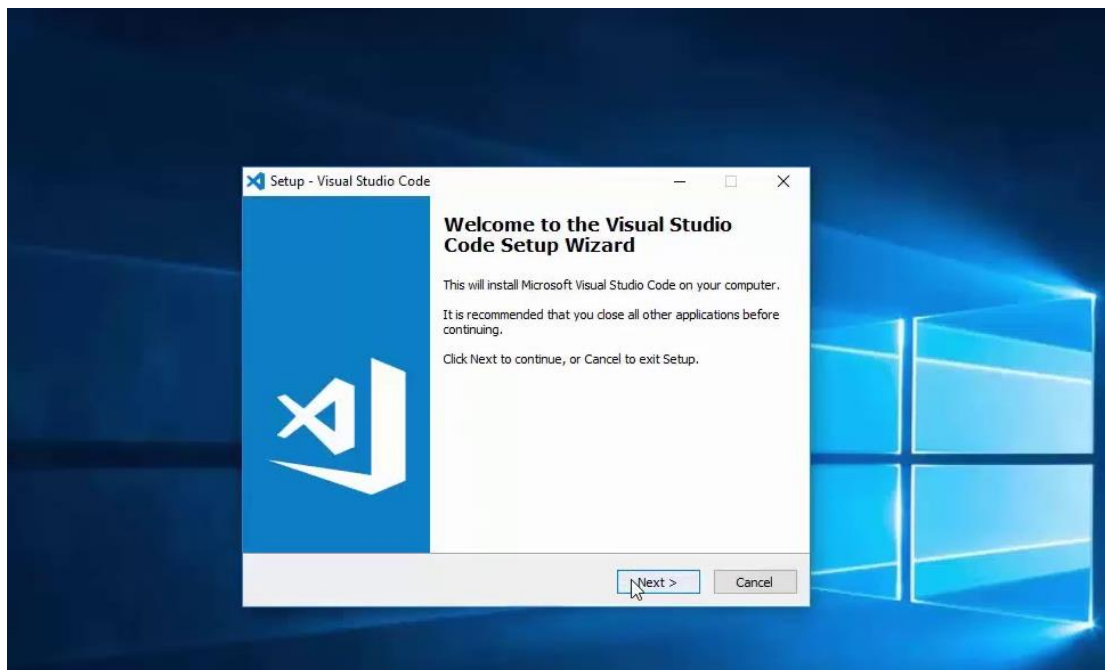


Figura 38.- Instalación de Visual Studio Code

1.1.1 Ionic 3.9.2

Para la instalación de este framework antes se debe instalar la herramienta de Node.js desde la página oficial.

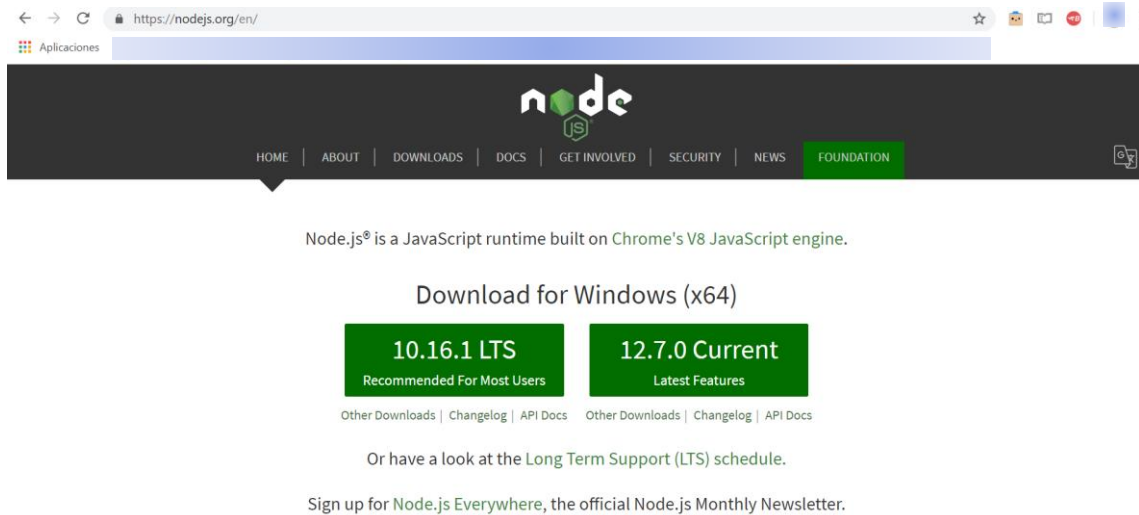


Figura 39.- Descarga de Node.js

Posteriormente desde el IDE Visual Studio Code o desde un símbolo de sistema en Windows, se ingresa la siguiente instrucción `npm install ionic @3.9.2`, en la version 3.9.2

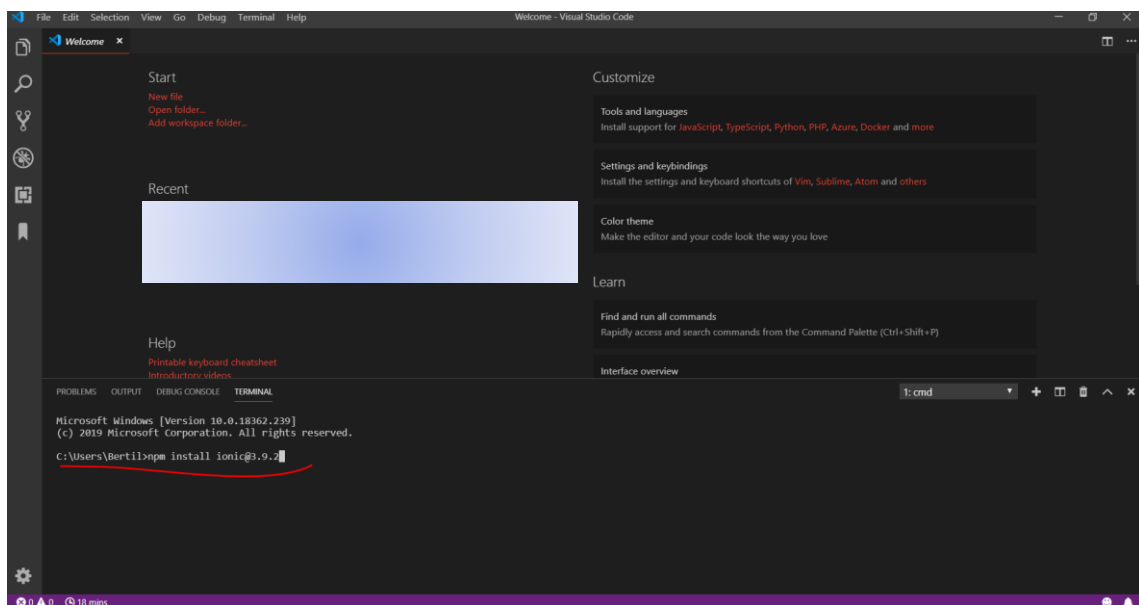


Figura 40.- Instalación Ionic Framework

1.1.2 Cordova

Este framework utiliza una arquitectura híbrida para el desarrollo de aplicaciones móviles, por lo cual se necesita de una librería (Cordova) que permita el acceso al hardware nativo del teléfono móvil. Con la siguiente instrucción `npm install -g ionic cordova`.

