



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
E INFORMÁTICOS

TEMA:

“APLICACIÓN MÓVIL DE GESTIÓN ADMINISTRATIVA Y ACADÉMICA EN
LA ESCUELA DE FORMACIÓN Y CAPACITACIÓN DE CONDUCTORES
PROFESIONALES AMBATO UTILIZANDO LA PLATAFORMA ANDROID.”

Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero
en Sistemas Computacionales e Informáticos

SUBLÍNEA DE INVESTIGACIÓN:

Aplicaciones para dispositivos móviles

AUTOR: Luis Alberto Marcalla Pilamunga
TUTOR: Ing. Clay Fernando Aldás Flores, Mg.

Ambato - Ecuador
Octubre, 2017

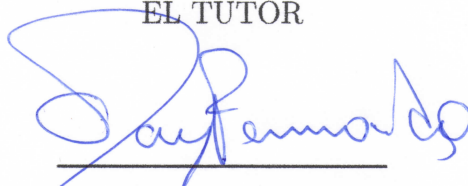
APROBACIÓN DEL TUTOR

En mi calidad de Tutor del Trabajo de Investigación sobre el Tema:

“Aplicación móvil de gestión administrativa y académica en la escuela de formación y capacitación de conductores profesionales Ambato utilizando la plataforma Android.”, del señor Luis Alberto Marcalla Pilamunga, estudiante de la Carrera de Ingeniería en Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, Octubre de 2017

EL TUTOR



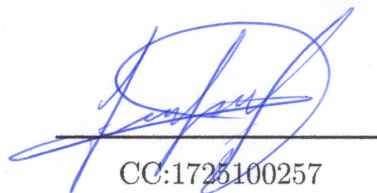
Ing. Clay Fernando Aldás Flores, Mg.

AUTORÍA

El presente trabajo de investigación titulado: **“Aplicación móvil de gestión administrativa y académica en la escuela de formación y capacitación de conductores profesionales Ambato utilizando la plataforma Android.”**. Es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, Octubre de 2017

Luis Alberto Marcalla Pilamunga



CC:1725100257

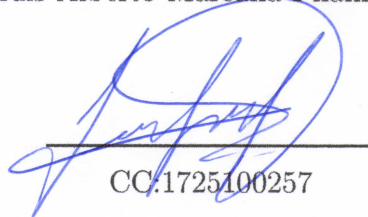
DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, Octubre de 2017

Luis Alberto Marcalla Pilamunga

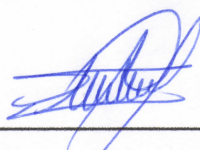


CC:1725100257

APROBACIÓN COMISIÓN CALIFICADORES

La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. David Guevara e Ing. Félix Fernández, revisó y aprobó el Informe Final del trabajo de graduación titulado **“Aplicación móvil de gestión administrativa y académica en la escuela de formación y capacitación de conductores profesionales Ambato utilizando la plataforma Android.”**, presentado por el señor Luis Alberto Marcalla Pilamunga de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ing. Pilar Urrutia, Mg.



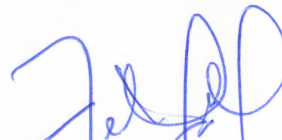
PRESIDENTE DEL TRIBUNAL

Ing. David Guevara, Mg.



DOCENTE CALIFICADOR

Ing. Félix Fernández, Ph. D.



DOCENTE CALIFICADOR

DEDICATORIA

Dedico la realización de la presente tesis a mis padres Abel y Elena quienes me han sabido inculcar valores y principios importantes para poder afrontar los obstáculos presentes en mi vida, sin su constante preocupación y apoyo incondicional este trabajo no hubiera sido posible su culminación.

Janneth y Jennifer, mis hermanas las cuales han estado siempre presente cuando las he necesitado. Su compañía en mi camino a servido de inspiración para lograr mis metas.

A mis queridas sobrinas Gisela y Emily, las más pequeñas del hogar.

Marcalla Pilamunga Luis Alberto

AGRADECIMIENTO

A Dios que me ha dado las fuerzas para no doblegarme ante las adversidades y coraje para avanzar firme en las dificultades que se me han presentado en el transcurso de mi vida.

A mi tutor Ing. Clay Aldás que con su vasto conocimiento y experiencia fue un pilar importante para la culminación del presente trabajo.

A mis compañeros de carrera que me hicieron compañía durante este largo trayecto, con su presencia y apoyo me ayudaron a alcanzar mi meta propuesta.

Marcalla Pilamunga Luis Alberto

ÍNDICE

APROBACIÓN DEL TUTOR	ii
AUTORÍA	iii
AUTORÍA	iv
APROBACIÓN COMISIÓN CALIFICADORA	v
Dedicatoria	vi
Agradecimiento	vii
Introducción	xx
CAPÍTULO 1 El problema	1
1.1 Tema de investigación	1
1.2 Planteamiento del problema	1
1.3 Delimitación	2
1.4 Justificación	3
1.5 Objetivos	4
1.5.1 General	4
1.5.2 Específicos	4
CAPÍTULO 2 Marco teórico	5
2.1 Antecedentes investigativos	5
2.2 Fundamentación teórica	6
2.2.1 Dispositivos móviles	6
2.2.2 Sistemas operativos móviles	7
2.2.2.1 Android	7
2.2.2.2 iOS	8
2.2.2.3 Windows Phone	9
2.2.2.4 Semejanzas y diferencias	9
2.2.3 Sistema operativo Android	10

2.2.4	Arquitectura de Android	11
2.2.5	Diseño adaptativo	13
2.2.5.1	Bootstrap	15
2.2.6	Patrones de diseño de software	17
2.2.6.1	MVC	18
2.2.7	Aplicaciones web	19
2.2.8	Herramientas Microsoft	20
2.2.8.1	C#	20
2.2.8.2	ASP.NET	21
2.2.8.3	SQL Server	23
2.2.9	Escuelas de capacitación para conductores profesionales	23
2.3	Propuesta de solución	25
CAPÍTULO 3 Metodología		26
3.1	Modalidad básica de la investigación	26
3.1.1	Investigación de campo	26
3.1.2	Investigación bibliográfica - documental	26
3.1.3	Investigación aplicada	26
3.2	Recolección de información	26
3.3	Procesamiento y análisis de datos	27
3.4	Desarrollo del proyecto	27
3.4.1	Metodología	27
3.4.2	Diferencias entre metodologías tradicionales y ágiles	28
3.4.3	Metodología XP (Extreme Programming)	28
3.4.3.1	Definición	29
3.4.3.2	Valores	29
3.4.3.3	Las historias de usuario	30
3.4.3.4	El proceso XP	30
CAPÍTULO 4 Desarrollo de la propuesta		33
4.1	Datos informativos	33
4.2	Fase I - Exploración	34
4.2.1	Historias de usuario	35
4.2.1.1	Aplicación web (Administración)	35
4.2.1.2	Aplicación móvil	38
4.2.2	IDE's de desarrollo	41
4.2.3	Diseño de aplicación móvil	42
4.2.4	Diseño de aplicación web	46

4.3	Fase II: Planificación de la entrega	49
4.4	Fase III: Iteraciones	50
4.4.1	Iteración primera	50
4.4.2	Iteración segunda	61
4.4.3	Iteración tercera	74
4.5	Fase IV: Producción	82
4.5.1	Pruebas de aceptación	82
4.5.1.1	Aplicación web (Administración)	82
4.5.1.2	Aplicación móvil	85
4.5.2	Implantación	87
4.6	Fase V: Mantenimiento	88
4.7	Fase VI: Muerte del proyecto	89
	CAPÍTULO 5 Conclusiones y recomendaciones	90
5.1	CONCLUSIONES	90
5.1.1	RECOMENDACIONES	91
	Bibliografía	92
	ANEXOS	97

ÍNDICE DE FIGURAS

2.1	Arquitectura de Android	11
2.2	Diseño adaptable en distintos dispositivos	14
2.3	Diagrama de flujo - MVC	18
4.1	Diseño móvil - Acceso a la aplicación móvil	43
4.2	Diseño móvil - Módulo escuela	43
4.3	Diseño móvil - Módulo perfil académico	44
4.4	Diseño móvil - Módulo académico	44
4.5	Diseño móvil- Módulo administrativo	45
4.6	Diseño móvil - Módulo eventos	45
4.7	Bibliotecas de Bootstrap	46
4.8	Diseño de aplicación web	48
4.9	Diagrama Entidad-Relacion de base de datos	55
4.10	Login de aplicación web - Vista en PC	56
4.11	Home de aplicación web - Vista en PC	58
4.12	Home de aplicación web - Vista en smartphone	58
4.13	Directorio de controllers - Aplicación web	61
4.14	Web API autenticación - Postman	68
4.15	Diseño de login - Aplicación móvil	70
4.16	Acceso a la aplicación - Aplicación móvil	73
4.17	Ver datos personales - Aplicación móvil	73
4.18	Proyecto escuela en android studio	80
4.19	Módulo académico - Aplicación móvil	80
4.20	Módulo administrativo - Aplicación móvil	81
4.21	Módulo eventos - Aplicación móvil	81
4.22	Prueba 1 - Acceso al sistema web	82
4.23	Prueba 2 - Mantenimiento de tablas	83
4.24	Prueba 3 - Ingreso de notas	83
4.25	Prueba 4 - Envío de notificaciones	84
4.26	Prueba 5 - Desarrollado de Web API	84
4.27	Prueba 6 - Acceso a la aplicación móvil	85

4.28 Prueba 7 - Implementación de menú	85
4.29 Prueba 8 - Información de escuela	86
4.30 Prueba 9 - Ver datos personales	86
4.31 Creación de firma - Firma de APK	87
4.32 Ejecución de firma - Firma de APK	88
4.33 Verificación de firma - Firma de APK	88
A.1 Instalación - Aplicación móvil	102
A.2 Login - Aplicación móvil	103
A.3 Escuela - Aplicación móvil	103
A.4 Perfil académico - Aplicación móvil	104
A.5 Académico - Aplicación móvil	105
A.6 Notificación recibida - Aplicación móvil	105
A.7 Lista de notificaciones - Aplicación móvil	106
A.8 Eventos - Aplicación móvil	107
B.1 Login - Aplicación web	109
B.2 Dropdown - Aplicación web	109
B.3 Lista de categorías de licencias - Aplicación web	110
B.4 Edición de categorías de licencias - Aplicación web	110
B.5 Horarios de clases - Aplicación web	111
B.6 Nuevo horario de clases - Aplicación web	111
B.7 Búsqueda por paralelo - Aplicación web	112
B.8 Estudiantes por paralelo - Aplicación web	112
B.9 Notificaciones de estudiante - Aplicación web	113
B.10 Nueva notificación de estudiante - Aplicación web	113
B.11 Notificación ingresada de estudiante - Aplicación web	113
B.12 Paralelos de docente - Aplicación web	114
B.13 Listado de calificaciones - Aplicación web	115
B.14 Actualización de calificaciones - Aplicación web	115
B.15 Informe de calificaciones 1 - Aplicación web	116
B.16 Informe de calificaciones 2 - Aplicación web	116

ÍNDICE DE TABLAS

2.1	Métodos HTTP de REST	22
3.1	Metodologías tradicionales y ágiles	28
4.1	Historia de usuario 1 - Acceso al sistema web	35
4.2	Historia de usuario 2 - Mantenimiento de tablas	36
4.3	Historia de usuario 3 - Ingreso de notas	36
4.4	Historia de usuario 4 - Envío de notificaciones	37
4.5	Historia de usuario 5 - Desarrollo de Web API	37
4.6	Historia de usuario 6 - Acceso a la aplicación	38
4.7	Historia de usuario 7 - Implementación de menú	38
4.8	Historia de usuario 8 - Escuela	39
4.9	Historia de usuario 9 - Perfil académico	39
4.10	Historia de usuario 10 - Académico	40
4.11	Historia de usuario 11 - Administrativo	40
4.12	Historia de usuario 12 - Eventos	41
4.13	Estimación del tiempo de desarrollo	49
4.14	Velocidad de proyecto	50
4.15	Actividad 1 - Historia 1 - Acceso a la administración web	51
4.16	Actividad 2 - Historia 1 - Acceso a la administración web	51
4.17	Actividad 3 - Historia 1 - Acceso a la administración web	52
4.18	Actividad 4 - Historia 1 - Acceso a la administración web	52
4.19	Actividad 1 - Historia 2 - Mantenimiento de datos	53
4.20	Actividad 2 - Historia 2 - Mantenimiento de datos	53
4.21	Actividad 1 - Historia 3 - Ingreso de notas	54
4.22	Actividad 1 - Historia 4 - Envío de notificaciones	54
4.23	Actividad 1 - Historia 5 - Desarrollo de Web API's	62
4.24	Actividad 2 - Historia 5 - Desarrollo de Web API's	62
4.25	Actividad 3 - Historia 5 - Desarrollo de Web API's	63
4.26	Actividad 4 - Historia 5 - Desarrollo de Web API's	63
4.27	Actividad 1 - Historia 6 - Desarrollo de Web API's	64

4.28	Actividad 2 - Historia 6 - Desarrollo de Web API's	64
4.29	Actividad 3 - Historia 6 - Desarrollo de Web API's	65
4.30	Actividad 4 - Historia 6 - Desarrollo de Web API's	65
4.31	Actividad 1 - Historia 7 - Implementación de menú	66
4.32	Actividad 2 - Historia 7 - Implementación de menú	66
4.33	Actividad 1 - Historia 8 - Información de escuela	67
4.34	Actividad 1 - Historia 9 - Ver datos personales	74
4.35	Actividad 2 - Historia 9 - Ver datos personales	74
4.36	Actividad 1 - Historia 10 - Ver historial académico	75
4.37	Actividad 2 - Historia 10 - Ver historial académico	75
4.38	Actividad 1 - Historia 11 - Ver notificaciones	76
4.39	Actividad 2 - Historia 11 - Ver notificaciones	76
4.40	Actividad 1 - Historia 12 - Ver evento	77
4.41	Actividad 2 - Historia 12 - Ver evento	77

Resumen

El uso de las tecnologías de la comunicación y de la información han ido creciendo a un ritmo apresurado, esto gracias a la aparición de nuevos equipos compactos de gran velocidad de procesamiento y de nuevos lenguajes de programación de alto nivel que conjuntamente facilitan a las personas en sus actividades del diario vivir. Esto obliga a mantenerse a la vanguardia de las nuevas tecnologías que marcan tendencia en nuestro mundo globalizado, la aparición de nuevos paradigmas es un hecho a tomar en cuenta para no quedar desfasados.

Teniendo esto en cuenta se planteó implantar una aplicación móvil bajo la plataforma Android en una institución dedicada a la instrucción de conductores profesionales de la ciudad de Ambato, que hasta hace un tiempo el uso de las tecnologías de información y de la comunicación era poco notorio y no se le sacaba el máximo provecho en el transcurso normal de los procesos académicos y administrativos, lo que ocasionaba el desperdicio de recursos valiosos para la institución. El presente proyecto plantea desarrollar e implantar una aplicación móvil para dispositivos con sistema operativo Android, esta aplicación consta con un panel de administración web que sirve para la alimentación de información de todo el conjunto como son datos de los estudiantes, administrativos y docentes. Ambas partes mantienen una comunicación constante por medio de web services basados en REST (Representational State Transfer) o Web API's (Application Programming Interface).

El panel de administración se trata de una aplicación web desarrollada en el framework de Microsoft .NET, específicamente una aplicación ASP.NET que fue desarrollado en Visual Studio. La aplicación ASP.NET además de ser responsive por el uso de estilos CSS con Bootstrap contiene los Web API's para la comunicación con la aplicación móvil (Android). Para todo lo que es desarrollo de la aplicación web, se uso el patrón de diseño de software MVC (Modelo Vista Controlador) por las ventajas que su implementación tiene.