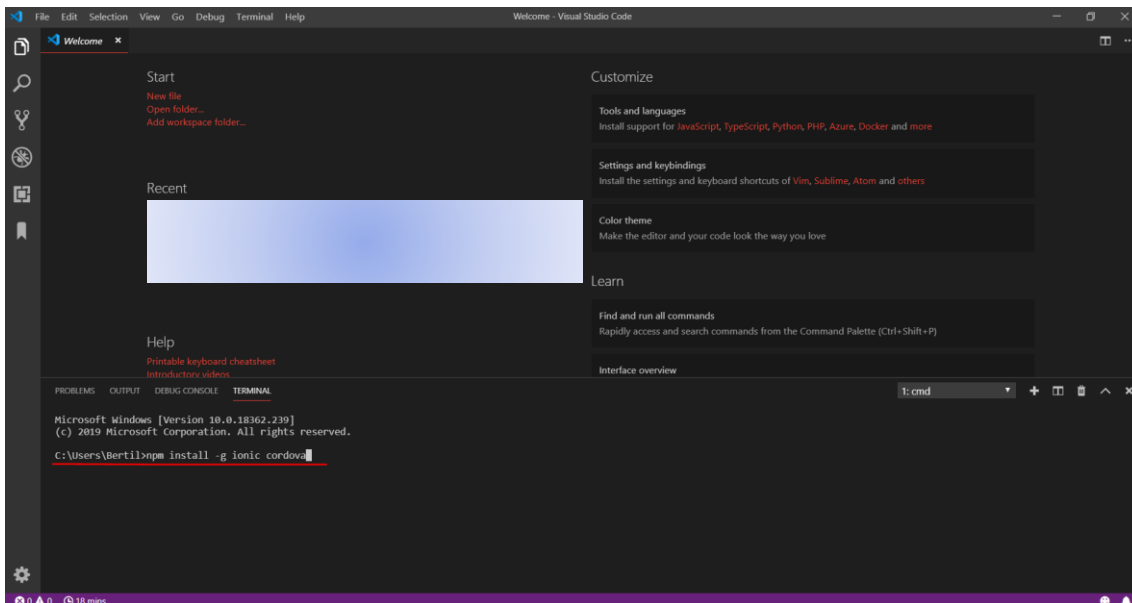


Figura 41.- Instalación de cordova

1.1.1 Android SDK & Android Studio

Se instala Android Studio el cual viene incluido Android SDK. Se procede a descargar la versión más reciente de API de Android que se esté manejando para este caso es el API 28.

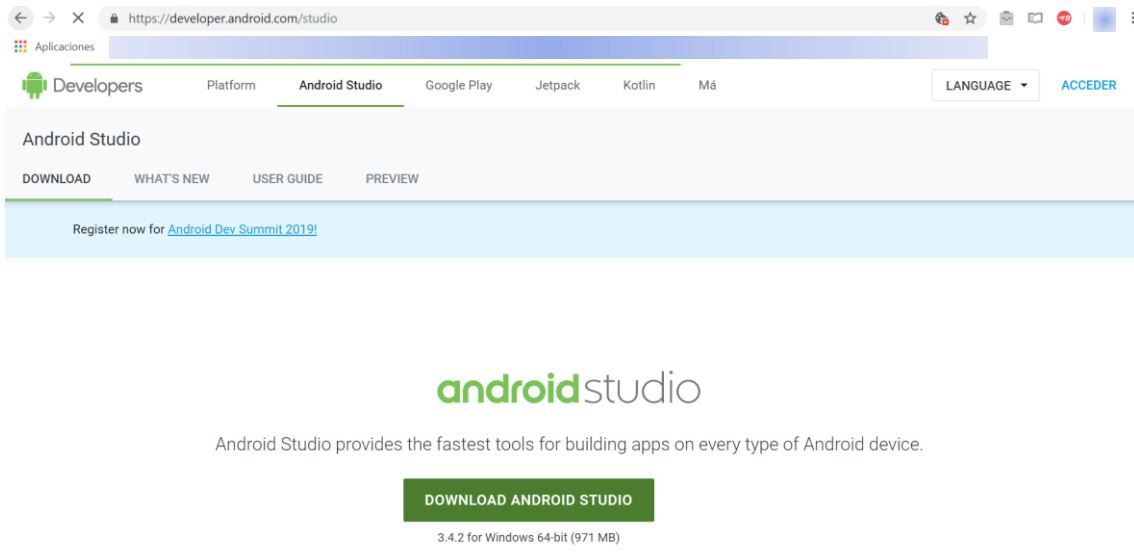


Figura 42.- Descarga de herramienta Android Studio

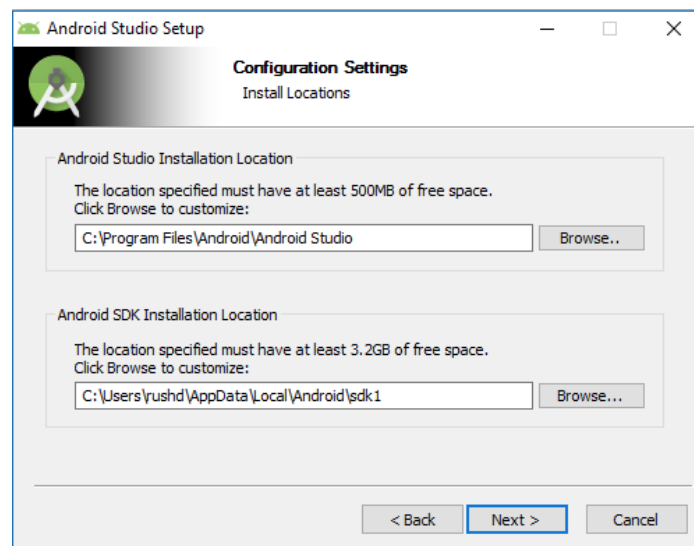


Figura 43.-Instalación Android Studio

Se procede a la configuración de Android SDK Tools para el proyecto de Ionic.

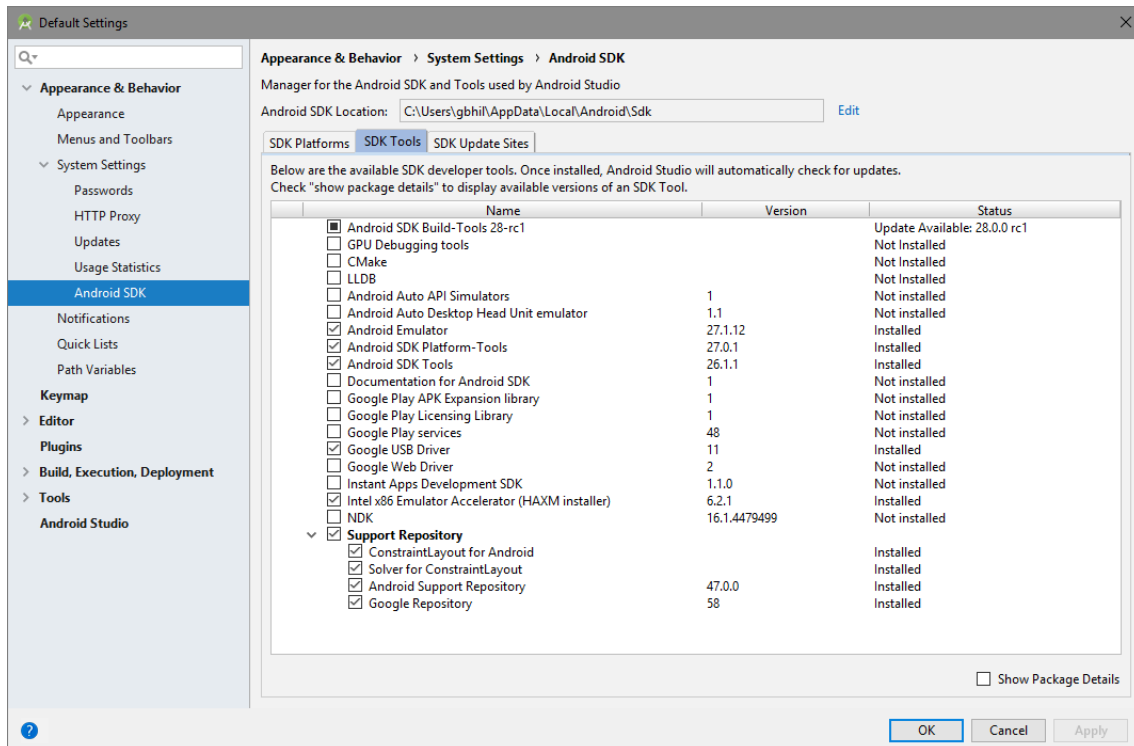


Figura 44.- Configuración de Android SDK Tools

1.2 EJECUCIÓN DEL PROYECTO

1.2.1 Creación

De igual forma que la instalación se puede realizar desde IDE de VS Code o desde un terminal de Windows, mediante la siguiente instrucción `ionic start [appname] [blank-tabs-none]`

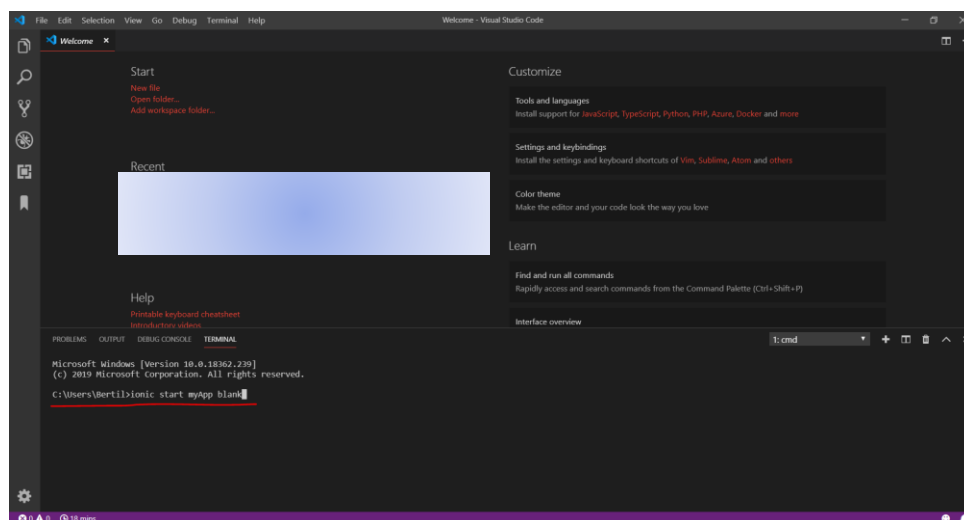


Figura 45.-Creación de proyecto Ionic

Por otra parte, para la compilación del proyecto en iOS con el mismo IDE de VS Code se puede realizar, pero para la exportación de la aplicación como tal es necesario instalar la herramienta de XCode dentro de un sistema de operativo de Macintosh. Este procedimiento no se realizó las razones se explican en el desarrollo del proyecto. Pag 55

1.2.2 Ejecución

Se ejecuta el proyecto desde el IDE de VS Code con la siguiente instrucción dependiendo si vas a visualizar en navegador en un emulador o directamente en un dispositivo móvil.

ionic cordova run browser

ionic serve

ionic cordova run android

ionic cordova run ios

ionic cordova run emulator

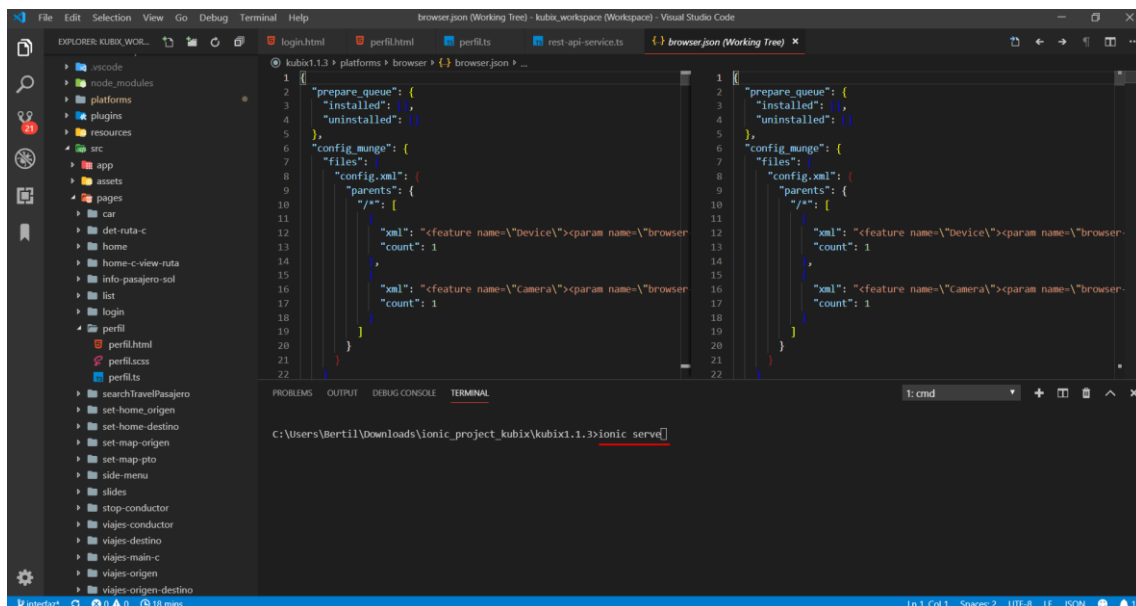


Figura 46.- Ejecución de proyecto

De esta forma se ejecuta en el navegador de forma no nativa. Pero si se quiere ejecutar de forma nativa y de esta forma probar hardware como cámara o almacenamiento u otros, se debe ejecutar *cordova run*.