La aplicación móvil fue desarrollada bajo la plataforma de Android usando el paradigma de diseño implementado por Google desde 2014 que es Material Design para una mejor experiencia de usuario, la aplicación móvil posee componentes propios de Material Design como son una Barra de Botones para la navegación, el uso de Fragmentos para los distintos módulos o pantallas, CardViews que son

tarjetas de presentación elegantes y efectos de transiciones entre pantallas para darle animación. Para hacer las petición a los Web API's alojados en el hosting la aplicación móvil hace uso de librerías de Retrofit que es un cliente de servicios web basados en REST para Java.

Respecto a los procesos administrativos, el personal de la institución puede acceder a la aplicación ASP.NET cuando necesite enviar un mensaje a un estudiante sobre su estado legal en la institución como son faltas injustificadas, pensiones no canceladas, anuncios de interés para la institución, estados de legalización de los títulos de conducción profesional, etc. que luego son recibidas de inmediato al estudiante por medio de la aplicación Android desde un smartphone. En lo relacionado a lo académico, los docentes tienen acceso a la aplicación ASP.NET donde pueden actualizar el historial académico de los paralelos a su cargo, al igual que los mensajes de los administrativos estas actualizaciones de notas pueden ser consultado desde la aplicación móvil de manera inmediata. El docente también puede imprimir un informe de calificaciones para dejarla posteriormente en Secretaría General.

Abstract

The use of communication and information technologies has been creating a favorable step, thanks to the appearance of new compact equipment of high speed of processing and appearance of new languages of programming of high level that jointly facilitate the activities of daily living. This forces us to remain at the forefront of new technologies that mark the trend in our globalized world, the emergence of new paradigms is a fact to take into account in order not to be outdated.

With this in mind, it was proposed to implement a mobile application under the Android platform in an institution dedicated to the instruction of professional drivers of the city of Ambato that until some time the use of information and communication technologies was not very noticeable and not The maximum benefit was obtained in the normal course of the academic and administrative processes causing waste of valuable resources for the Institution. This project proposes to develop and implement a mobile application for Android devices, this application consists of a web administration panel that serves to feed information of the whole as data of students, administrative and teaching. Both parties maintain constant communication through web services based on REST (Representational State Transfer) or Web API's (Application Programming Interface).

The administration panel is a web application developed in the framework of Microsoft .NET, specifically an ASP.NET application that was developed in Visual Studio. The ASP.NET application in addition to being responsive to the use of CSS styles with Bootstrap contains the Web APIs for communication with the mobile application (Android). For all that is web application development, the MVC (Model Vista Controller) software design pattern is used because of the advantages that its implementation has.

The mobile application was developed under the platform of Android using the paradigm of design implemented by Google since 2014 that is Material Design for a better user appearance, the mobile application has its own components of Material Design such as a button bar for navigation, The use of Fragments for different modules or screens, CardViews that are elegant business cards and effects transitions between screens to give you animation. To make requests to the Web APIs hosted in the hosting the mobile application makes use of Retrofit libraries that is a REST based web services client for Java.

Regarding administrative processes, the staff of the Institution can access the ASP.NET application when they need to send a message to a student about their legal status in the institution such as unjustified absences, pensions not canceled, announcements of interest to the institution, States of legalization of professional driving titles, etc. Which are then immediately received to the student via the Android application from a smartphone. As far as academics are concerned, teachers have access to the ASP.NET application where they can update the academic history of the parallels at their expense, as well as the administrative messages these note updates can be viewed from the mobile application.

Glosario de términos

Algoritmo

Es un conjunto prescrito de instrucciones o reglas bien definidas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos que no generen dudas a quien deba realizar dicha actividad. Dados un estado inicial y una entrada, siguiendo los pasos sucesivos se llega a un estado final y se obtiene una solución.

API

La interfaz de programación de aplicaciones (API) es el conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción. Son usadas generalmente en las bibliotecas.

APK

Se refiere a la extensión que tiene el fichero compacto que contiene todo el código de la aplicación.

IDE

Es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica.

JSON

Acrónimo de JavaScript Object Notation, es un formato de texto ligero para el intercambio de datos.

Objeto

Es una unidad dentro de un programa de computadora que consta de un estado y de un comportamiento, que a su vez constan respectivamente de datos almacenados y de tareas realizables durante el tiempo de ejecución.

Retrofit

Es un cliente REST para Android y Java, desarrollada por Square, muy simple y fácil de aprender. Permite hacer peticiones GET, POST, PUT, PATCH, DELETE y HEAD

INTRODUCCIÓN

El presente Trabajo Estructurado de Manera Independiente denominado: “APLICACIÓN MÓVIL DE GESTIÓN ADMINISTRATIVA Y ACADÉMICA EN LA ESCUELA DE FORMACIÓN Y CAPACITACIÓN DE CONDUCTORES PROFESIONALES AMBATO UTILIZANDO LA PLATAFORMA ANDROID.”; para el entendimiento del mismo, se lo ha dividido en los siguientes capítulos:

CAPÍTULO I denominado “EL PROBLEMA”, identifica el problema que actualmente se puede observar en la realización de los procesos académicos y administrativo inherentes en la escuela de conducción. Además se detalla la justificación y los objetivos que se van a cumplir en la presente investigación.

CAPÍTULO II denominado “MARCO TEÓRICO”, muestra las investigaciones previas que sirven de soporte para el desarrollo de la investigación, además la información de estudios similares anteriormente realizados, así como los aspectos conceptuales que sustentan el tema en general, y el conjunto de conceptos y fundamentos teóricos que han sido analizados en base al problema establecido.

CAPÍTULO III denominado “METODOLOGÍA”, define el tipo de investigación que ha sido desarrollada, el tratamiento de los procesos que señala la modalidad de investigación, asimismo se presenta el tipo de análisis de los datos según el tipo de investigación.

CAPÍTULO IV denominado “DESARROLLO DE LA PROPUESTA”, determina una extensa información sobre el tema investigado en donde se detalla conceptos, métodos y la forma de programar, además se muestra paso a paso el desarrollo de la aplicación y las etapas que se cumplieron de acuerdo a metodología seleccionada.

CAPÍTULO V denominado “CONCLUSIONES Y RECOMENDACIONES”, expone de forma clara y concisa las consideraciones mas relevantes que se han obtenido al finalizar el proyecto, además se indican recomendaciones para futuras adecuaciones.

CAPÍTULO 1

El problema

1.1. Tema de investigación

“Aplicación móvil de gestión administrativa y académica en la Escuela de formación y capacitación de conductores profesionales Ambato utilizando la plataforma Android.”

1.2. Planteamiento del problema

Actualmente las ventajas que proporciona la tecnología ha servido de gran utilidad en nuestra vida cotidiana, a nivel mundial la tecnología juega un papel importante al minimizar las barreras de ubicación y tiempo haciendo que las comunicaciones estén disponibles en todo momento así las empresas están siempre comunicados con sus clientes y proveedores, ser una empresa competitiva es uno de los resultados de este hecho. Por tal razón es común que diversas cadenas empresariales, corporativas e incluso establecimientos educativos, estén buscando día a día soluciones innovadoras que les permita estar a la altura de los retos y demandas con los que se enfrenta en el mundo comercial.

En Ecuador un gran número de empresas u organismos públicos y privados se han visto en la gran necesidad de ofrecer servicios desde el móvil de sus clientes con tan solo necesitar conexión a internet, este método de proveer servicios se ha vuelto una herramienta poderosa para competir en el mercado por hoy en día. Los usuarios prefieren realizar sus trámites, solicitudes, consultas sin la necesidad de trasladarse al sitio en cuestión.

En la provincia de Tungurahua muchas instituciones educativas públicas y privadas cuentan con sistemas informáticos para administrar procesos escolares lo que ha aumentando la eficiencia de las mismas, el uso de las tecnologías en este ámbito

ha permitido a las instituciones educativas ofrecer servicios de calidad e inmediatos a los estudiantes y padres de familia. Las escuelas de capacitación de conductores profesionales también forman parte de este cúmulo de organismos que se dedican a la educación y capacitación de los ciudadanos.

En la ciudad de Ambato se pueden encontrar diversas escuelas de capacitación de conductores profesionales, la más notable es la del cantón con el mismo nombre, esta institución denominada “Escuela de Formación y Capacitación de Conductores Profesionales Ambato” esta localizada entre la avenida Los Incas y Nazacota Puento. Desde 1920 esta prestigiosa escuela presta servicios de capacitación para la obtención de título de conductor profesional para quienes optan por esta opción como una profesión de vida.

La Escuela de Formación y Capacitación de Conductores Profesionales Ambato observando las ventajas que tiene el uso de las tecnologías en los procesos inherentes en otras instituciones semejantes a ella, no desea quedarse atrás, y por eso es importante el desarrollo del presente proyecto. Con la implantación de una aplicación móvil se pretende reforzar los procesos académicos y administrativos que se desarrollan durante cada ciclo académico, consecuentemente se brindará un mejor servicio a sus seguidores.

1.3. Delimitación

Área Académica: Software

Línea de Investigación: Desarrollo de software.

Sublínea de Investigación: Aplicación para dispositivos móviles.

Delimitación Espacial: La presente investigación se desarrollará en la “Escuela de Formación y Capacitación de Conductores Profesionales Ambato”.

Delimitación Temporal: La presente investigación se desarrollara durante seis meses a partir de la aprobación del perfil de tesis.

1.4. Justificación

La tecnología móvil avanza a pasos agigantados y con ella la necesidad de estar actualizados en el uso de herramientas que nos permitan agilizar procesos, optimizar tiempo y recursos, así como realizar tareas sin depender del lugar de ubicación, siendo de mucho interés el desarrollo de aplicaciones móviles.

La gestión administrativa consiste en brindar un soporte administrativo a los procesos organizacionales de las diferentes áreas funcionales de una entidad, a fin de lograr resultados efectivos y con una gran ventaja competitiva revelada en sus objetivos alcanzados. Por el motivo de que representa un pilar fundamental para las organizaciones, el desarrollo del presente proyecto apoyara, coordinará y facilitará las comunicaciones necesarias para la consecución de los procesos administrativos.

La institución educativa para conductores profesionales no es muy diferente a los centros educativos de nivel secundario ya que cuenta con una malla curricular que deben cumplir los solicitantes de una licencia profesional, inspector general, profesores especializados para cada módulo; lo que la diferencia del común centro educativo es que esta posee instructores encargados de dar las clases prácticas (conducción) a horas ya fijadas. Las interacciones entre los distintos actores ya mencionados anteriormente se hacen obligatorias durante un ciclo académico como por ejemplo disposiciones de secretaria que los estudiantes deben acatar, comunicaciones del inspector, horas determinadas por los instructores para dar sus clases, todas estas actividades necesitan ser coordinadas eficazmente entre el personal administrativo y estudiantes para lo cual la aplicación dará todas las facilidades para tal objetivo con el uso de herramientas visuales y notificaciones que los smartphone y sistemas Android soportan.

La poca utilización de herramientas informáticas para la publicación de notas es notorio dentro de la institución, teniendo que obligatoriamente asistir al lugar para la consulta de notas, desperdiciando valiosos recursos, con la implementación de una aplicación móvil la consulta de notas se realizará dentro y fuera del plantel de una manera cómoda, aumentando así la eficiencia del personal al reducir tiempo en sus labores.

La realización de este trabajo muestra al personal administrativo y a los estudiantes las notables ventajas que tienen el uso de las tecnologías si las orientamos correctamente. La sofisticada tecnología de los smartphone provee a la aplicación de una

interfaz amigable, y fácil de manejar lo cual ocasiona el acercando de las personas con las TIC's (Tecnologías de Información y de la Comunicación), a la vez que muestra las obvias ventajas que tiene estar a la vanguardia de la tecnología y ser partícipe de ella.

No se ha visto hasta el momento ninguna escuela de capacitación para conductores profesionales de la provincia de Tungurahua, que implemente este tipo de tecnología para la administración institucional o académica en beneficio de los estudiantes, sin duda la realización del proyecto impulsara aún más el uso de las TIC's dentro de nuestra sociedad.

1.5. Objetivos

1.5.1. General

- Desarrollar una Aplicación móvil de gestión administrativa y académica en la “Escuela de Formación y Capacitación de Conductores Profesionales Ambato” utilizando la plataforma Android.

1.5.2. Específicos

- Analizar los procesos administrativos inherentes en la institución que permitan desarrollar la aplicación propuesta.
- Utilizar responsive design (diseño adaptativo) para el diseño y construcción del sistema web que permita administrar la aplicación móvil.
- Implantar la aplicación móvil de gestión administrativa y académica en la institución.

CAPÍTULO 2

Marco teórico

2.1. Antecedentes investigativos

En los repositorios de la Universidad Técnica de Ambato y de la Universidad Politécnica Salesiana se puede encontrar proyectos sobre aplicaciones móviles parecidos al propuesto, estos trabajos fueron desarrollados por sus estudiantes previo a la obtención del título de tercer nivel. Los antecedentes hallados servirán para recolectar información que beneficiará al avance del presente trabajo, a continuación los trabajos investigativos con una brebe explicación de su contenido.

Tipantasig Llanganate Edison Marcelo estudiante de la carrera de Ingeniería en Sistemas Computaciones e Informáticos de la Universidad Técnica de Ambato realizó una aplicación móvil utilizando plataforma Android para mejorar la calidad del servicio de consulta de información de consumo eléctrico de la EEASA en la Empresa Besixplus Cia. Ltda, donde recomienda considerar la utilización de herramientas confiadas por los propios programadores de Android Studio ya que brindan el soporte y documentación necesaria para garantizar un desarrollo exitoso y una aplicación ajustada a estándares [1].

Oscar Efrén Acosta Mayorga estudiante de la carrera de Ingeniería en Sistemas Computaciones e Informáticos de la Universidad Técnica de Ambato realizó una Aplicación móvil bajo la plataforma android para la gestión de calificaciones en la unidad educativa “Augusto Nicolás Martínez”, en su conclusión acerca de las metodologías de desarrollo recalca que Extreme Programming XP es una buena alternativa para desarrollar aplicaciones para dispositivos móviles, ya que brinda la posibilidad de definir o actualizar los requisitos a medida que se avanza en el proyecto en base a las historias de usuario y las pruebas dentro de cada iteración [2].

Johanna Nataly Picón Cajamarca y Sebastián Zhinin Zhinin de la Universidad Politécnica Salesiana realizaron el proyecto Diseño e Implementación de una aplicación móvil para trabajo operativo de los vendedores de la empresa Agrota cia. Ltda. donde afirman que un modelo de negocio basado en dispositivos móviles implica la ejecución de tareas concretas, cortas y rápidas. Android han demostrado ser lo suficientemente estable para soportar una alta transaccionalidad, disponibilidad y concurrencia de usuarios. A nivel empresarial una aplicación móvil facilita la movilidad del negocio, la comunicación con el cliente además de agregar un plus empresarial sobre la forma en la que se ofertan los productos frente a la competencia [3].

Quisi Peralta Diego Fernando realizó el tema “Diseño e Implementación de una aplicación para dispositivos Android en el Marco del Proyecto Pequeñas y Pequeños Científicos de la Universidad Politécnica Salesiana”, concluye una de las principales razones para utilizar la android es su alto nivel de utilización en plataformas celulares y tablets, en consecuencia se permite que se pueda distribuir, modificar, estudiar sin limitaciones ya que es un sistemas completamente libre [4].

2.2. Fundamentación teórica

2.2.1. Dispositivos móviles

Una gran cantidad de dispositivos electrónicos se clasifican actualmente como dispositivos móviles, desde teléfonos hasta tablets, pasando por dispositivos como lectores de RFID (Radio Frequency Identification). Con tanta tecnología clasificada como móvil, puede resultar complicado identificarlos, por tal motivo se listará las características que lo diferencian de otros dispositivos [5]:

- **Movilidad**

Es la cualidad de un dispositivo para ser transportado o movido con frecuencia y facilidad. Por tanto, el concepto de movilidad es una característica básica. Los dispositivos móviles son aquellos que son lo suficientemente pequeños como para ser transportados y utilizados durante su transporte.