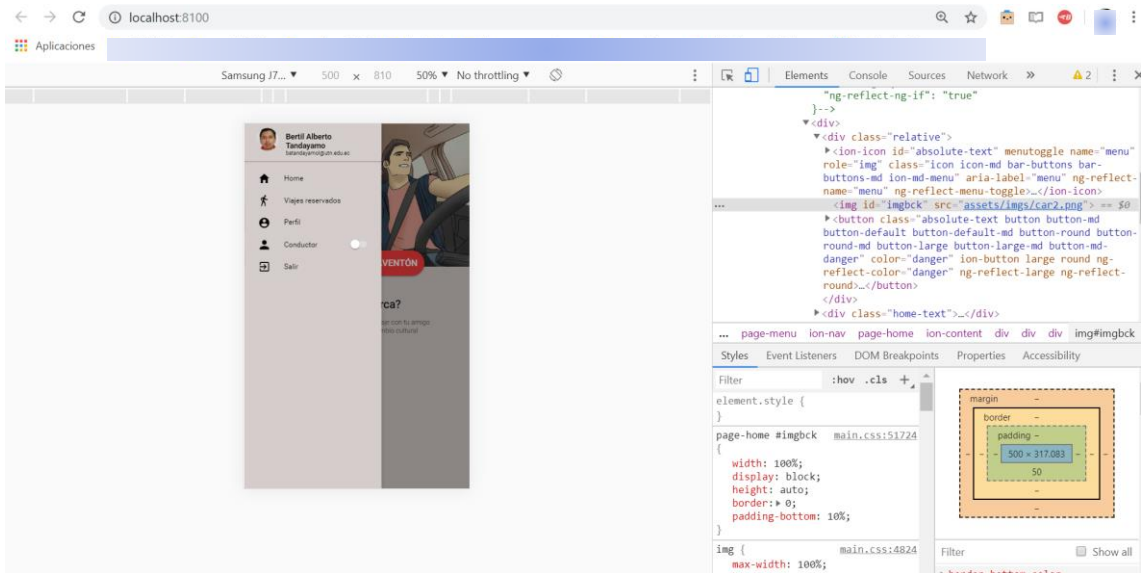
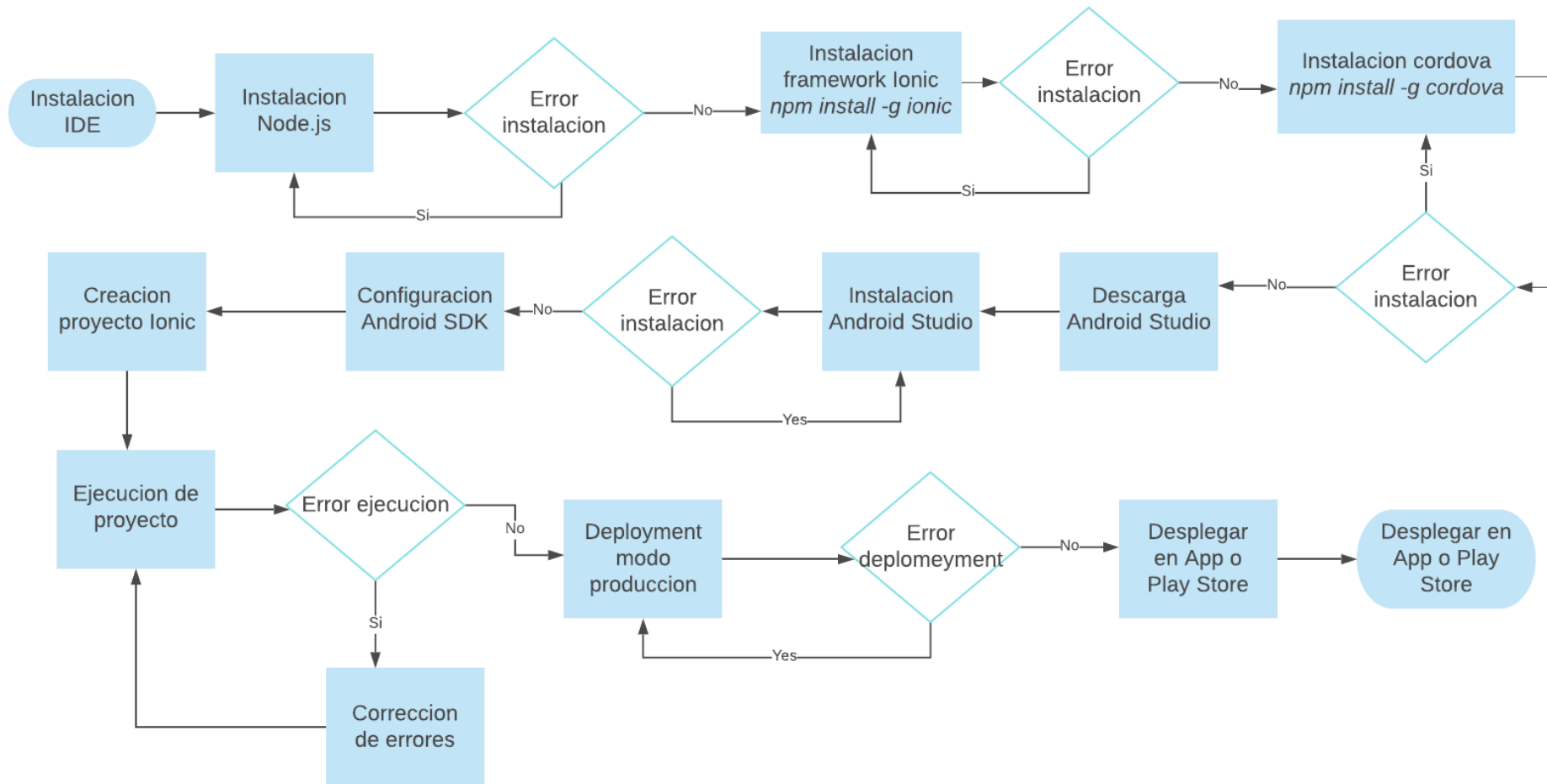


Figura 47.-Proyecto visualización en modo navegador

1.2.3 Proceso de ejecución de una aplicación en ionic



ANEXO 2. INTERFACES DE APLICACIÓN MÓVIL

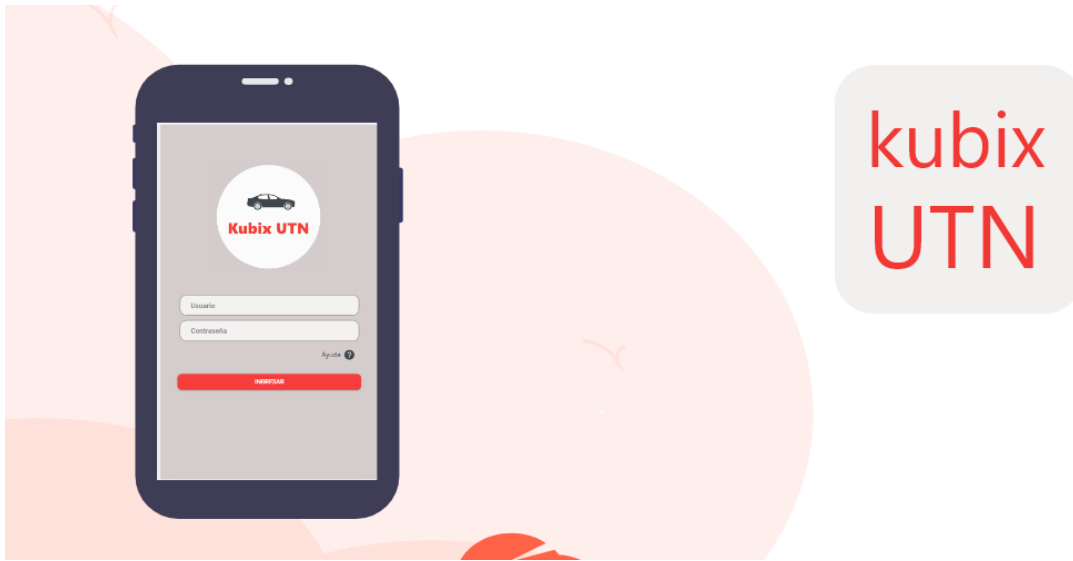


Figura 48.- Carátula para Play Store



Figura 49.- Inicio de sesión / Login

Figura 50.- Home usuario modo pasajero

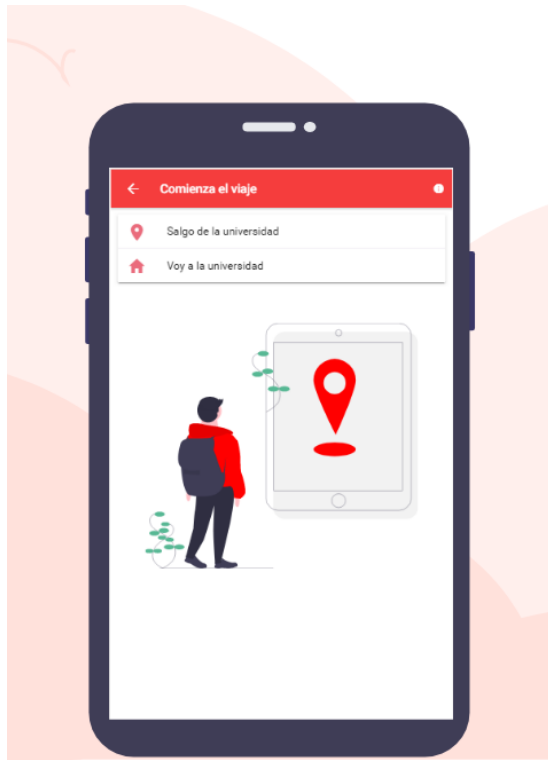


Figura 51.- Universidad origen o destino

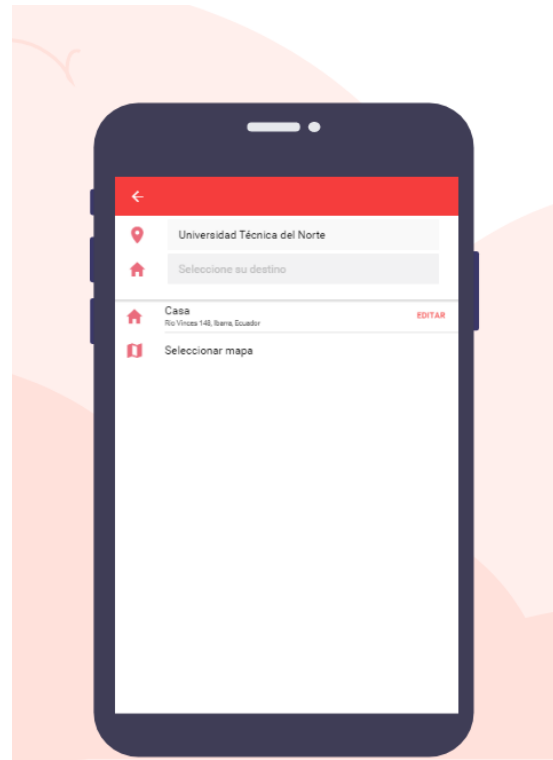


Figura 52.- Seleccionar destino

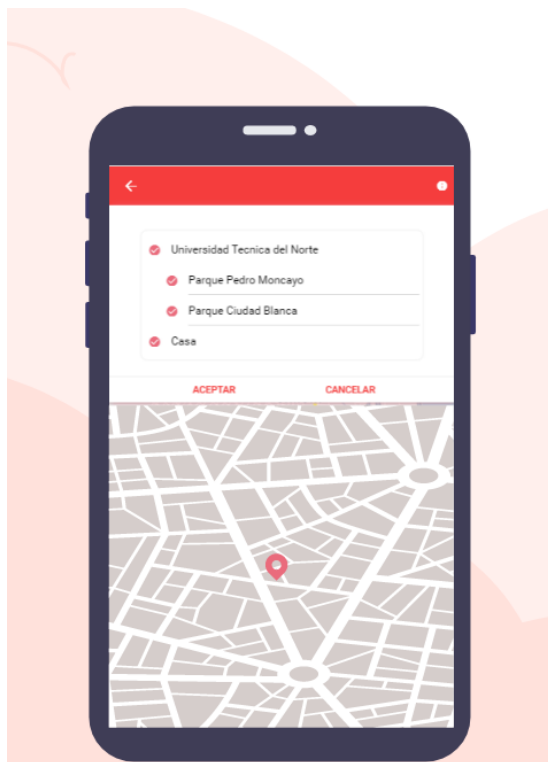


Figura 53.- Seleccionar lugares de paso

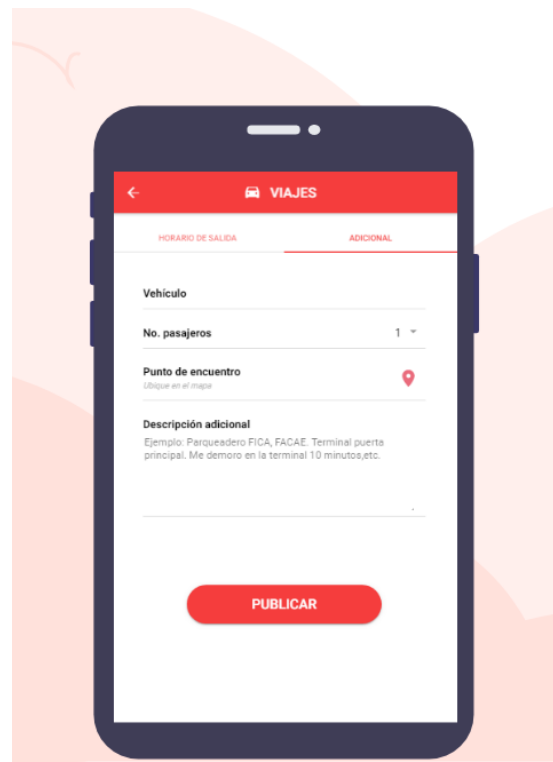


Figura 54.- Detalles adicionales publicar ruta

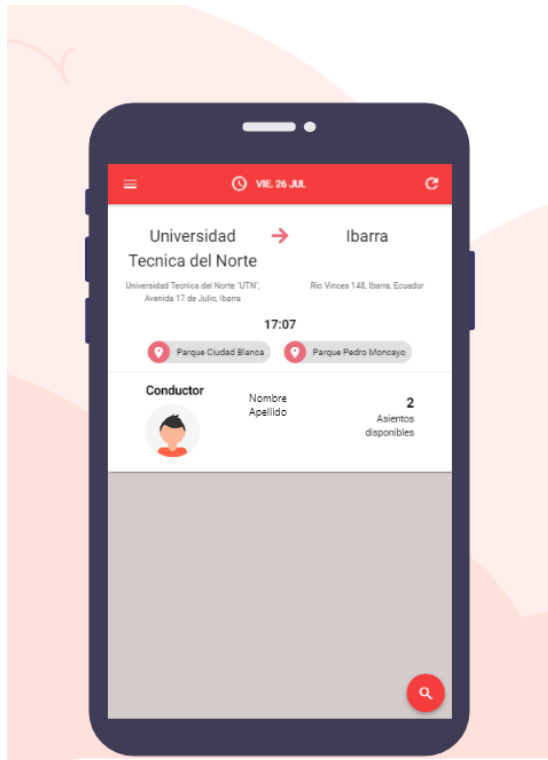


Figura 55.- Viajes disponibles para este día