- **Tamaño reducido**

Cualidad de un dispositivo móvil de ser fácilmente usado con una o dos manos sin necesidad de ninguna ayuda o soporte externo. El tamaño reducido también permite transportar el dispositivo cómodamente por parte de una persona.

- **Comunicación inalámbrica**

Por comunicación inalámbrica se entiende la capacidad que tiene un dispositivo de enviar o recibir datos sin la necesidad de un enlace cableado. Por lo tanto, un dispositivo inalámbrico es aquel capaz de comunicarse o de acceder a una red sin cables por ejemplo un teléfono móvil o una PDA (Personal Digital Assistant).

- **Interacción con las personas**

Se entiende por interacción el proceso de uso que establece un usuario con un dispositivo. Entre otros factores, en el diseño de la interacción intervienen disciplinas como la usabilidad y la ergonomía.

2.2.2. Sistemas operativos móviles

El sistema operativo es considerado el programa principal y éste es capaz de administrar todos sus recursos para ser utilizados de manera eficiente, cómoda y sin interrupciones, de tal manera que el usuario pueda mantener una comunicación sin problema haciendo uso de los recursos que el hardware le suministra [6]. Existen en la actualidad, tres sistemas operativos que dominan el mercado: Android, iOS y Windows Phone.

2.2.2.1. Android

Android es un sistema operativo con una plataforma abierta para dispositivos móviles adquirido por Google y la Open Handset Alliance, su finalidad es satisfacer la necesidad de los operadores móviles y fabricantes de dispositivos, además de fomentar el desarrollo de aplicaciones, cualidad que ningún otro sistema operativo incluye en sus conceptos (Google, 2010).

Android bajo la definición de Google se considera un “software stack” o una pila de software, ya que está conformada por [6]:

- El sistema operativo, donde todas las funciones se desarrollan.
- El middleware que permite la conexión entre redes
- Las aplicaciones o API's (Application Programming Interface) que constituyen todos los programas que el teléfono puede ejecutar.

Este sistema operativo se torna realmente atractivo por diversas características, entre ellas se encuentran [6]:

- Plataforma totalmente libre basado en Linux que permite desarrollar aplicaciones y/o modificar las ya existentes con lenguaje de Java.
- Es multitasking permitiendo mantener distintas aplicaciones corriendo al mismo tiempo.
- Compatible con una gran variedad de hardware en el mercado (tablets y dispositivos celulares de marcas como: Motorola, Samsung, ZTE, Huawei, Ericsson por nombrar algunas) permitiendo al usuario elegir el dispositivo que mejor se ajusta a sus necesidades.
- Posee un portal llamado Android Market donde se tiene acceso a muchas aplicaciones que pueden ser utilizadas.
- Permite realizar actualizaciones del sistema operativo en línea siempre y cuando el dispositivo soporte los requerimientos del mismo.
- Puede operar soluciones tecnológicas referentes al uso de redes sociales, mensajería instantánea, correo electrónico, modificación y lectura de procesadores de palabras, hojas de cálculo, presentaciones, lectura de formatos pdf, entre otros.
- Se puede conseguir mucha información a través de documentos web o libros.
- Como característica importante, cuenta con el gran apoyo y la capacidad.

2.2.2.2. iOS

iOS que en un principio fue denominado iPhone OS (system operative), éste es un sistema operativo móvil que pertenece a la empresa Apple Inc. es la base del iPhone, el iPad y el iPod touch. Incluye una colección de apps que te permiten hacer las pequeñas y grandes cosas de todos los días de manera intuitiva y divertida. Cabe mencionar que la existencia de este sistema fue revelada por la empresa Apple en enero de 2007, pero su lanzamiento oficial fue hasta junio del mismo año. iOS siempre se ha caracterizado por su calidad y originalidad, y desde sus inicios siempre ha mantenido determinadas características como por ejemplo su interfaz elegante e intuitiva que brinda a sus usuarios la capacidad de explotar al máximo sus dispositivos, haciendo uso de la serie de aplicaciones que contiene que de alguna u otra forma siempre son de ayuda en las tareas diarias [7].

Las características más notables de este sistema operativo móvil son:

1. Interfaz elegante e intuitiva.
2. Funcionalidades y apps integradas para que tú y tu dispositivo hagan más.
3. Hardware y software hechos el uno para el otro.
4. Con la App Store, casi no hay límite para lo que tu dispositivo iOS puede hacer.
5. iCloud.
6. Seguridad.
7. Privacidad.
8. iOS habla muchos idiomas.
9. Fácil de actualizar.

2.2.2.3. Windows Phone

Es un sistema operativo móvil desarrollado por la empresa Microsoft para teléfonos inteligentes y otros dispositivos móviles. Fue lanzado al mercado el 21 de octubre de 2010 en Europa y el 8 de Noviembre en Estados Unidos, con la finalidad de suplantar el conocido Windows Mobile. Microsoft decidió realizar un cambio completo en este nuevo sistema operativo con respecto al otro, no solo se cambió el nombre, sino que se desarrolló desde cero, presentando una interfaz completamente nueva, mejor comportamiento y un mayor control sobre las plataformas de hardware que lo ejecutan, todo con el propósito de volver a ser competitivo en el mundo de los móviles [8].

2.2.2.4. Semejanzas y diferencias

- Facilidad de uso

La facilidad de uso es una de las principales diferencias entre estos tres sistemas. iOS y Windows Phone diseñados para ser completamente intuitivos, en lo que aventajan por mucho al Android, que a pesar de los esfuerzos de Google y desarrolladores independientes, sigue siendo un poco complejo de configurar y utilizar en ciertos aspectos. Windows Phone ocupa el primer puesto por ser sumamente sencillo de utilizar, hace falta escasos clics para acceder a cualquier función.

- Aplicaciones de terceros

Otra diferencia muy marcada entre los tres sistemas, Android permite que cualquier desarrollador pueda realizar apps o programas para el sistema operativo, lo que le otorga una enorme cantidad de aplicaciones disponibles, algunas gratuitas y otras de pago, lo que aventaja enormemente a iOS que sólo permite utilizar aplicaciones desarrolladas por Apple o por grandes compañías de software. Windows Phone, sin embargo, a pesar de llevar casi la misma política de iOS, ha lanzado una enorme cantidad de aplicaciones, que pudieran otorgarle un segundo lugar en este sentido.

- Eficiencia

La eficiencia de estos sistemas puede medirse de varias formas, y el uso de batería es una de ellas. Windows Phone lleva la delantera en este sentido, mientras que iOS le sigue de cerca, Android lleva la peor parte en este aspecto. La búsqueda de voz es otro factor a tomar en cuenta, Android esta en primer lugar integrando búsquedas en Google muy poderosas, seguido de Windows Phone y por último iOS.

La apariencia un aspecto muy importante, iOS con una interfaz elegante y sofisticada ocupa el primer lugar, seguido de Windows Phone y dejando muy atrás a Android.

2.2.3. Sistema operativo Android

Android constituye una pila de software pensada especialmente para dispositivos móviles y que incluye tanto un sistema operativo, como middleware y diversas aplicaciones de usuario. Representa la primera incursión seria de Google en el mercado móvil y nace con la pretensión de extender su filosofía a dicho sector.

Todas las aplicaciones nativas para Android se programan en lenguaje Java y son ejecutadas en una máquina virtual especialmente diseñada para esta plataforma, que ha sido bautizada con el nombre de Dalvik. El núcleo de Android está basado en Linux 2.6.

Con Android se busca reunir en una misma plataforma todos los elementos necesarios que permitan al desarrollador controlar y aprovechar al máximo cualquier funcionalidad ofrecida por un dispositivo móvil (llamadas, mensajes de texto, cámara, agenda de contactos, conexión Wi-Fi, Bluetooth, aplicaciones ofimáticas, videojuegos, etc.), así como poder crear aplicaciones que sean verdaderamente portables, reutilizables y de rápido desarrollo. En otras palabras, Android quiere mejorar y estandarizar el desarrollo de aplicaciones para cualquier dispositivo móvil y, por ende, acabar con la perjudicial fragmentación existente hoy día.

Además de todo ello, otro aspecto básico para entender la aparición de Android es que pretende facilitar la integración de estos dispositivos con las posibilidades cada día mayores ofrecidas por la Web. Por ejemplo, una aplicación desarrollada en Android podría ser aquella que indicase al usuario, a través de Google Maps, la localización de sus diferentes contactos de la agenda y que avisase cuando éstos se encuentren a una distancia cercana o en una ubicación determinada.

Mejorar el desarrollo y enriquecer la experiencia del usuario se convierte, por tanto, en la gran filosofía de Android y en su principal objetivo [9].

2.2.4. Arquitectura de Android

Android es una plataforma para dispositivos móviles que contiene una pila de software donde se incluye un sistema operativo, middleware y aplicaciones básicas para el usuario. En las siguientes líneas se dará una visión global por capas de cuál es la arquitectura empleada en Android. Cada una de estas capas utiliza servicios ofrecidos por las anteriores, y ofrece a su vez los suyos propios a las capas de niveles superiores, tal como se escribe a continuación [10].

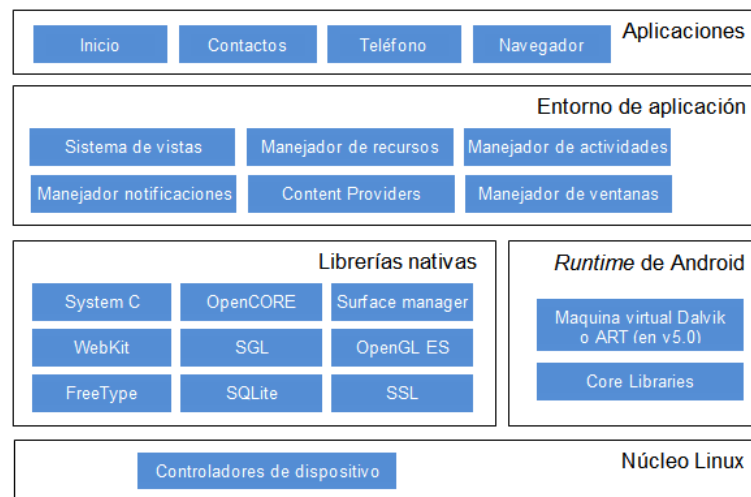


Figura 2.1: Arquitectura de Android

Fuente: [11]

- **Aplicaciones:** Este nivel contiene, tanto las incluidas por defecto de Android como aquellas que el usuario vaya añadiendo posteriormente, ya sean de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las API y librerías de los niveles anteriores.
- **Framework de Aplicaciones:** Representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación. Toda aplicación que se

desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por Google o terceras compañías, o incluso las que el propio usuario cree, utilizan el mismo conjunto de API y el mismo "framework", representado por este nivel. Entre las API más importantes ubicadas aquí, se pueden encontrar las siguientes:

- *Activity Manager*: Conjunto de API que gestiona el ciclo de vida de las aplicaciones en Android.
 - *Window Manager*: Gestiona las ventanas de las aplicaciones y utiliza la librería Surface Manager.
 - *Telephone Manager*: Incluye todas las API vinculadas a las funcionalidades propias del teléfono (llamadas, mensajes, etc.).
 - *Content Provider*: Permite a cualquier aplicación compartir sus datos con las demás aplicaciones de Android. Por ejemplo, gracias a esta API la información de contactos, agenda, mensajes, etc. será accesible para otras aplicaciones.
 - *View System*: Proporciona un gran número de elementos para poder construir interfaces de usuario, como listas, mosaicos, botones, tamaño de ventanas, control de las interfaces mediante teclado, etc. Incluye también algunas vistas estándar para las funcionalidades más frecuentes.
 - *Location Manager*: Posibilita a las aplicaciones la obtención de información de localización y posicionamiento.
 - *Notification Manager*: Mediante el cual las aplicaciones, usando un mismo formato, comunican al usuario eventos que ocurran durante su ejecución: una llamada entrante, un mensaje recibido, conexión Wi-Fi disponible, ubicación en un punto determinado, etc. Si llevan asociada alguna acción, en Android denominada Intent, (por ejemplo, atender una llamada recibida) ésta se activa mediante un simple clic.
- **Bibliotecas**: La siguiente capa se corresponde con las librerías utilizadas por Android. Éstas han sido escritas utilizando C/C++ y proporcionan a Android la mayor parte de sus capacidades más características. Junto al núcleo basado en Linux, estas librerías constituyen el corazón de Android. Entre las librerías más importantes ubicadas aquí, se pueden encontrar las siguientes:
- *Biblioteca libc*: Incluye todas las cabeceras y funciones según el estándar del lenguaje C. Todas las demás librerías se definen en este lenguaje.

- *Biblioteca Surface Manager*: Es la encargada de componer los diferentes elementos de navegación de pantalla. Gestiona también las ventanas pertenecientes a las distintas aplicaciones activas en cada momento.
- *Biblioteca Media Libraries*: Proporciona todos los códecs necesarios para el contenido multimedia soportado en Android (vídeo, audio, imágenes estáticas y animadas, etc.)
- *FreeType*: Permite trabajar de forma rápida y sencilla con distintos tipos de fuentes.
- *Biblioteca SSL(Secure Sockets Layer)*: Posibilita la utilización de dicho protocolo para establecer comunicaciones seguras.
- *Biblioteca SQLite*: Creación y gestión de bases de datos relacionales.
- *Biblioteca WebKit*: Proporciona un motor para las aplicaciones de tipo navegador y forma el núcleo del actual navegador incluido por defecto en la plataforma Android.

2.2.5. Diseño adaptativo

El diseño web responsive o adaptativo es una técnica de diseño web que busca la correcta visualización de una misma página en distintos dispositivos. Desde ordenadores de escritorio a tablets y móviles.

Se trata de redimensionar y colocar los elementos de la web de forma que se adapten al ancho de cada dispositivo permitiendo una correcta visualización y una mejor experiencia de usuario. Se caracteriza porque los layouts (contenidos) e imágenes son fluidos y se usa código media-queries de CSS (Cascading Style Sheets). El diseño responsive permite reducir el tiempo de desarrollo, evita los contenidos duplicados, y aumenta la viralidad de los contenidos ya que permite compartirlos de una forma mucho más rápida y natural [12].

Características

1. Los layout e imágenes son fluidos y se adaptan a cada pantalla.
2. Permite reducir el tiempo de desarrollo
3. Evita los contenidos duplicados
4. Aumenta la viralidad de los contenidos porque permite compartirlos de una forma mucho más rápida y natural.

Sitio web adaptativo en diferentes pantallas

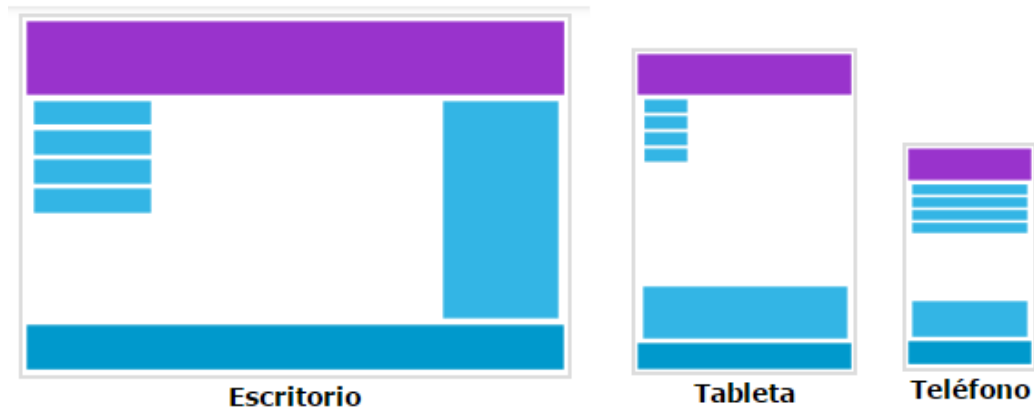


Figura 2.2: Diseño adaptable en distintos dispositivos
Fuente: [13]

Marcos de trabajo para crear sitios web adaptativos

Los frameworks CSS consiguen en definitiva ahorrar tiempo y esfuerzo al diseñador web, ya que parte desde una base sólida y testeada, entonces pueden centrar sus esfuerzos en la estructura lógica de la web y de los elementos básicos. A continuación se enlistan tres ejemplos de frameworks para desarrollar un sitio responsive.