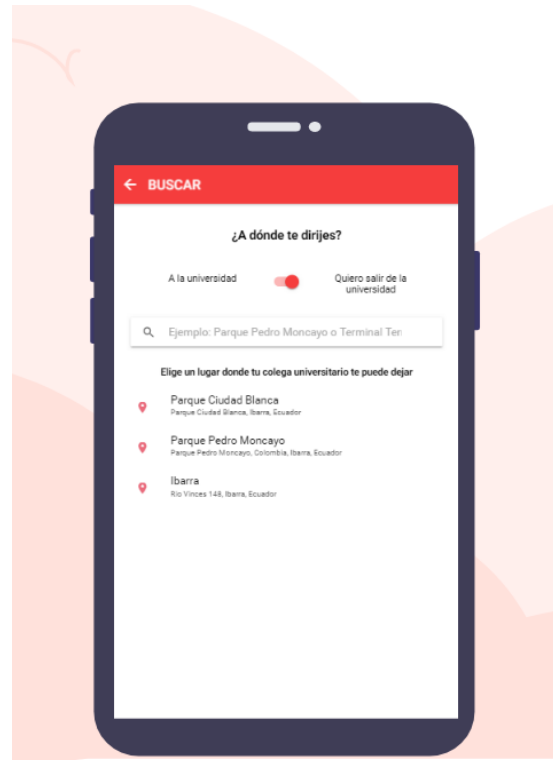


Figura 56.- Buscar viajes por lugares

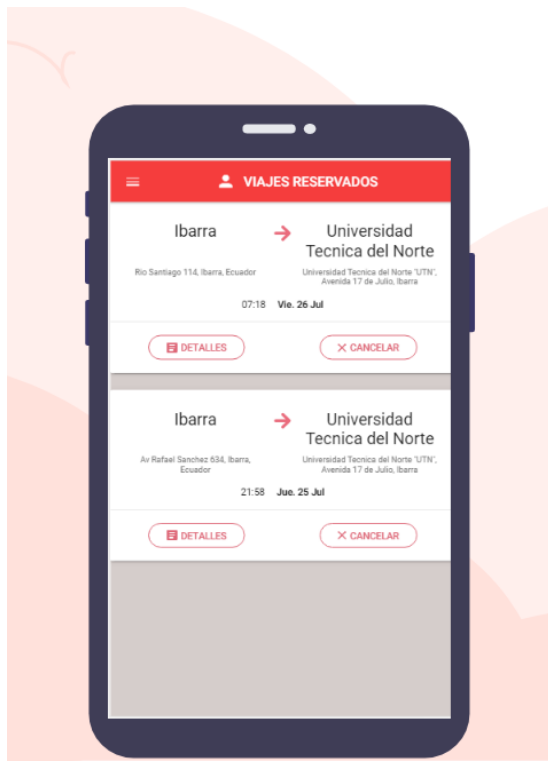


Figura 57.- Viajes reservados usuario pasajero

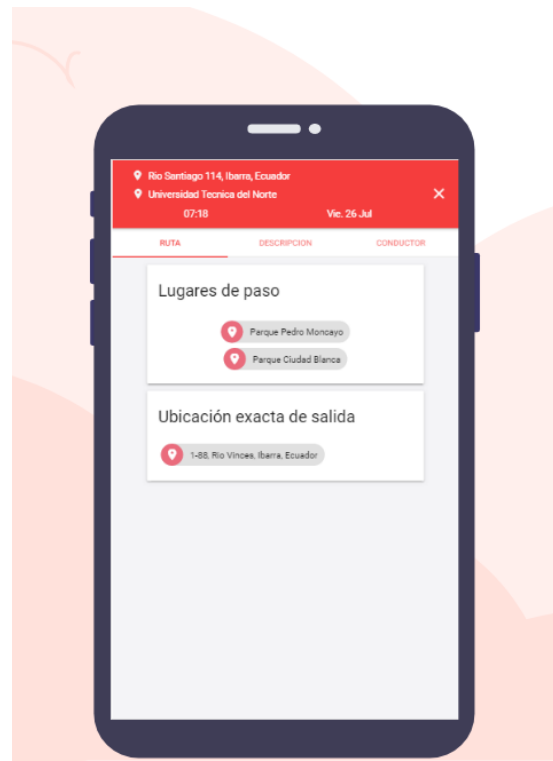


Figura 58.- Detalles de viajes reservados

ANEXO 3. VALIDACIÓN DE RESULTADOS

3.1 INSTALACIÓN HERRAMIENTA APACHE JMETER

Descarga la herramienta Apache JMeter desde la página oficial.

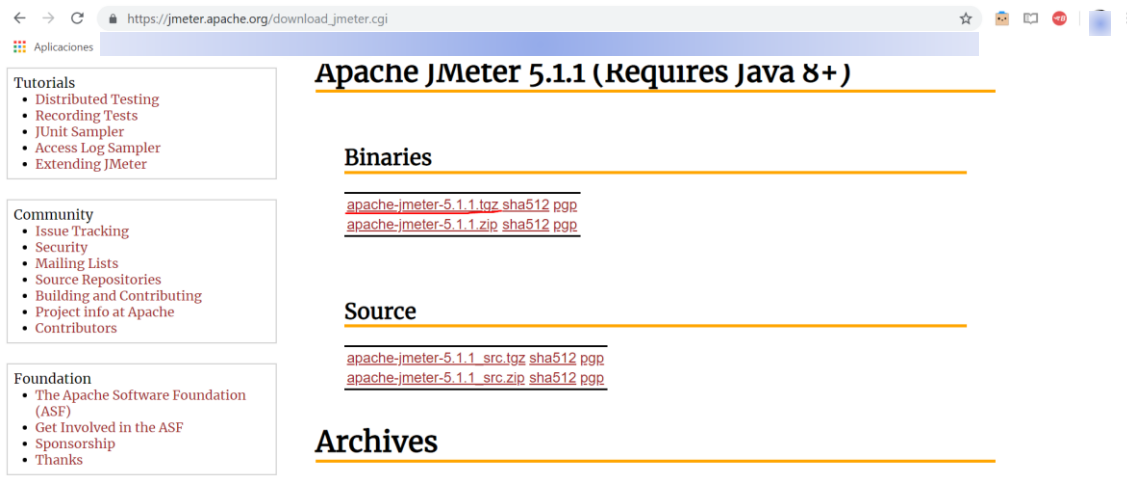


Figura 59.- Descarga de herramienta Apache JMeter

Se descomprime el fichero en cualquier directorio.

Para iniciar el software se ubica en el siguiente directorio `[ubicación_carpeta] / bin / jmeter.bar` en Windows y `[ubicación_carpeta]/bin/jmeter.sh` en Linux.

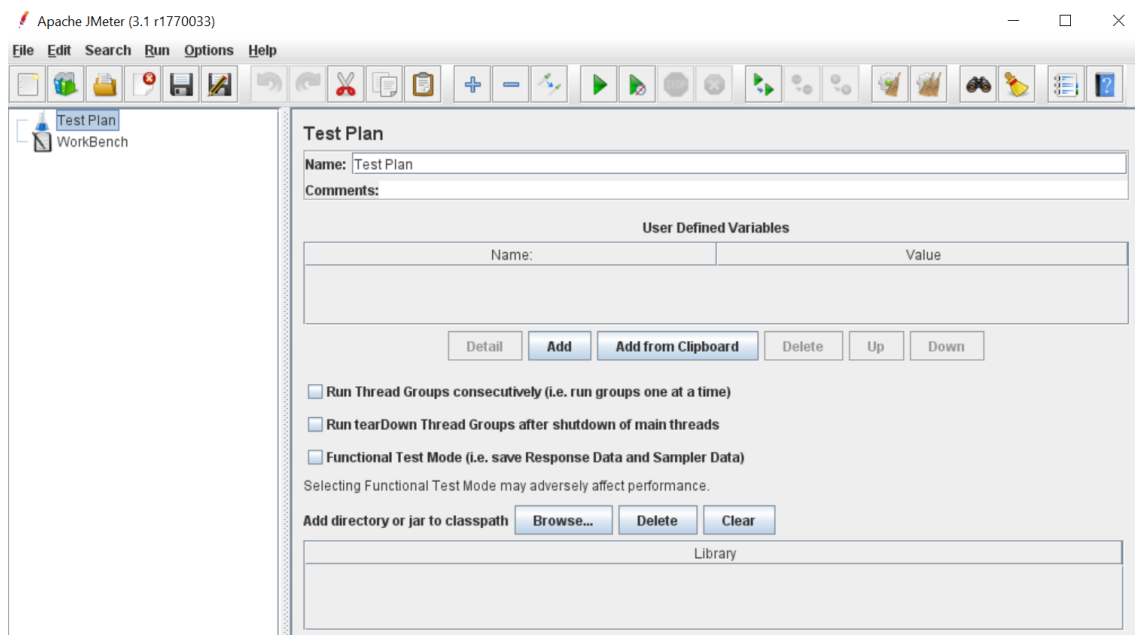


Figura 60.- Ventana inicial de Apache JMeter

3.2 SET UP DE AMBIENTE DE PRUEBA

Clic en el menú *File > templates* se selecciona el template de recording y *créate*.

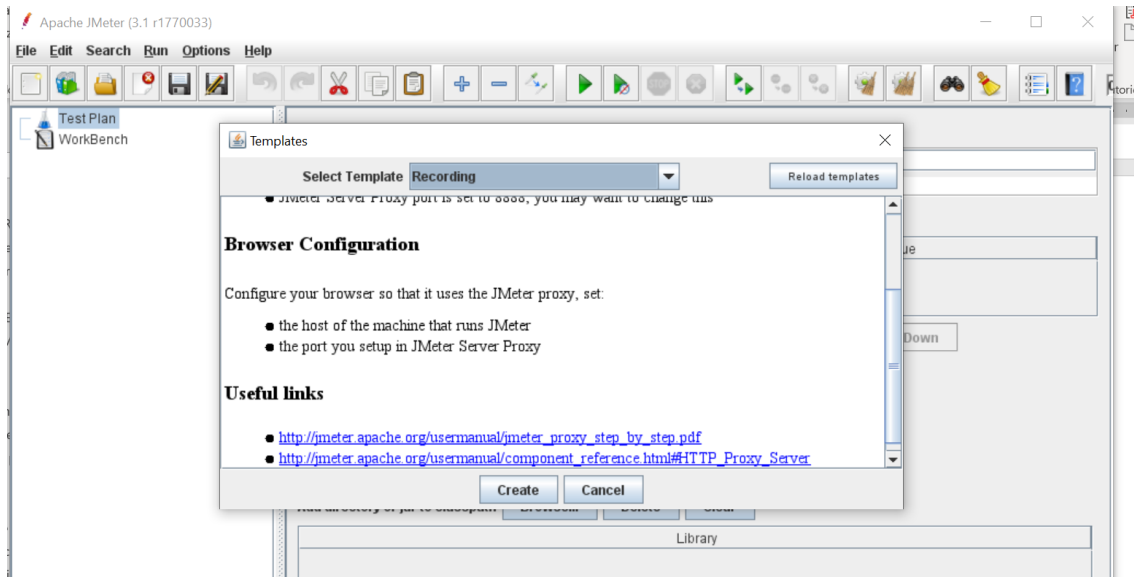


Figura 61.- Creación de template

Una vez establecido el template de tipo *recording*, se debe revisar la IP de nuestra maquina host. La máquina local actuara como servidor de proxy para conexión con el dispositivo móvil. Acceder al teléfono móvil mediante la dirección IP. En la sección de *Thread Group* establecer el número de threads o de acuerdo con la tabla de comparaciones que se ha establecido *ver tabla 32*. Click en el botón de *Start*.

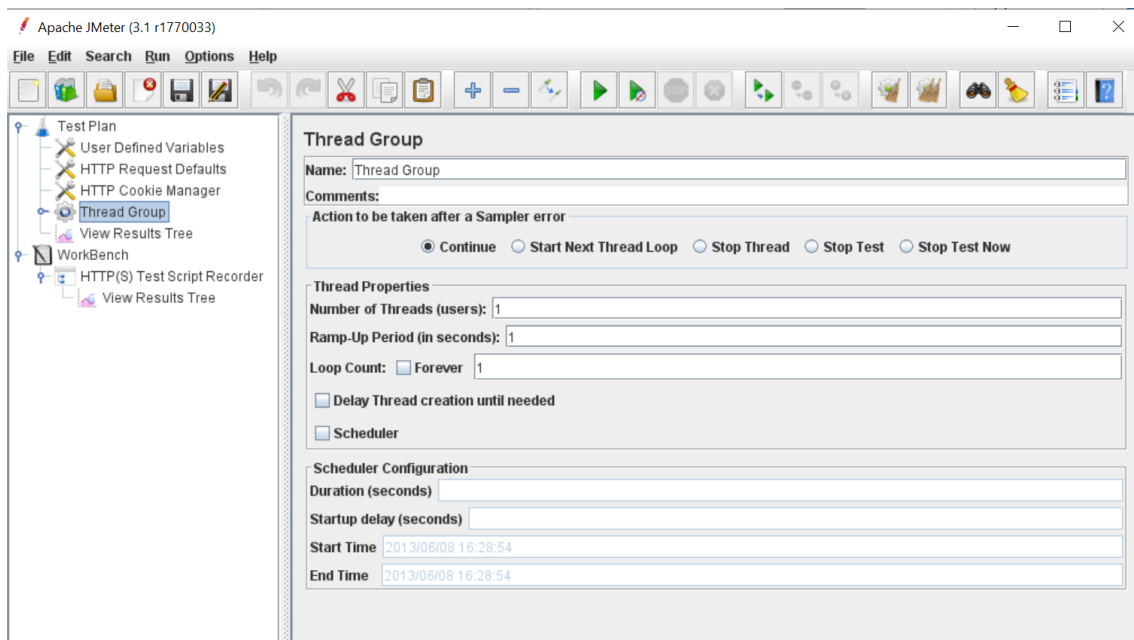


Figura 62.-Thread Group

Realizar el proceso de creación de ruta en el teléfono móvil y las capturas se podrán visualizar en la sección de *Thread Group*. Una vez finalizado clic en botón de *Stop* y guardar el archivo en formato jmx.