- ***Bootstrap***: Es el más popular en la red donde se puede encontrar un sin número de ejemplos de como implementar cada uno de sus componentes, el crecimiento de su comunidad hacen posible encontrar plantillas pagadas y gratuitas, lo que aligera el desarrollo web.
- ***Foundation***: Inicialmente lanzado como un framework de front-end de código abierto, uno de sus clientes fue Facebook y eBay. Sus objetos de interfaz son similares a los Bootstrap pero cuenta con una comunidad 'no tan grande.
- ***Pure CSS***: La ventaja de este marco de trabajo es su ligero tamaño, con un peso minimizado de 5.7KB, esto podría ayudar en la carga rápida de la aplicación web. Además está basado en Normalize CSS y no utiliza JavaScript por lo que es posible utilizarlo en cualquier proyecto web.

Razones para desarrollar una web con Bootstrap

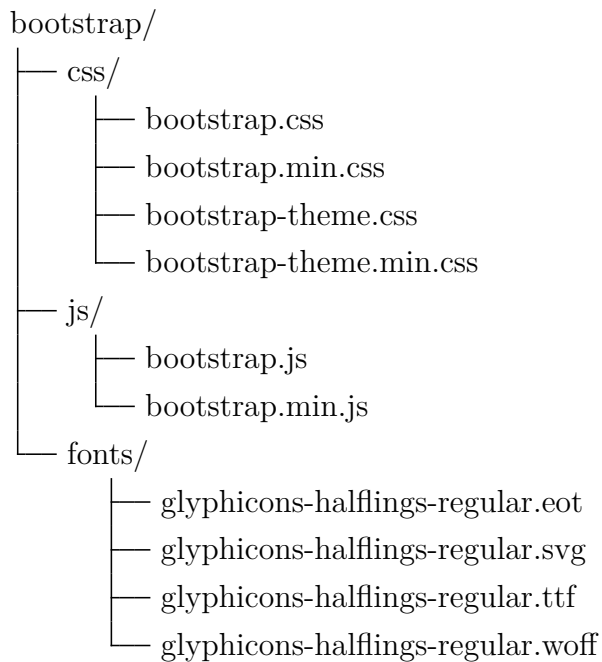
La creación de páginas web adaptativos se facilitan cuando se utiliza un framework CSS para ello, debido a que ya están probadas y normalizadas por grandes empresas de software, además de las ventajas anotadas anteriormente. Después de una búsqueda se encontró que Bootstrap presenta ventajas significativas para el desarrollo de sitios web y tiene más popularidad en la comunidad de desarrolladores web. La investigación mostró estos datos a favor de este marco de trabajo por lo que se decidió usarla para el desarrollo del presente proyecto.

- ***Es fácil de utilizar:*** La utilización de este marco de trabajo es relativamente fácil para quien ya practica las hojas de estilos en las páginas web, y si se presenta una duda de implementación de algunos de sus componentes se dispone además de la documentación oficial, otras fuentes de consulta como son repositorios y páginas de terceros.
- ***Creado por Twitter:*** Cuenta con el respaldo de un grande respecto a las redes sociales como es Twitter, que quienes están constantemente actualizándola en beneficio de los consumidores.
- ***Comunidad Web:*** Como se dijo anteriormente su comunidad web es grande y sigue creciendo gradualmente. Un punto a favor para el diseñador que desea observar ejemplos del uso de este marco de trabajo.
- ***JavaScript:*** Bootstrap está un poco más avanzado en JS porque sus componentes son más simples, usan menos código y están bien testados.
- ***Formularios:*** Tiene un mejor diseño para los inputs, radios y checkboxes, lo que mejora el aspecto de la página en cualquier navegador que se visualice.
- ***Documentación:*** Toda la documentación de este marco de trabajo se lo puede encontrar en su sitio oficial, junto a ejemplos de como implementar cada uno de sus componentes a nuestro sitio web. También existen repositorios, blog , cursos, etc que nos muestran como utilizarlo adecuadamente.

2.2.5.1. Bootstrap

Es un marco de trabajo originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo [14].

Contenidos de la versión original de Bootstrap



Componentes de Bootstrap

Bootstrap brinda una gran cantidad de componentes reutilizables construidos para proporcionar la iconografía, menús desplegados, los grupos de entrada, navegación, alertas, y mucho más. Entre ellos se tiene:

1. Glyphicons
2. Dropdowns
3. Headers
4. Divider
5. Button groups
6. Button toolbar
7. Nesting
8. Vertical variation
9. Button dropdowns
10. Split button dropdowns
11. Sizing
12. Dropup variation

Sitio oficial de Bootstrap para más detalles: <http://getbootstrap.com/>

2.2.6. Patrones de diseño de software

Un patrón de diseño es una solución repetible a problemas típicos y recurrentes en el diseño del software. Son soluciones basadas en la experiencia y que se ha demostrado que funcionan. No son un diseño terminado que puede traducirse directamente a código, sino más bien una descripción sobre cómo resolver el problema, la cual puede ser utilizada en diversas situaciones.

Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias [15].

“Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí para resolver un problema de diseño general en un contexto particular” [16].

Categorías de patrones

Según la escala o nivel de abstracción [15]:

- **Patrones de arquitectura:** Aquellos que expresan un esquema organizativo estructural fundamental para sistemas de software.
- **Patrones de diseño:** Aquellos que expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas de software.
- **Dialectos:** Patrones de bajo nivel específicos para un lenguaje de programación o entorno concreto.

Patrones de arquitectura

Los patrones arquitectónicos, o patrones de arquitectura, también llamados arquetipos ofrecen soluciones a problemas de arquitectura de software en ingeniería de software. Dan una descripción de los elementos y el tipo de relación que tienen junto con un conjunto de restricciones sobre cómo pueden ser usados. Un patrón arquitectónico expresa un esquema de organización estructural esencial para un sistema de software, que consta de subsistemas, sus responsabilidades e interrelaciones. En comparación con los patrones de diseño, los patrones arquitectónicos tienen un nivel de abstracción mayor [17].

Ejemplos de patrones arquitectónicos incluyen los siguientes:

- MVC (Model View Controller)

- Programación por capas
- Tres niveles
- Invocación implícita
- Arquitectura en pizarra
- Arquitectura dirigida por eventos

El patrón de arquitectura de software implementado en el presente proyecto con ayuda del framework ASP.NET (Active Server Pages) es MVC, con este patrón se pretende separar los datos y la lógica de negocio de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por el otro la interacción con el usuario.

2.2.6.1. MVC

Modelo Vista Controlador en español es un patrón de arquitectura de software que, utilizando 3 componentes (Vistas, Models y Controladores) separa la lógica de la aplicación de la lógica de la vista en una aplicación. Es una arquitectura importante puesto que se utiliza tanto en componentes gráficos básicos hasta sistemas empresariales; la mayoría de los frameworks modernos utilizan MVC para la arquitectura, entre ellos podemos mencionar Django, AngularJS y muchos otros más [18].

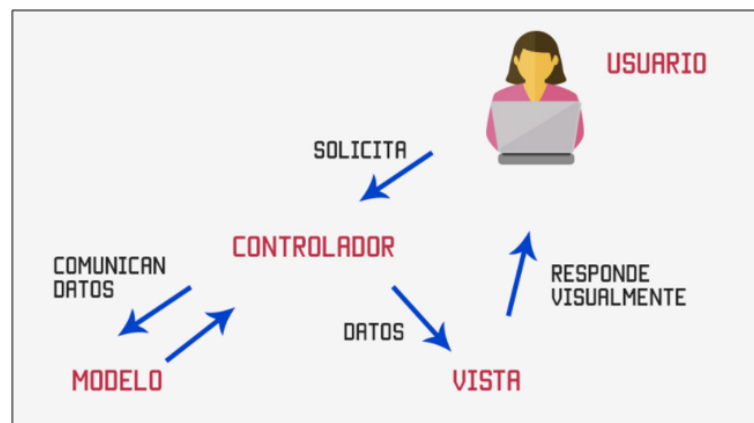


Figura 2.3: Diagrama de flujo - MVC
Fuente: [18]

El patrón de arquitectura "modelo vista controlador", es una filosofía de diseño de aplicaciones, compuesta por [19]:

1. Modelo

- a) Contiene el núcleo de la funcionalidad (dominio) de la aplicación.
- b) Encapsula el estado de la aplicación.
- c) No sabe nada / independiente del Controlador y la Vista.

2. Vista

- a) Es la presentación del Modelo.
- b) Puede acceder al Modelo pero nunca cambiar su estado.
- c) Puede ser notificada cuando hay un cambio de estado en el Modelo.

3. Controlador

- a) Reacciona a la petición del Cliente, ejecutando la acción adecuada y creando el modelo pertinente.

2.2.7. Aplicaciones web

En la ingeniería de software se denomina aplicación web a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una Intranet mediante un navegador. En otras palabras, es una aplicación software que se codifica en un lenguaje soportado por los navegadores web, y en la que se confía la ejecución de la aplicación al navegador. Las ventajas más significativas de las aplicaciones web son [20]:

- **Compatibilidad multiplataforma:** Las aplicaciones web tienen un camino mucho más sencillo para la compatibilidad multiplataforma que las aplicaciones de software descargables.
- **Actualización:** Las aplicaciones basadas en web están siempre actualizadas con el último lanzamiento.
- **Inmediatez de acceso:** Las aplicaciones basadas en web no necesitan ser descargadas, instaladas y configuradas. Usted accede a su cuenta online trabajar sin importar cuál es su configuración o su hardware.
- **Múltiples usuarios concurrentes:** Las aplicaciones basadas en web pueden realmente ser utilizada por múltiples usuarios al mismo tiempo.

Microsoft .NET como marco de trabajo para aplicaciones web

Luego de haber realizado un estudio de qué son y cómo se comportan las aplicaciones web, la siguiente actividad se centraría en hallar un marco de trabajo para el desarrollo de la aplicación web destinada al control de la aplicación Android. Microsoft la primera opción de trabajo por las ventajas e innovaciones constantes en el campo de desarrollo de software.

La empresa tecnológica Microsoft ha crecido exponencialmente desde sus inicios, ahora no solo distribuye su famoso sistema operativo Windows, si no que incursiona en la tecnología móvil con el sistemas operativo para teléfonos inteligentes Window Phone. Los negocios en la nube también han sido de preocupación para para esta multinacional, cubriendo esta demanda con Windows Azure que ofrece servicios de computo en la nube para pequeñas y grandes empresas. Estas son algunas de sus iniciativas, por no nombrar más.

Microsoft al ser una empresa de largo trayecto e impacto en la sociedad brinda seguridad, confianza, además de soporte técnico para todos sus productos, se la eligió para la implementación de la administración web.

A continuación las herramientas Microsoft utilizadas para el presente trabajo de investigación, cada una de ellas fue estudiada para su posterior utilización y también se midió su compatibilidad con otras plataformas como es Android.

2.2.8. Herramientas Microsoft

Luego de la elección de Microsoft como la proveedora del marco de trabajo para el desarrollo de la administración web, más específicamente el paquete ASP.NET que cuenta con su propio molde de trabajo para aplicaciones web que implementen el patrón de arquitectura de software Modelo Vista Controlador. Se eligió uno de los lenguajes de alto nivel que ASP.NET proporciona, como es C#, un lenguaje de fácil maniobrabilidad a la vez que elegante. Para el desarrollo de los servicios web y la comunicación con la aplicación móvil, ASP.NET posibilita la implementación de Web API's que son servicios web basados en REST (Representational State Transfer). A continuación conceptos más precisos de lo hablado anteriormente.

2.2.8.1. C#

Visual C# es uno de los lenguajes de programación de alto nivel que pertenecen al paquete .NET otros lenguajes son Visual Basic, C/C++. Con él se pueden escribir tanto programas convencionales como para Internet. El paquete .NET incluye un compilador (programa traductor de C# que produce un código escrito en un lenguaje

intermedio, común para todos los lenguajes de dicha plataforma, que será el que la máquina virtual ejecutará) esto es, cada lenguaje de la plataforma tiene su compilador que produce código correspondiente a un único lenguaje.

Una característica importante del lenguaje C#, es que es un lenguaje de programación orientado a objetos (POO). Además es fácil de aprender. Tiene un tamaño pequeño que favorece el desarrollo y reduce las posibilidades de cometer errores; a la vez es potente y flexible [21].

2.2.8.2. ASP.NET

ASP.NET es un Framework para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios web XML. apareció en enero de 2002 con la versión 1.0 del .Net Framework, y es la tecnología sucesora de la tecnología Active Server Pages (ASP). ASP.NET esta construido sobre el Common Lenguaje Runtime, permitiendo a los programadores escribir código ASP.NET usando cualquier lenguaje admitido por el .NET Framework [22].

ASP.NET MVC

El marco de ASP.NET MVC proporciona una alternativa al modelo de formularios Web Forms de ASP.NET para crear aplicaciones web, es un marco de presentación de poca complejidad y fácil de comprobar que (como las aplicaciones basadas en formularios Web Forms) se integra con las características de ASP.NET existentes, tales como páginas maestras y la autenticación basada en pertenencia. El marco de MVC se define en el ensamblado System.Web.Mvc [23].

ASP.NET Web API

Es un framework de la familia .NET que tiene como objetivo el facilitarnos en gran medida la construcción de servicios basados en REST, como lo usan actualmente Twiter, World of Warcraft, Amazone o Windows Azure. Nos ahorra el trabajo repetitivo, y nos provee de una arquitectura MVC con unas buenas bases desde la que iniciarnos en el desarrollo, también incluye las referencias para utilizar Entity Framework [24].

Los beneficios intrínsecos en este tipo de servicios basados en REST son.

- Escalabilidad de la interacción con los componentes. La web ha crecido exponencialmente sin degradar su rendimiento. Una prueba de ellos es la

variedad de clientes que pueden acceder a través de la web: estaciones de trabajo, sistemas industriales y dispositivos móviles.

- Compatibilidad con componentes intermedios. Los más populares intermedios son varios tipos de proxys para Web. Algunos de ellos, las caches, se utilizan para mejorar el rendimiento. Otros permiten reforzar las políticas de seguridad: firewalls.
- Generalidad de interfaces. Gracias al protocolo HTTP (HyperText Transfer Protocol), cualquier cliente puede interactuar con cualquier servidor HTTP sin ninguna configuración especial. Esto no es del todo cierto para otras alternativas, como SOAP (Simple Object Access Protocol) para los Servicios Web.

Los métodos HTTP más importantes son PUT, GET, POST y DELETE. Ellos suelen ser comparados con las operaciones asociadas a la tecnología de base de datos, operaciones CRUD: CREATE, READ, UPDATE, DELETE [25].

La tabla siguiente muestra con claridad los métodos HTTP:

Acción	HTTP	SQL
Create	PUT	Insert
Read	GET	Select
Update	POST	Update
Delete	DELETE	Delete

Tabla 2.1: Métodos HTTP de REST

Fuente: [25]

Servidor IIS

Los servicios de Internet Information Server (o IIS), son los servicios de software que admiten la creación, configuración y administración de sitios Web o aplicaciones web, además de otras funciones de Internet. Este servicio convierte a un PC en un servidor web para Internet o una Intranet, es decir que en los ordenadores que tienen este servicio instalado se pueden publicar páginas web tanto local como remotamente. Se basa en varios módulos que le dan capacidad para procesar distintos tipos de páginas. Por ejemplo, Microsoft incluye los de Active Server Pages (ASP) y ASP.NET. También pueden ser incluidos los de otros fabricantes, como PHP3 o Perl [26].