Una vez guardado el archivo se procede a realizar la prueba con n número de threads. Al ser un gran número de threads se deber incrementar el pool de conexiones de apache JMeter dentro del mismo archivo de *jmeter.bat* con cualquier editor de texto de su preferencia. En la variable de *HEAP* en la segunda variable se debe estar el numero en MB del tamaño del pool.

```
rem See the unix startup file for the rationale of the following parameters,
rem including some tuning recommendations
set HEAP=-Xms512m -Xmx8192m
set NEW=-XX:NewSize=128m -XX:MaxNewSize=128m
set SURVIVOR=-XX:SurvivorRatio=8 -XX:TargetSurvivorRatio=50%
set TENURING=-XX:MaxTenuringThreshold=2
rem Java 8 remove Permanent generation, don't settings the PermSize
if %current_minor% LEQ "8" (
    rem Increase MaxPermSize if you use a lot of Javascript in your Test Plan :
    set PERM=-XX:PermSize=64m -XX:MaxPermSize=128m
)

set CLASS_UNLOAD=-XX:+CMSClassUnloadingEnabled
rem set DEBUG=-verbose:gc -XX:+PrintTenuringDistribution

rem Always dump on OOM (does not cost anything unless triggered)
set DUMP=-XX:+HeapDumpOnOutOfMemoryError

rem Additional settings that might help improve GUI performance on some platforms
rem See: http://java.sun.com/products/java-media/2D/perf\_graphics.html

set DDRAW=
rem Setting this flag to true turns off DirectDraw usage, which sometimes helps to get rid of a lot of rendering problems on Win32.
rem set DDRAW=%DDRAW% -Dsun.java2d.noddraw=true

rem Setting this flag to false turns off DirectDraw offscreen surfaces acceleration by forcing all createVolatileImage calls to become createImage calls, and di
rem set DDRAW=%DDRAW% -Dsun.java2d.ddoffscreen=false

rem Setting this flag to true enables hardware-accelerated scaling.
rem set DDRAW=%DDRAW% -Dsun.java2d.dscale=true

rem Server mode
rem Collect the settings defined above
```

Figura 63.- Heap de conexión Apache JMeter

Una vez guardado los cambios se deberán realizar el test en modo consola, por lo que en modo GUI ocupa mucho espacio en memoria.

Mediante la siguiente instrucción *jmeter -t C:\Apachejmeter_kubixtest\pub-kubix.jmx -n -l C:\Apachejmeter_kubixtest\results.csv*

Esta instrucción estará enviando a prueba el archivo.jmx que se había guardado con los threads ya establecidos y se guardado en esa ruta en un archivo .csv. Aclarando que este tipo de pruebas se deben ejecutar de manera unitaria con respecto al número de comprobaciones en la tabla 32.

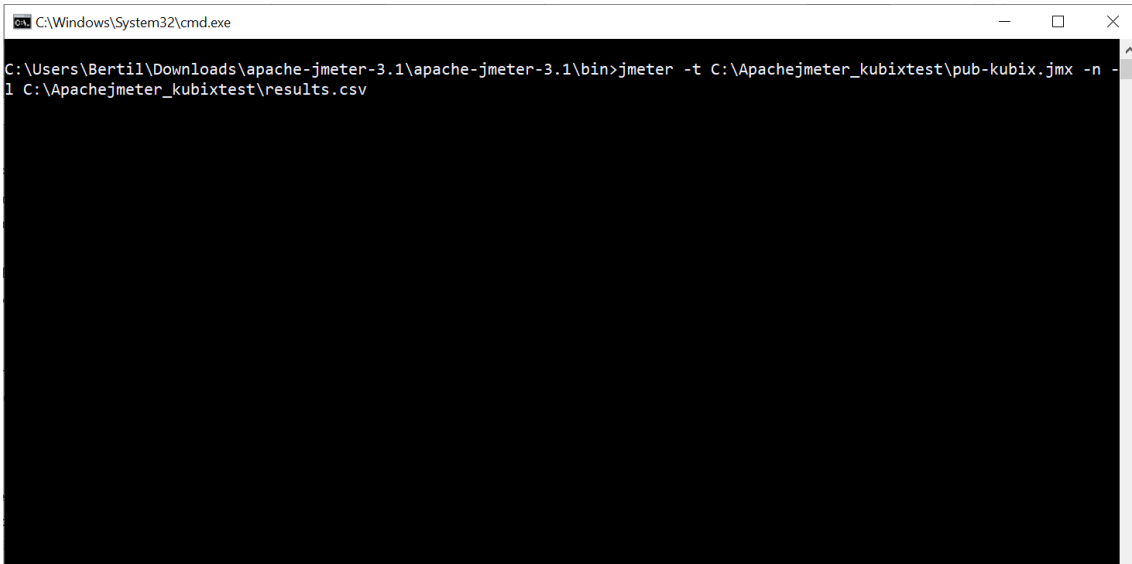


Figura 64.- Ejecución en modo consola

Al final se visualiza un archivo .csv con datos listos para hacer su validación.

timeStamp	elapsed	label	responseCoc	threadName	success	bytes	sentBytes	grpThreads	allThreads	Latency	SampleCoun	ErrorCount	Hostname	IdleTime	Conn
05:24:15	6494	http://localh	Non HTTP re	Thread Grou	false	2204	0	8	8	0	1	1	BERTIL-PC	0	0
05:24:15	7394	http://localh	Non HTTP re	Thread Grou	false	2204	0	8	8	0	1	1	BERTIL-PC	0	0
05:24:21	688	http://cloud:	200 Thread	Grou true		73682	388	9	9	458	1	0	BERTIL-PC	0	0
05:24:21	688	http://cloud:	200 Thread	Grou true		73682	388	9	9	458	1	0	BERTIL-PC	0	0
05:24:22	209	http://cloud:	200 Thread	Grou true		835	377	9	9	209	1	0	BERTIL-PC	0	0
05:24:22	209	http://cloud:	200 Thread	Grou true		835	377	9	9	209	1	0	BERTIL-PC	0	0
05:24:17	9172	http://localh	Non HTTP re	Thread Grou	false	2204	0	9	9	0	1	1	BERTIL-PC	0	0
05:24:17	3748	http://localh	Non HTTP re	Thread Grou	false	811	323	9	9	3748	1	0	BERTIL-PC	0	0
05:24:17	9176	http://localh	Non HTTP re	Thread Grou	false	2204	0	12	12	0	1	1	BERTIL-PC	0	0
05:24:22	3775	http://cloud:	200 Thread	Grou true		811	323	13	13	3775	1	0	BERTIL-PC	0	0
05:24:26	309	http://cloud:	200 Thread	Grou true		1996	377	13	13	309	1	0	BERTIL-PC	0	0
05:24:26	344	http://cloud:	200 Thread	Grou true		1996	377	13	13	344	1	0	BERTIL-PC	0	0
05:24:26	635	http://cloud:	200 Thread	Grou true		73682	388	13	13	415	1	0	BERTIL-PC	0	0
05:24:26	716	http://cloud:	200 Thread	Grou true		73682	388	13	13	417	1	0	BERTIL-PC	0	0
05:24:27	210	http://cloud:	200 Thread	Grou true		835	377	14	14	209	1	0	BERTIL-PC	0	0
05:24:27	174	http://cloud:	200 Thread	Grou true		835	377	14	14	166	1	0	BERTIL-PC	0	0
05:24:27	172	http://cloud:	200 Thread	Grou true		811	323	14	14	172	1	0	BERTIL-PC	0	0
05:24:27	218	http://cloud:	200 Thread	Grou true		811	323	14	14	218	1	0	BERTIL-PC	0	0
05:24:26	958	http://cloud:	200 Thread	Grou true		236504	330	14	14	333	1	0	BERTIL-PC	0	0
05:24:26	919	http://cloud:	200 Thread	Grou true		236504	330	14	14	368	1	0	BERTIL-PC	0	0

Figura 65.- Archivo .csv con resultados de prueba