2.2.8.3. SQL Server

Es un Sistema Gestor de Bases de datos relacionales (SGBD) que además ahora en sus versiones más actuales cuenta con diferentes tipos de herramientas incorporadas en el programa, está basado en el lenguaje Transact-SQL y es capaz de poner grandes cantidades de información a muchos usuarios simultáneamente. Almacena datos de documentos estructurados, semiestructurados o no estructurados como imágenes, música y archivos directamente dentro de la base de datos. Se obtiene más rendimiento de los datos, poniendo a disposición servicios integrados como son consultas, búsquedas, sincronizaciones, informes y análisis [27].

2.2.9. Escuelas de capacitación para conductores profesionales

Para mejor entendimiento de todo el contexto que abarca la realización del presente proyecto se continúa a nombrar textualmente algunas de las normas y reglamentos para Escuelas de Capacitación de Conductores Profesionales emitida por la ANT (Agencia Nacional de Tránsito), las cuales son necesarias tener en cuenta [28].

ECCP

Se denominará Escuelas de Capacitación para Conductores Profesionales - ECCP, a los Sindicatos de Conductores Profesionales; a los Institutos Técnicos de Educación Superior; Universidades y Escuelas Politécnicas debidamente autorizados por la Secretaría Nacional de Educación Superior - SENESCYT; a la Federación Ecuatoriana de Operadores y Mecánicos de Equipos Camioneros - FEDESOMECEC; y, al Servicio Ecuatoriano de Capacitación Profesional - SECAP, debidamente autorizados por la ANT, que se encuentren habilitados en el país para la capacitación y formación de conductores profesionales.

Estructura organizacional

Las escuelas de capacitación para conductores profesionales - ECCP, para su funcionamiento deberán tener como mínimo la siguiente estructura organizacional:

1. Director General Administrativo
2. Director Pedagógico
3. Tesorero
4. Secretario
5. Consejo Académico

6. Cuerpo Docente de nivel superior reconocido por el órgano competente
7. Asesor Técnico en Educación y Seguridad Vial
8. Instructores de Conducción y Educación Vial
9. Inspector
10. Contador
11. Personal administrativo indispensable para su correcto funcionamiento

De los alumnos

1. Se consideran alumnos de la Escuela de Capacitación para Conductores profesionales, a los aspirantes a obtener el título de conductor. Previa la matriculación de los cursos de capacitación, el alumno deberá cumplir con los requisitos constantes en el presente reglamento.
2. Al momento de la matrícula, los alumnos adquieren la obligación de observar los horarios respectivos en la ECCP. Cualquier inasistencia, deberán justificarla con el Inspector.
3. El período de duración de los cursos de capacitación de conductores profesionales, será diferente de acuerdo al tipo de licencia: C, C1, D, D1, E, E1, y G; así como también en el tipo de licencia A1, debiendo cumplir las horas presenciales señaladas en la respectiva malla curricular que será aprobada por la Dirección Ejecutiva de la ANT.
4. Las horas teóricas y prácticas impartidas para los diferentes cursos deberán ser consideradas horas pedagógicas de 45 minutos y el cupo por aula no podrá ser superior a 30 alumnos
5. La asistencia de los alumnos a las horas teóricas deberá ser mínimo de un 80% y 90% a las horas de práctica y recuperará las mismas, en caso de ausencia justificada, con el fin, de cumplir la totalidad de la malla aprobada caso contrario, no aprobará el curso
6. Las calificaciones deberán ser efectuadas de acuerdo a una escala de 0 a 20 puntos para determinar el logro de los módulos teóricos y prácticos. La calificación mínima requerida para la aprobación de cada materia será de 16 sobre 20.

2.3. Propuesta de solución

Desarrollar una aplicación móvil bajo la plataforma Android que será controlada por un panel de administración, ambas mantendrán una constante comunicación recibiendo y enviando información con el uso de Web API's (web service). El panel de administración estará desarrollado en el Framework de Microsoft .NET por lo que será una aplicación ASP.NET, este servirá para el ingreso de datos de administrativos, docentes y demás información relevantes para el conjunto en sí.

Los administrativos de la institución cuando lo requieran podrán acceder con sus credenciales al panel de administración alojada en un hosting remoto por medio de un navegador web e Internet, una vez autenticados pueden enviar una notificación hacia un determinado estudiante debidamente matriculado y registrado en la base de datos del sistema. Los docentes que imparten los módulos también tendrán acceso a la aplicación web para poder subir las lista de calificaciones de determinado paralelo.

La aplicación móvil una vez instalada en el smartphome de un estudiante y debidamente autenticado tendrá acceso inmediato a la información publicada en la aplicación web, este podrá saber las mensajes enviados por cualquier administrativo así como su puntaje académico en cada uno de sus módulos que va recibiendo de acuerdo a la malla curricular aprobada por la ANT (Agencia Nacional de Tránsito) para conductores profesionales.

CAPÍTULO 3

Metodología

3.1. Modalidad básica de la investigación

3.1.1. Investigación de campo

Se considera esta modalidad porque el investigador acudirá al lugar o establecimiento en donde se producen los hechos para obtener información referente a los objetivos del proyecto de grado. Las técnicas a ser utilizadas para la obtención de la información serán: las encuestas, las entrevistas y la observación.

3.1.2. Investigación bibliográfica - documental

Se considera esta modalidad por la razón de que es necesaria acudir a distintas fuentes obtenidas de libros, artículos y revistas científicas, tesis desarrolladas en otras instituciones, proyectos de empresas, con la finalidad de obtener información referente al proyecto de grado, que nos permite profundizar el tema de la investigación.

3.1.3. Investigación aplicada

Por la utilización de los conocimientos adquiridos a lo largo de la carrera universitaria.

3.2. Recolección de información

La información relacionada a los datos se recolectará en la escuela de formación de conductores profesionales, así como también la información de campo de investigación referente al problema, se obtendrá a través de Internet mediante la revisión de documentos técnicos como: tesis universitarias, libros relacionados a la programación, blogs informáticos, cursos online y artículos científicos, con la finalidad de alcanzar los objetivos planteados.

3.3. Procesamiento y análisis de datos

Lo primero que se realizará al recopilar la información, será analizar los resultados obtenidos con relación al problema ya planteado y seleccionar los que se requiere para el desarrollo del proyecto para poder cumplir con los objetivos planteados anteriormente.

Para el procesamiento de la información se realizará las siguientes actividades:

1. Recolección de datos referente a los requisitos para la aplicación móvil a través de entrevistas y encuestas realizadas al personal administrativo y estudiantes que conforman la escuela.
2. Revisión del reglamento institucional relativo a las actividades del personal administrativo y estudiantes durante un ciclo académico.
3. Revisión, análisis y clasificación de la información obtenida.
4. Tabulación de la información clasificada.
5. Análisis y revisión de artículos y documentos relacionados a la investigación referente al problema y a la información obtenida.
6. Interpretación de los resultados, haciendo uso de gráficos interactivos para facilitar el respectivo análisis y la interpretación.
7. Análisis y revisión de los lineamientos, normas y estándares que deben ser utilizados para el desarrollo de aplicaciones móviles.

3.4. Desarrollo del proyecto

Para el desarrollo del presente proyecto se ha optado por utilizar una metodología fuera de las tradicionales, tal es el caso de XP que se colocó y preferido por la mayoría de programadores para el desarrollo de aplicaciones móviles. A continuación se explicará conceptos para el mejor entendimiento de la metodología seleccionada.

3.4.1. Metodología

“Una metodología es una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por implementar nuevos sistemas de información. Una metodología está formada por fases, cada una de las cuales se puede dividir en sub-fases, que guiarán a los desarrolladores de sistemas a elegir las técnicas más apropiadas en cada momento del proyecto y también a planificarlo, gestionarlo, controlarlo y evaluarlo” [29].

3.4.2. Diferencias entre metodología tradicionales y ágiles

A continuación se muestran en forma de cuadro las mas notables diferencias entre los dos tipos de metodología de desarrollo de software.

Metodología Tradicionales	Metodología Ágiles
Más Artefactos. El modelado es esencial, mantenimiento de modelos	Pocos Artefactos. El modelado es prescindible, modelos desechables.
Más Roles, más específicos	Pocos Roles, más genéricos y flexibles
Existe un contrato prefijado	No existe un contrato tradicional, debe ser bastante flexible
El cliente interactúa con el equipo de desarrollo mediante reuniones	Cliente es parte del equipo de desarrollo
Aplicables a proyectos de cualquier tamaño, pero suelen ser especialmente efectivas/usadas en proyectos grandes y con equipos posiblemente dispersos	Orientada a proyectos pequeños. Corta duración (o entregas frecuentes), equipos pequeños (< 10 integrantes) y trabajando en el mismo sitio
Se promueve que la arquitectura se defina tempranamente en el proyecto	La arquitectura se va definiendo y mejorando a lo largo del proyecto
Énfasis en la definición del proceso: roles, actividades y artefactos	Énfasis en los aspectos humanos: el individuo y el trabajo en equipo
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo	Basadas en heurísticas provenientes de prácticas de producción de código
Se espera que no ocurran cambios de gran impacto durante el proyecto	Se esperan cambios durante el proyecto

Tabla 3.1: Metodologías tradicionales y ágiles

Fuente: [30]

3.4.3. Metodología XP (Extreme Programming)

Después de un estudio de las metodologías ágiles y las ventajas que presentan frente a las metodologías tradicionales se ha seleccionado Extreme Programming por las razones que se señalan a continuación.

- **Rápida respuesta a los cambios.** Al ser procesos evolutivos, los equipos

de trabajo pueden implementar soluciones sobre la marcha. Ya no es necesario esperar hasta el final para corregir fallos.

- **Intervención del cliente en el proceso.** El cliente interviene de una forma activa en cada una de las etapas del proceso. Puede aportar ideas y opinar sobre los resultados que se le van entregando progresivamente.
- **Entregas del producto a intervalos.** Las entregas parciales o en bloques mejoran la optimización de recursos y optimizan las labores de seguimiento y control. El producto final es, en realidad, la suma de varios productos parciales que han sido monitorizados varias veces.
- **Eliminación de tareas innecesarias.** Al priorizar las tareas de un proceso, los responsables del mismo saben con certeza cuáles tienen un mayor peso y cuáles resultan secundarias o, incluso, innecesarias. Esta distinción ayuda a centralizar esfuerzos y a unificar criterios de actuación.

3.4.3.1. Definición

Es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico [31].

3.4.3.2. Valores

Extreme Programming se basa en los siguientes valores principales [32]:

1. Comunicación: La necesidad de los desarrolladores de intercambiar ideas e información sobre el proyecto ya sea con los directores del proyecto o los usuarios de forma confiable, y fácil. La comunicación debe ser continua y rápida.
2. Sencillez: Cuando se tenga que elegir entre varias opciones, en lo posible elegir soluciones simples, sin que esto signifique aplicar enfoques simplistas; la programación extrema define un diseño simple en el que se realice el menor número posible de clases, métodos y que no tenga código duplicado.

3. Retroalimentación: Debe ser rápida en todos los niveles, principalmente se consigue ejecutando y probando el código, por lo que las entregas tempranas y frecuentes son muy importantes.
4. Coraje: Todas las personas que participen en el proyecto deben tener la capacidad de expresar su valoración sobre el proyecto. Deberían ser abiertos y dejar que todos revisen e incluso modificasen su trabajo. Los cambios no deberían ser vistos con terror y los desarrolladores deberían tener el valor de encontrar mejores soluciones y modificar el código siempre que sea necesario y factible.
5. Respeto: Debe manifestarse en diversas formas y situaciones, son la base para una buena relación y cooperación entre todos los componentes del equipo de trabajo.

3.4.3.3. Las historias de usuario

Las historias de usuario son la técnica utilizada en XP para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas [33].

3.4.3.4. El proceso XP

Un proyecto XP tiene éxito cuando el cliente selecciona el valor de negocio a implementar basado en la habilidad del equipo para medir la funcionalidad que puede entregar a través del tiempo. El ciclo de desarrollo consiste en los siguientes pasos

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.

5. Vuelve al paso 1.

El ciclo de vida ideal de XP consiste de seis fases: Exploración, Planificación de la Entrega, Iteraciones, Producción, Mantenimiento y Muerte del Proyecto [33].

Fase I: Exploración

En esta fase, los clientes plantean a grandes rasgos las historias de usuario que son de interés para la primera entrega del producto. Al mismo tiempo el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Se prueba la tecnología y se exploran las posibilidades de la arquitectura del sistema construyendo un prototipo.

Fase II: Planificación de la Entrega

En esta fase el cliente establece la prioridad de cada historia de usuario, y correspondientemente, los programadores realizan una estimación del esfuerzo necesario de cada una de ellas. Se toman acuerdos sobre el contenido de la primera entrega y se determina un cronograma en conjunto con el cliente. Una entrega debería obtenerse en no más de tres meses. Esta fase dura unos pocos días.

Fase III: Iteraciones

Los elementos que deben tomarse en cuenta durante la elaboración del Plan de la Iteración son: historias de usuario no abordadas, velocidad del proyecto, pruebas de aceptación no superadas en la iteración anterior y tareas no terminadas en la iteración anterior. Todo el trabajo de la iteración es expresado en tareas de programación, cada una de ellas es asignada a un programador como responsable, pero llevadas a cabo por parejas de programadores.

Fase IV: Producción

La fase de producción requiere de pruebas adicionales y revisiones de rendimiento antes de que el sistema sea trasladado al entorno del cliente. Al mismo tiempo, se deben tomar decisiones sobre la inclusión de nuevas características a la versión actual, debido a cambios durante esta fase.

Fase V: Mantenimiento

Mientras la primera versión se encuentra en producción, el proyecto XP debe mantener el sistema en funcionamiento al mismo tiempo que desarrolla nuevas iteraciones. Para realizar esto se requiere de tareas de soporte para el cliente. De esta forma, la velocidad de desarrollo puede bajar después de la puesta del sistema en producción.

La fase de mantenimiento puede requerir nuevo personal dentro del equipo y cambios en su estructura.

Fase VI: Muerte del Proyecto

Es cuando el cliente no tiene más historias para ser incluidas en el sistema. Esto requiere que se satisfagan las necesidades del cliente en otros aspectos como rendimiento y confiabilidad del sistema. Se genera la documentación final del sistema y no se realizan más cambios en la arquitectura. La muerte del proyecto también ocurre cuando el sistema no genera los beneficios esperados por el cliente o cuando no hay presupuesto para mantenerlo.

CAPÍTULO 4

Desarrollo de la propuesta

4.1. Datos informativos

Título

Aplicación móvil de gestión administrativa y académica en la “Escuela de Formación y Capacitación de Conductores Profesionales Ambato” utilizando la plataforma Android.

Institucionales

- Facultad de ingeniería en Sistemas, Electrónica e Industrial
- Escuela de Formación y Capacitación de Conductores Profesionales Ambato

Beneficiarios

Administrativos, Docentes y Estudiantes de la Escuela de Formación de Conductores Profesionales.

Ubicación

Provincia:	Tungurahua
Cantón:	Ambato
Dirección:	Av. Los Incas y Nazacota Puento
Teléfono:	03 2840618 - 2421272

Equipo responsable

Tutor:	Ing. Mg. Clay Aldás Flores.
Investigador:	Sr. Luis Alberto Marcalla

Tomando en cuenta que la metodología Extreme Programming (XP) es la seleccionada es la seleccionada, se detalla a continuación el proceso de desarrollo del proyecto.

4.2. Fase I - Exploración

Fase inicial de la Xtreme Programming donde se realiza las historias de usuario conjuntamente con el propietario de software a desarrollarse, simultáneamente el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto.

Hacer un análisis de los procesos inherentes en la Escuela de Capacitación y Formación de Conductores profesionales también es necesario hacerlo en esta fase para tener una mejor visión del software a desarrollar, la institución educativa dedicada a la introducción de los nuevos conductores profesionales del Ecuador no se diferencia de las demás instituciones de nivel secundario. La institución en estudio dispone de personal administrativo como son secretaria, inspector, docentes encargados de impartir cada uno de los módulos de la malla curricular para la obtención del título de conductor profesional, también instructores los cuales están encargados de las horas de práctica de conducción.

Los categorías de licencia en la institución son de tipo C, D y E cada una de ellas dispone de sus módulos respectivos de acuerdo a la malla curricular y cada periodo académico dura de 5 a 9 meses dependiendo del tipo de licencia optado, las convalidaciones de módulos son permitidas, cuando alguien por ejemplo tiene licencia tipo C y quiere promoverse a la D entonces solo tomaría los módulos que le faltan completar de la malla curricular. La apertura de cursos de capacitación va depender de la demanda de estudiantes y las disposiciones de la ANT que es encargado de dar los permisos respectivos. Para la obtención de información necesaria durante el transcurso del presente trabajo se optó por buscarla en el Reglamento para Escuelas de Conductores Profesionales emitida por su ente rector que es la Agencia Nacional de Tránsito.

Durante la realización de esta fase se señaló que la aplicación móvil debe de tener un panel de administración para darle más versatilidad y comodidad al proyecto, teniendo en cuenta esto se decidió realizar un panel de administración. Luego de un periodo de investigación y análisis se decidió desarrollar una aplicación ASP.NET con la ayuda de Visual Studio utilizando LINQ (Language Integrated Query) para acceder a la base de datos, el patrón de arquitectura MVC para la programación, el

uso de estilos CSS de Bootstrap para que el sitio sea adaptativo y finalmente el uso de los Web API's para la comunicación con la aplicación móvil.

Teniendo estos puntos en cuenta se prosigue a la documentación de la aplicación móvil y su panel de administración el cual es una aplicación web desarrollada en ASP.NET.

4.2.1. Historias de usuario

Las historias de usuario esenciales para la obtención de los requerimientos por parte del desarrollador fueron realizadas en la localidad de la institución junto con un representante de la misma. Las historias de usuario realizadas fueron divididas según si correspondían a la aplicación web o a la aplicación móvil.

4.2.1.1. Aplicación web (Administración)

A continuación se detallan las historias de usuario desarrolladas conjuntamente con un representante de la institución para el desarrollo de la aplicación web.

Historia de Usuario	
Número: 1	Nombre: Acceso al sistema web
Usuario: Administrador	
Modificación de historia número: 1	Iteración Asignada: 1
Prioridad de negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 8.5
Riesgo de Desarrollo: Media (Alta/Media/Baja)	Puntos Reales: 9.0
Programador responsable: Luis Marcalla	
Descripción: La aplicación web destinada a la administración de la aplicación Android debe de disponer de tres tipos de usuarios (Administrador/Administrativo/Docente), cada uno con sus respectivas funciones y restricciones.	
Observaciones: Debido a los cambios repentinos de los horarios de clase y la poca estadía del docente se prescinde de un control de fechas. Después de que cada usuario finalice de realizar sus tareas tendrá la opción de Salir.	

Tabla 4.1: Historia de usuario 1 - Acceso al sistema web

Elaborado por: Investigador

Historia de Usuario	
Número: 2	Nombre: Mantenimiento de tablas
Usuario: Administrador	
Modificación de historia número: 1	Iteración Asignada: 1
Prioridad de negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 9.0
Riesgo de Desarrollo: Alta (Alta/Media/Baja)	Puntos Reales: 9.5
Programador responsable: Luis Marcalla	
Descripción: El usuario Administrador está encargado de realizar el mantenimiento (inserciones, ediciones, lectura y eliminaciones) de las tablas del sistema, y de mantener los datos siempre consistentes.	
Observaciones: La integridad y consistencia de los datos deberán ser un hecho fiable dentro del contexto de la aplicación web y por consiguiente de la aplicación Android.	

Tabla 4.2: Historia de usuario 2 - Mantenimiento de tablas

Elaborado por: Investigador

Historia de Usuario	
Número: 3	Nombre: Ingreso de notas
Usuario: Docente	
Modificación de historia número: 1	Iteración Asignada: 1
Prioridad de negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 9.0
Riesgo de Desarrollo: Alta (Alta/Media/Baja)	Puntos Reales: 9.8
Programador responsable: Luis Marcalla	
Descripción: Luego de haber finalizado cada módulo el docente inmediatamente pasará a subir las notas de los estudiantes ingresando a la aplicación web.	
Observaciones: Debido al corto periodo de cada módulo y el repentino cambio de horarios se prescinde de una restricción de fechas.	

Tabla 4.3: Historia de usuario 3 - Ingreso de notas

Elaborado por: Investigador

Historia de Usuario	
Número: 4	Nombre: Envío de notificaciones
Usuario: Administrativo	
Modificación de historia número: 1	Iteración Asignada: 1
Prioridad de negocio: Baja (Alta/Media/Baja)	Puntos Estimados: 7.0
Riesgo de Desarrollo: Baja (Alta/Media/Baja)	Puntos Reales: 7.5
Programador responsable: Luis Marcalla	
Descripción: Tanto el usuario Inspector y Secretaria serán capaces de enviar notificaciones (mensajes personalizados) hacia cualquiera de los estudiantes debidamente registrados.	
Observaciones: Debido a que el Inspector y Secretaria tratan distintos asuntos con los estudiantes, cada usuario tendrá restringido alterar las acciones del otro usuario respectivamente.	

Tabla 4.4: Historia de usuario 4 - Envío de notificaciones

Elaborado por: Investigador

Historia de Usuario	
Número: 5	Nombre: Desarrollo de Web API's
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 2
Prioridad de negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 9.0
Riesgo de Desarrollo: Alta (Alta/Media/Baja)	Puntos Reales: 9.5
Programador responsable: Luis Marcalla	
Descripción: La aplicación web sirve para el ingreso y posterior consulta de datos. Otro sistema, en este caso una aplicación Android podrá acceder a dicha información usando estándares para la transferencia de información.	
Observaciones: La seguridad es un aspecto muy importante por eso se resalta la necesidad de usar técnicas y métodos de seguridad informática.	

Tabla 4.5: Historia de usuario 5 - Desarrollo de Web API

Elaborado por: Investigador

4.2.1.2. Aplicación móvil

A continuación se detallan las historias de usuario desarrolladas conjuntamente con un representante de la institución para el desarrollo de la aplicación móvil.

Historia de Usuario	
Número: 6	Nombre: Acceso a la aplicación móvil
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 2
Prioridad de negocio (Alta/Media/Baja): Alta	Puntos Estimados: 7.5
Riesgo de Desarrollo (Alta/Media/Baja): Alta	Puntos Reales: 7.0
Programador responsable: Luis Marcalla	
Descripción: Los estudiantes matriculados y debidamente registrados en la aplicación web podrán acceder a la aplicación con su usuario y contraseña, donde tendrán alcance a su histórico de notas, notificaciones por parte del personal de la institución y a un "muro" de noticias.	
Observaciones: Para el nombre de usuario y contraseña se podrá utilizar el correo electrónico y el número de cedula, la contraseña podrá ser modificada posteriormente.	

Tabla 4.6: Historia de usuario 6 - Acceso a la aplicación

Elaborado por: Investigador

Historia de Usuario	
Número: 7	Nombre: Implementación de menú
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 2
Prioridad de negocio (Alta/Media/Baja): Alta	Puntos Estimados: 6.5
Riesgo de Desarrollo (Alta/Media/Baja): Media	Puntos Reales: 7.0
Programador responsable: Luis Marcalla	
Descripción: Los estudiantes una vez dentro de la aplicación móvil dispondrán de una navegación entendible y fácil de usar, una barra de botones propio de los nuevos paradigmas de diseño móvil, accederán a todas las posibles pantallas de la aplicación con dicha herramienta.	
Observaciones: En el caso de inserción de imágenes estos deberán de representar o simbolizar perfectamente el contexto asociado.	

Tabla 4.7: Historia de usuario 7 - Implementación de menú

Elaborado por: Investigador

Historia de Usuario	
Número: 8	Nombre: Información de escuela
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 2
Prioridad de negocio: Media (Alta/Media/Baja)	Puntos Estimados: 5.0
Riesgo de Desarrollo: Media (Alta/Media/Baja)	Puntos Reales: 4.0
Programador responsable: Luis Marcalla	
Descripción: El módulo escuela mostrará una pequeña introducción del aspecto ideológico de la institución como por ejemplo misión y visión.	
Observaciones:	

Tabla 4.8: Historia de usuario 8 - Escuela

Elaborado por: Investigador

Historia de Usuario	
Número: 9	Nombre: Ver datos personales
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 3
Prioridad de negocio: Media (Alta/Media/Baja)	Puntos Estimados: 6.0
Riesgo de Desarrollo: Media (Alta/Media/Baja)	Puntos Reales: 6.5
Programador responsable: Luis Marcalla	
Descripción: La opción 2 o "Perfil Académico" mostrará datos informativos del estudiante, haciéndole saber que su ingreso a la aplicación fue exitoso.	
Observaciones: La información mostrada puede contener nombres, dirección, paralelo y el tipo de licencia a obtener del estudiante. En esta pantalla es pertinente disponer de un menú con la opción de cerrar sesión.	

Tabla 4.9: Historia de usuario 9 - Perfil académico

Elaborado por: Investigador

Historia de Usuario	
Número: 10	Nombre: Ver historial académico
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 3
Prioridad de negocio (Alta/Media/Baja): Alta	Puntos Estimados: 8.0
Riesgo de Desarrollo (Alta/Media/Baja): Alta	Puntos Reales: 9.0
Programador responsable: Luis Marcalla	
Descripción: "Académico" visualizará en un tabla todos los módulos que recibe el estudiante correspondiente a la categoría de licencia que el opto y sus respectivos puntajes alcanzados.	
Observaciones: La opción no. 3 debe dar a cada estudiante una idea clara de las materias que van a ser impartidas en clases y prepararlo a la vez.	

Tabla 4.10: Historia de usuario 10 - Académico

Elaborado por: Investigador

Historia de Usuario	
Número: 11	Nombre: Ver notificaciones
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 3
Prioridad de negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 7.0
Riesgo de Desarrollo: Media (Alta/Media/Baja)	Puntos Reales: 7.0
Programador responsable: Luis Marcalla	
Descripción: "Administrativo" desplegará una lista de notificaciones pudiendo ser del Inspector o Secretaria donde se le informará al estudiante las dificultades que se presente en cada uno de sus respectivos departamentos.	
Observaciones: Las notificaciones deberán de tener una fecha de vencimiento y así evitar el constante envío de una notificación ya recibida, optimizando valiosamente los recursos con el que dispone la plataforma web.	

Tabla 4.11: Historia de usuario 11 - Administrativo

Elaborado por: Investigador

Historia de Usuario	
Número: 12	Nombre: Ver eventos
Usuario: Estudiante	
Modificación de historia número: 1	Iteración Asignada: 3
Prioridad de negocio: Alta (Alta/Media/Baja)	Puntos Estimados: 7.0
Riesgo de Desarrollo: Media (Alta/Media/Baja)	Puntos Reales: 7.0
Programador responsable: Luis Marcalla	
Descripción: La opción 5 o "Eventos" será destinada a la publicación de noticias, eventos, etc. La publicación de un nuevo evento se encargara el usuario Administrador por medio de la aplicación web.	
Observaciones: Para la subida de nuevos eventos los campos para esto deberán de tener sus respectivas validaciones.	

Tabla 4.12: Historia de usuario 12 - Eventos

Elaborado por: Investigador

4.2.2. IDE's de desarrollo

Como en la fase de exploración de la metodología XP se establece las herramientas a usarse en el transcurso de todo el proyecto, se procede a hacer un resumen de las IDE's (Integrated Development Environment) seleccionadas y el porqué de su elección.

Para el desarrollo de la aplicación móvil se utilizará Android Studio en su versión 2.3, esta IDE propia para crear aplicaciones Android lanzada por Google para los desarrolladores de aplicaciones móviles brinda soporte y actualizaciones constantes para todo tipo de dispositivo móvil donde se ejecute Android como sistema operativo.

Para el panel de administración se eligió usar el paquete de .NET para la creación de aplicaciones web o aplicaciones ASP.NET debido a que permite crear software con lenguajes de alto nivel como C#, nos provee de ADO.NET(ActiveX Data Objects) y LINQ que son componentes de la plataforma para agregar capacidades de consulta de manera nativa en los lenguajes .NET.

En vista de que se eligió desarrollar una aplicación ASP.NET se creyó necesario seleccionar SQL Server para la base de datos debido a que tanto Visual Studio y SQL Server pertenecen a la misma empresa, esto contribuirá a la compatibilidad y mejorará el rendimiento de ambas partes.

A continuación un cuadro resumen de lo anteriormente dicho:

Software	IDE	Lenguaje de programación
Aplicación Móvil	Android Studio 2.3	Java
Aplicación Web (Administración)	Microsoft Visual Studio Enterprise 2015 Versión 14.0.2 - ASP.NET Framework 5	C#, LINQ
Motor de base de datos	Microsoft Sql Server 2012 Management Studio	Transact-SQL

4.2.3. Diseño de aplicación móvil

Para el diseño de la aplicación móvil para la plataforma Android se utilizó el nuevo diseño de Android 5.0 (Lollipop), el cual es Material Design o Diseño Material, este nuevo paradigma de diseño es un concepto, una filosofía, unas pautas enfocadas al diseño, pero también en la web y en cualquier plataforma.

El diseño material recibe su nombre por estar basado en objetos materiales, piezas colocadas en un espacio (lugar) y con un tiempo (movimiento) determinado. Es un diseño donde la profundidad, las superficies, los bordes, las sombras y los colores juegan un papel principal. Los elementos clave es la luz y las sombras, una iluminación realista proporciona indicios de cómo se comportará un elemento y en qué nivel se encuentra.

En el diseño de la aplicación se utilizaron métricas de diseño de material design, y además objetos propios de este nuevo patrón de diseño. El objeto más notable de este patrón en la aplicación es el Buttonbar, se trata de una barra de botones que sirve para la navegabilidad en la aplicación. Los CardView muestran la información en forma de tarjetas de una manera elegante para el usuario, este elemento fue usado en el módulo de eventos.

La maquetación del login, las distintas pantallas y la barra de menú de la aplicación Android se muestran a continuación.

Acceso a la aplicación móvil



Figura 4.1: Diseño móvil - Acceso a la aplicación móvil
Elaborado por: Investigador

Módulo Escuela



Figura 4.2: Diseño móvil - Módulo escuela
Elaborado por: Investigador

Módulo Perfil



Figura 4.3: Diseño móvil - Módulo perfil académico
Elaborado por: Investigador

Módulo Académico



Figura 4.4: Diseño móvil - Módulo académico
Elaborado por: Investigador

Módulo Administrativo



Figura 4.5: Diseño móvil- Módulo administrativo
Elaborado por: Investigador

Módulo Eventos

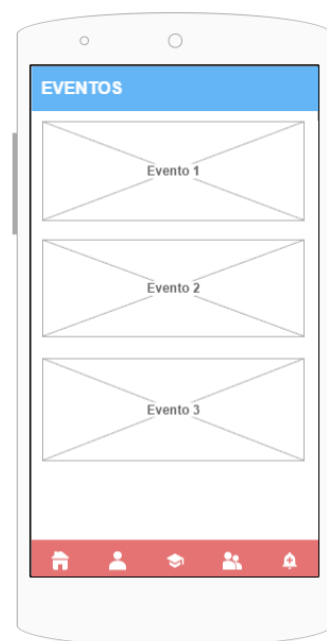


Figura 4.6: Diseño móvil - Módulo eventos
Elaborado por: Investigador

4.2.4. Diseño de aplicación web

Para la administración web se propuso usar el marco de trabajo Bootstrap para que la página se adaptara a cualquier dispositivo donde se mostrara, esto incluyendo PC's de escritorio, tabletas y celulares inteligentes. En vista de eso se procede a nombrar los procedimientos que se hicieron para que la aplicación web sea adaptativo.

Descarga de bibliotecas de Bootstrap

El primer paso fue la de obtener las bibliotecas de Bootstrap, para eso se dirigió al sitio oficial de Bootstrap para descargar los elementos necesarios. La versión de Bootstrap utilizada en el proyecto fue la 3.3.7.

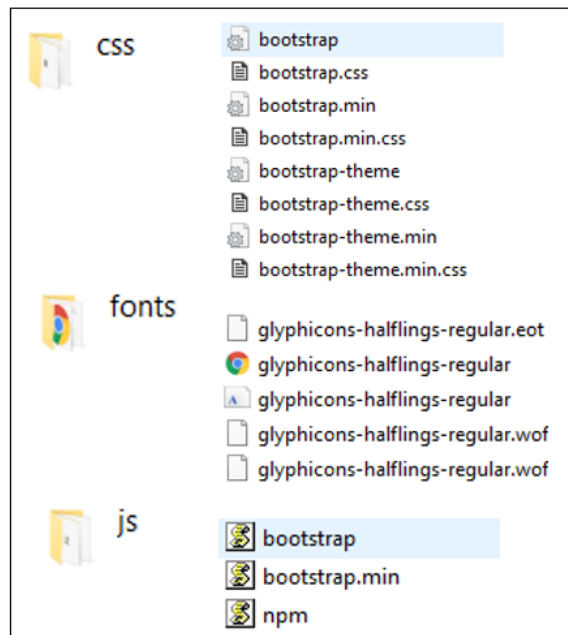


Figura 4.7: Bibliotecas de Bootstrap
Elaborado por: Investigador

Los archivos necesarios para la implementación de Bootstrap constan o están divididas en tres carpetas. La primera carpeta CSS contiene las hojas de estilos CSS adaptativos, dependiendo de las características del proyecto se puede usar la versión normal o la versión comprimida, la versión comprimida agiliza la carga de la pagina web por sus notables características. La segunda carpeta se puede apreciar fonts, esta contiene los iconos que se puede utilizar para que las pestañas se vean más atractivas al usuario final. La tercera carpeta contiene los archivos de javascript que junto con las hojas de estilo CSS adaptan los elementos de acuerdo al dispositivo donde se visualiza la página web.

Implementación de Bootstrap en solución de ASP.NET

Luego de obtener las bibliotecas necesarias fue necesario añadirlas a la solución de ASP.NET, entonces se procedió a crear las carpetas respectivas y copiar los archivos descargados.

Una vez que las bibliotecas de Bootstrap constaban en la solución de ASP.NET hace falta hacer la referencia necesaria, para quien ha trabajado con hojas de estilo CSS esto no presenta inconveniente alguno, así como la implementación de los componentes, debido a que si se desea usar un elemento solo se hace uso de una clase en específico. Por ejemplo si se piensa hacer uso de un "Jumbotron" se escribe la clase respectiva en un div, los div son elementos HTML y son esenciales para la esquematización de páginas web.

```
1 <html>
2 <head>
3 <!-- Referencias de las hojas de estilo de Bootstrap -->
4 <link href="~/Content/Bootstrap.css" rel="stylesheet" />
5 <link href="~/Content/Bootstrap.min.css" rel="stylesheet" />
6 <title>Administración web </title>
7 </head>
8 <body >
9 <h2>ESCUELA DE FORMACIÓN Y CAPACITACIÓN DE CONDUCTORES PROFESIONALES
    AMBATO </h2>
10 <!-- Jumbotron -->
11 <div class="jumbotron" style="background: url(/Content/Images/
    escu1_recorte.jpg)>
12 <!-- Imagen de la institución de capacitación -->
13 </div>
14 </body>
15 </html>
```

Componentes de Bootstrap en proyecto

En la documentación oficial de Bootstrap se muestran al menos 12 elementos que tiene este marco de trabajo para incorporar en páginas o aplicaciones web, en el presente trabajo no se utiliza todos, pero si los más representativos y populares. En la siguiente sección se enumera los componentes usados en la "vista" home de la aplicación web, esta pantalla dará la bienvenida a los usuarios validados anteriormente por la aplicación web.

1. Barra de título.- La barra de título muestra el nombre de la aplicación web. Al igual que todos los componentes de Bootstrap se adapta al tamaño de la pantalla donde se visualiza.
2. Navbar .- Es básicamente una barra de navegación que se va adaptando a la pantalla sin dejar de lado sus funciones y apariencia. Lo mejor de esta barra es de que cuando la pantalla es muy pequeña para mostrarse toda, esta se encoge y muestra un botón, que si se la presiona desplegará las pestañas en modo horizontal.
3. Jumbotron.- Es una forma de presentar un título o un contenido en forma muy destacada en una página web, para el presente trabajo se la utilizó para mostrar una imagen de la fachada de la institución.
4. Rejillas.- Bootstrap utiliza un sistema de rejillas para la adaptación de sus componentes, funciona así: se puede asignar un máximo de 12 rejillas por bloque, un bloque es toda una sección horizontal del documento, estas se van "acomodando" según el tamaño de pantalla. Para el ejemplo se observa que en la vista de PC los contenidos de la parte inferior forman dos columnas en un bloque, en este caso cada rejilla tiene el valor de 6, 6 más 6 forman las doce rejillas que cada bloque puede tener. En la vista de smartphone los elementos ya no se muestran en dos columnas, solo en una, esto sucede debido a que cada rejilla ahora tiene el valor de 12 y por ende ocupa toda la pantalla.

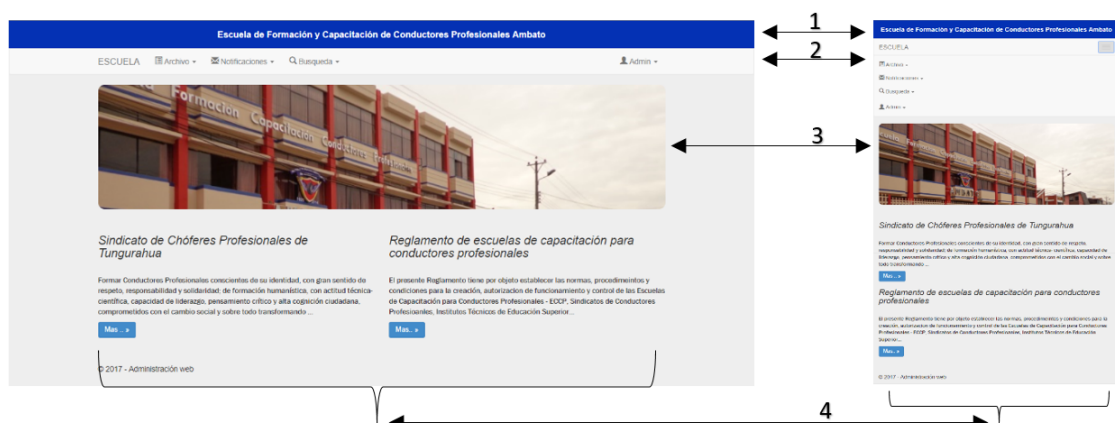


Figura 4.8: Diseño de aplicación web
Elaborado por: Investigador

4.3. Fase II: Planificación de la entrega

Valoración de Historias de Usuarios

A partir de las historias de usuarios se considera la valoración de las mismas especificando un tiempo estimado para la elaboración de cada una, las cuales están definidas en una semana de cinco días y un día de 4 horas.

Estimación de Historias de Usuarios

Para el desarrollo de la aplicación propuesta, se realizó una estimación del esfuerzo para cada una de las historias de usuario arrojando una estimación aproximada del tiempo de desarrollo.

No.	Historia de Usuario	Tiempo estimado		
		Semanas	Días	Horas
1	Acceso a sistema web	2	10	40
2	Mantenimiento de tablas	2	10	40
3	Ingreso de notas	1	5	20
4	Envío de notificaciones	1	5	20
5	Desarrollo de Web API's	2	10	40
6	Acceso a la aplicación móvil	1.5	8	32
7	Implementación de menú	1	5	20
8	Información de escuela	1	5	20
9	Ver datos personales	1.5	8	32
10	Ver historial académico	2	10	40
11	Ver notificaciones	1.5	8	32
12	Ver eventos	1.5	8	32
Tiempo estimado total		18	92	368

Tabla 4.13: Estimación del tiempo de desarrollo

Elaborado por: Investigador

Velocidad del proyecto

Para una mejor organización de tareas y trabajo se dispuso que por cada iteración se debe realizar 4 historias de usuario, como la fase de exploración arrojó unas doce historias de usuario el proyecto se fracciona en tres iteraciones; cada iteración producirá una parte del sistema operativa y funcional.

	Iteración 1	Iteración 2	Iteración 3
Horas	120	112	136
Semanas	6	5.5	6.5
Historias implementadas	1,2,3,4	5,6,7,8	9,10,11,12
Cantidad de historias	4	4	4

Tabla 4.14: Velocidad de proyecto

Elaborado por: Investigador

División en iteraciones

El proyecto esta dividido en 3 iteraciones, por consiguiente se obtuvo un total de tres entregas para las cuales se desarrollaron partes del proyecto completamente funcionales.

La primera iteración se refirió fundamentalmente al panel de administración web mientras que las demás iteraciones se relacionaron con el desarrollo de la aplicación Android.

4.4. Fase III: Iteraciones

Siguiendo las técnicas de la metodología XP se exponen las iteraciones en las que se dividió el proyecto, cada iteración consta con sus respectivas actividades anteriormente programadas. Al final de cada iteración se obtuvo un producto completamente funcional y de valor para la institución.

La primera iteración se enfocó en el desarrollo de la administración web y las dos iteraciones restantes en los Web API's y la aplicación móvil.

4.4.1. Iteración primera

En esta iteración se desarrolló la administración web que son las cuatro primeras historias de usuario descritas anteriormente.

En la siguiente página se muestran las actividades relacionados a esta iteración con los resultados respectivos.

Aplicación Web (Administración)

Actividades - Historia 1

Tarea	
Número: 1	Numero de historia: 1
Nombre de la tarea: Diseño del Modelo (Diagramación de la base de datos)	
Tipo de tarea: Diseño	Iteración Asignada: 1
Fecha inicio: 02/01/2017	Fecha fin: 06/01/2017
Programador responsable: Luis Marcalla	
Descripción: Se elaborará el diagrama de base de datos para luego poder generar los modelos (MVC), Visual Studio nos da una herramienta que nos permite generar nuestros modelos desde la base de datos lo que ayudará en la realización del proyecto.	

Tabla 4.15: Actividad 1 - Historia 1 - Acceso a la administración web

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 1
Nombre de la tarea: Diseño de la interfaz para acceso al sistema	
Tipo de tarea: Diseño	Iteración Asignada: 1
Fecha inicio: 09/01/2017	Fecha fin: 09/01/2017
Programador responsable: Luis Marcalla	
Descripción: Se mostrará un formulario para ingresar usuario y contraseña, los colores a usar son los que acostumbra usar la institución en sus logos, comunicados y demás. El acceso al sistema deber usar librerías de bootstrap para que sea responsive.	

Tabla 4.16: Actividad 2 - Historia 1 - Acceso a la administración web

Elaborado por: Investigador

Siguiendo con las actividades de la historia número 1 a continuación.

Tarea	
Número: 3	Numero de historia: 1
Nombre de la tarea: Codificación de los controladores para cuentas de usuario	
Tipo de tarea: Desarrollo	Iteración Asignada: 1
Fecha inicio: 10/01/2017	Fecha fin: 11/01/2017
Programador responsable: Luis Marcalla	
Descripción: Bajo el paradigma de programación utilizado en el presente proyecto (MVC) es necesario codificar un Controlador cuenta con sus respectivas acciones que ayudaran a validar al usuario dentro de la aplicación web	

Tabla 4.17: Actividad 3 - Historia 1 - Acceso a la administración web

Elaborado por: Investigador

Tarea	
Número: 4	Numero de historia: 1
Nombre de la tarea: Diseño de plantilla bootstrap	
Tipo de tarea: Diseño	Iteración Asignada: 1
Fecha inicio: 12/01/2017	Fecha fin: 13/01/2017
Programador responsable: Luis Marcalla	
Descripción: La plantilla deberá ser responsive design, adaptable a cualquier dispositivo donde se ejecute. La plantilla será utilizada en todo momento, solamente actualizará su interior para un mejor rendimiento para lo cual es indispensable el uso de Ajax.	

Tabla 4.18: Actividad 4 - Historia 1 - Acceso a la administración web

Elaborado por: Investigador

A continuación las actividades de la historia número 2.

Actividades - Historia 2

Tarea	
Número: 1	Numero de historia: 2
Nombre de la tarea: Uso de Scaffolding en los modelos	
Tipo de tarea: Desarrollo	Iteración Asignada: 1
Fecha inicio: 16/01/2017	Fecha fin: 20/01/2017
Programador responsable: Luis Marcalla	
Descripción: A partir de este proceso que proporciona Visual Studio se generan los controladores y las vistas de los modelos de la base de datos, para posteriormente realizar las operaciones de select, insert, update y delete dentro de la base.	

Tabla 4.19: Actividad 1 - Historia 2 - Mantenimiento de datos

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 2
Nombre de la tarea: Rediseño de vistas.	
Tipo de tarea: Desarrollo	Iteración Asignada: 1
Fecha inicio: 23/01/2017	Fecha fin: 27/01/2017
Programador responsable: Luis Marcalla	
Descripción: Las vistas deberán de ser reestructuradas para una mejor experiencia de usuario, el uso de clases y efectos css son importantes en todas las vistas generadas anteriormente.	

Tabla 4.20: Actividad 2 - Historia 2 - Mantenimiento de datos

Elaborado por: Investigador

A continuación las actividades de la historia número 3.

Actividades - Historia 3

Tarea	
Número: 1	Numero de historia: 3
Nombre de la tarea: Controladores para Academy	
Tipo de tarea: Desarrollo	Iteración Asignada: 1
Fecha inicio: 30/01/2017	Fecha fin: 03/02/2017
Programador responsable: Luis Marcalla	
Descripción: Los controladores estarán encargados de validar al usuario Docente, mostrar los alumnos por paralelo que tiene a su disposición para el posterior ingreso de notas, los cuales van a ser enviados a la aplicación móvil por medio de las Web API.	

Tabla 4.21: Actividad 1 - Historia 3 - Ingreso de notas

Elaborado por: Investigador

Actividades - Historia 4

Tarea	
Número: 1	Numero de historia: 4
Nombre de la tarea: Codificación de Controladores para notificaciones	
Tipo de tarea: Desarrollo	Iteración Asignada: 1
Fecha inicio: 06/02/2017	Fecha fin: 10/02/2017
Programador responsable: Luis Marcalla	
Descripción: Estos controladores servirán a los usuarios Inspector y Secretaria, con estos serán capaces de enviar mensajes personalizados a cualquier estudiante ya registrado.	

Tabla 4.22: Actividad 1 - Historia 4 - Envío de notificaciones

Elaborado por: Investigador

Resultados

La primera tarea realizada fue la diagramación de la base de datos a implementarse, y de allí se generarían los modelos respectivos para el funcionamiento del patrón de arquitectura MVC.

A continuación se presentan los resultados de la primera iteración.

- Diagrama Entidad-Relación implementada (SQL Server). Tarea 1 de historia 1.

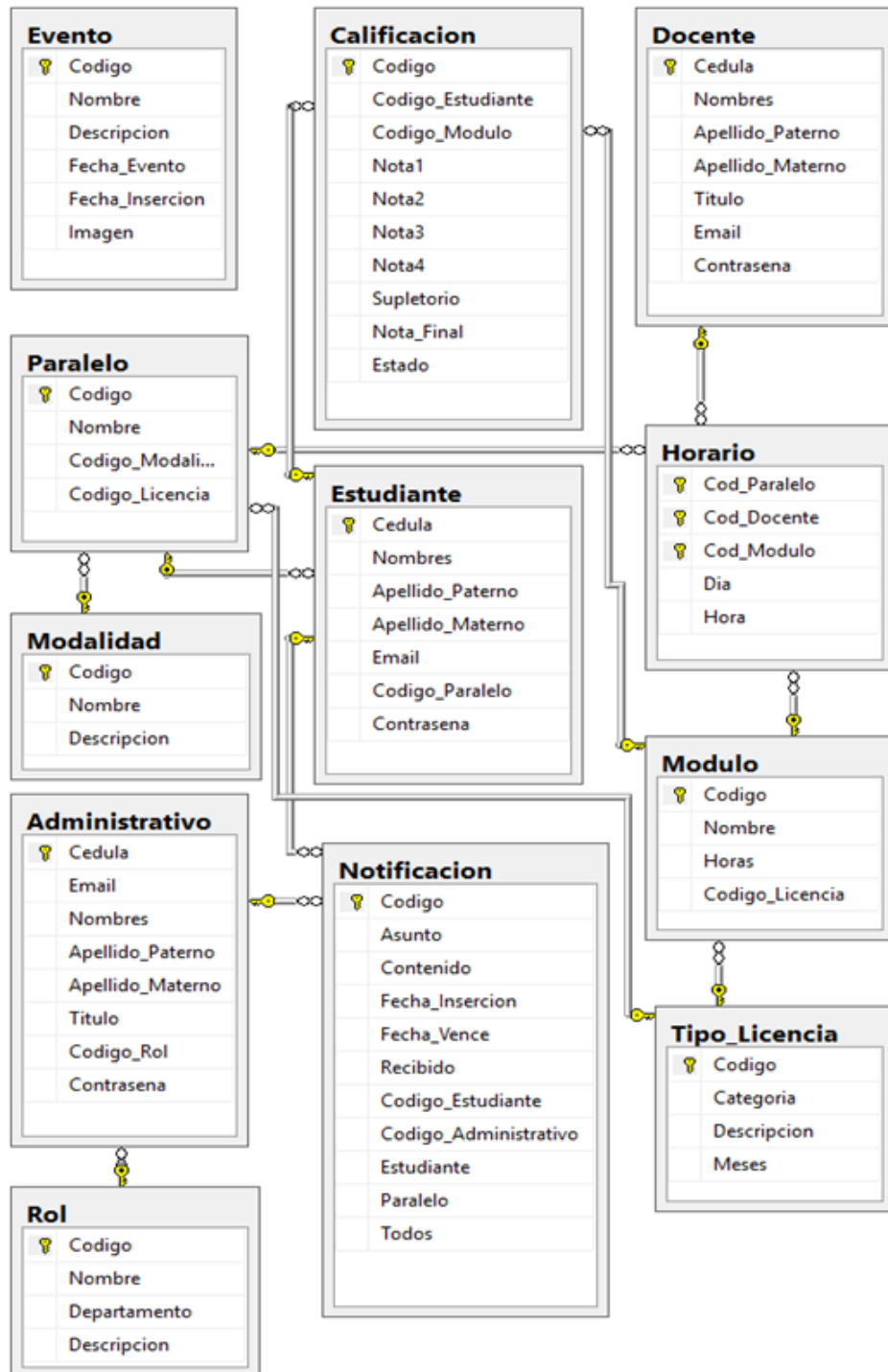


Figura 4.9: Diagrama Entidad-Relacion de base de datos
Elaborado por: Investigador

- Login de la aplicación web (ASP.NET). Tarea 2 de historia 1.

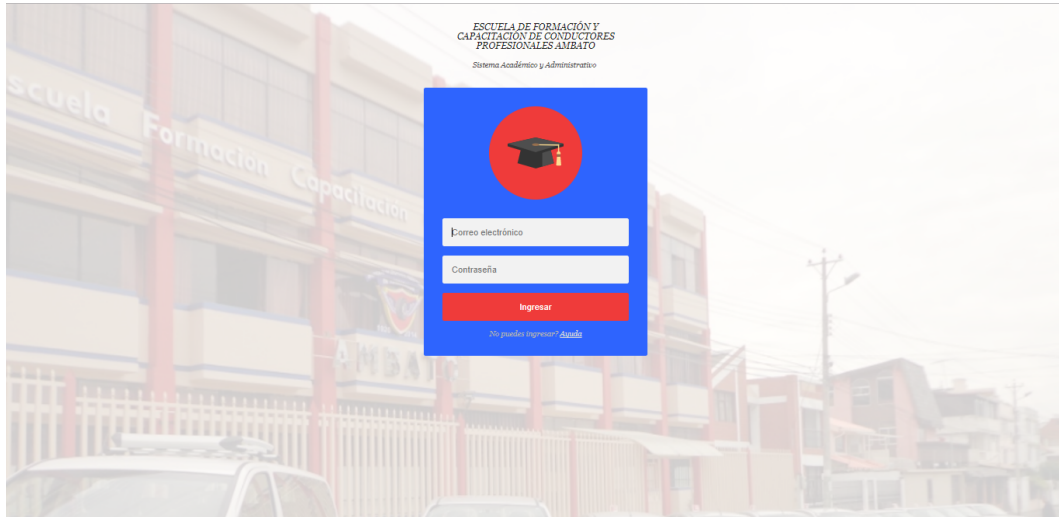


Figura 4.10: Login de aplicación web - Vista en PC
Elaborado por: Investigador

Los controladores equivalen a la lógica de negocio, se encargan de ejecutar las operaciones del usuario con ayuda de las vistas y los modelos. El controlador encargado de identificar al tipo de usuario que ingresa a la aplicación web utilizando C# como lenguaje de programación a continuación.

- Controlador para autenticación de usuarios usando C# (ASP.NET). Tarea 3 de historia 1.

```

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.Mvc;
6 using EscuelaConduccion.Models;
7 using EscuelaConduccion.Helper;
8 using EscuelaConduccion.Controllers.SysAdmin;
9 namespace Escuela.Controllers {
10 public class LoginController: Controller {
11     public ActionResult Index() {
12         if (!(Session["Rol"] == null)) {
13             return RedirectToAction("Index", "Home");
14             return View();
15         }
16
17     public ActionResult Salida()

```

```

18     {Session.Remove("Rol");
19     return RedirectToAction("Index", "Home");}
20
21     // POST: /Ingreso/
22     [HttpPost, ActionName("Ingresar")]
23     public ActionResult Inicio(User usuario){
24     AunenticationHelper obj = new AunenticationHelper();
25     if (obj.FindAdministartivo(usuario) != null){
26     var administrativo = obj.FindAdministartivo(usuario);
27     SessionAdministrativoHelper SessionAdministrativo = new
        SessionAdministrativoHelper(administrativo);
28     return RedirectToAction("Index", "Home");} else if (obj.UsuarioDocente
        (usuario) == true){
29     var Docente = obj.BuscarDocente(usuario.Cedula);
30     SessionDocenteHelper objSesDoc = new SessionDocenteHelper(Docente);

        DocenteHelper objDocente = new DocenteHelper();
31     AuxIdentidad.Paralelos = objDocente.CodigosParalelos(Docente);
32     SessionParaleloDocente objParalelo = new SessionParaleloDocente(
        AuxIdentidad.Paralelos);
33     return RedirectToAction("Index", "Home");}
34     else {
35     ViewData["MensajeError"] = "Usuario no encontrado, verifique sus datos
        !";
36     return View("Index");}
37     }
38     }

```

Al igual que el login de aplicación web, el home de la aplicación hace uso estilos CSS de Bootstrap para adaptarse en la pantalla que se muestra. Una breve descripción de la estructura de esta pantalla y de sus componentes Bootstrap a continuación. En la parte superior se encuentra la barra de título con el nombre de la institución, debajo de la barra de título se encuentra el Navigation Bar, este elemento tiene la facultar de contraerse en dispositivos móviles como se muestra en la figura 4.15 y al igual que en la vista de PC despliga las pestañas disponibles para el usuario sin problemas. El Dropdows tiene la capacidad de enlistar y organizar un grupo de pestañas cuando el espacio se ve reducido, como por ejemplo la pestañá matenimiento mantenimiento agrupa todas las entidades de base de datos. El Jumbotron es un componente ligero y flexible que opcionalmente puede extender toda la ventana de visualización para mostrar el contenido clave del sitio, en este caso contiene la localidad de la institución.

■ Home de la aplicación web (ASP.NET). Tarea 4 de historia 1.



Figura 4.11: Home de aplicación web - Vista en PC
Elaborado por: Investigador

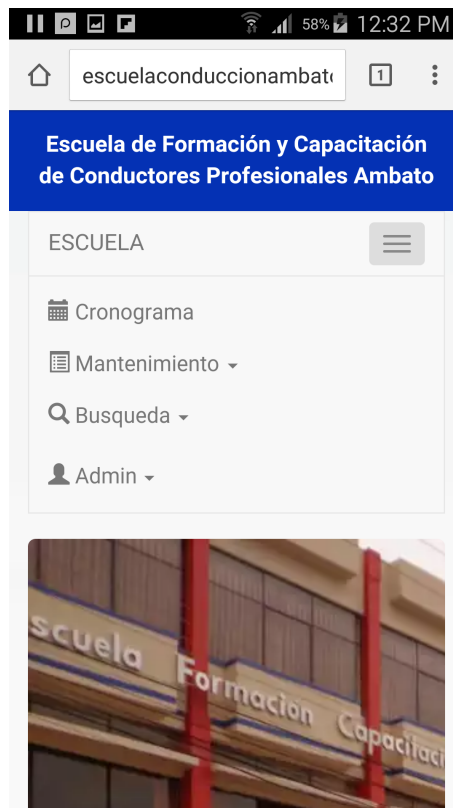


Figura 4.12: Home de aplicación web - Vista en smartphone
Elaborado por: Investigador

El modelo, parte del patrón de diseño de software utilizado en el presente proyecto y el medio para acceder a una entidad de la base de datos cuando requiramos realizar una operación de los métodos CRUD (create, read, update, delete) en la base de datos. ASP.NET MVC nos facilita el trabajo al generar nuestros modelos a partir de la base de datos (SQL Server). El modelo estudiante a continuación.

- **Modelo generado a partir de la entidad Estudiante (ASP.NET).**

Tarea 1 de historia 2.

```

1 namespace EscuelaConduccion.Models {
2     using System;
3     using System.Collections.Generic;
4     using System.ComponentModel.DataAnnotations;
5     using System.ComponentModel.DataAnnotations.Schema;
6
7     public partial class Estudiante{
8         public Estudiante(){
9             this.Calificacion = new HashSet<Calificacion>();
10            this.Notificacion = new HashSet<Notificacion>(); }
11
12            //Cedula
13            [Key]
14            [StringLength(10,ErrorMessage="Cedula no válida",MinimumLength=10)]
15            [Required(ErrorMessage = "Campo obligatorio")]
16            [Display(Name="Cedula")]
17            public string Cedula{ get; set; }
18
19            //Nombres
20            [Required(ErrorMessage = "Campo obligatorio")]
21            [Display(Name = "Nombres")]
22            [StringLength(20, ErrorMessage = "No. de caracteres excedidos")]
23            public string Nombres { get; set; }
24
25            //Apellido paterno
26            [Required(ErrorMessage = "Campo obligatorio")]
27            [Display(Name = "Apellido Paterno")]
28            [StringLength(20, ErrorMessage = "El Apellido Materno no puede ser
                mayor de {0} caracteres.")]
                public
                string Apellido_Paterno { get; set; }
29
30            //Apellido materno
31            [StringLength(20,ErrorMessage="El Apellido Materno no puede ser mayor
                de {0} caracteres.",MinimumLength = 1)]
32            [Required(ErrorMessage = "Campo obligatorio")]
33            [Display(Name = "Apellido Materno")]

```

```

34  public string Apellido_Materno { get; set; }
35
36  //Email
37  [Required(ErrorMessage = "Campo obligatorio")]
38  [DataType(DataType.EmailAddress)]
39  [Display(Name = "Email")]
40  public string Email { get; set; }
41
42  //Codigo del paralelo
43  [Required(ErrorMessage = "Campo obligatorio")]
44  [Range(1, 100, ErrorMessage = "Seleccione un paralelo")]
45  [Display(Name = "Paralelo")]
46  public int Codigo_Paralelo { get; set; }
47
48  //Nombre completo
49  [NotMapped]
50  [Display(Name = "Nombre")]
51  public string NombreCompleto{get{return string.Format("{0}{1}{2}",
    Apellido_Paterno ,Apellido_Materno ,Nombres);}}
52
53  public virtual ICollection<Calificacion> Calificacion { get; set; }
54  public virtual Paralelo Paralelo { get; set; }
55  public virtual ICollection<Notificacion> Notificacion { get; set; }
56  } }

```

- Código para el envío de notificaciones usando C# y LINQ en base de datos. Tarea 1 de historia 4.

```

1  public ActionResult NuevaNotificacion(string cedula){
2  if (cedula == null){
3  return new HttpStatusCodeResult(HttpStatusCode.BadRequest);}
4  var DatosEstudiante = objeEstudiante.profileStudentByCedula(cedula)
    ;
5  var Estudiante = db.Estudiante.Find(cedula);
6  objSesEst = new SessionEstudianteHelper(Estudiante);
7  string nombreParalelo = objSesEst.getParaleloEstudiante();
8  Paralelo ParaleloEstudiante =
9  db.Paralelo.Where(i => i.Nombre==
    nombreParalelo).First();
    var NotificacionesPorEstudiantes =
10  (from notificationDb in db.Notificacion
11  where notificationDb.Codigo_Estudiante
    == cedula

```

```

12         select notificationDb).OrderByDescending(x=>x.
           Fecha_Insercion);
13     cedula_estudiante = cedula;
14     return View(NotificacionesPorEstudiantes);
15 }

```

La estructura de la solución de la aplicación web es fácil de entender si se tiene claro el patrón de arquitectura de software MVC, en el directorio de la solución se observar la división de los componentes de acuerdo a su tipo y función que desempeña. La carpeta models contiene el modelo generados con la ayuda de ADO.NET, cada modelo es una clase de un objeto o entidad de la base de datos. La carpeta controllers contienen los métodos de acción (read, insert, delete, update) de cada uno de los modelos. La carpeta views contiene las pantallas que observa el usuario cuando navega por la página, los views contiene HTML, estilos Bootstrap y también javascript.

- **Captura de solución (ASP.NET).**

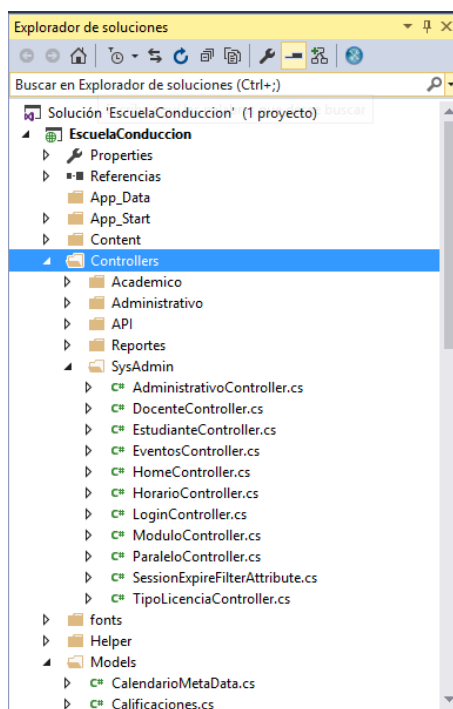


Figura 4.13: Directorio de controllers - Aplicación web
Elaborado por: Investigador

4.4.2. Iteración segunda

En esta iteración se desarrollo y probaron los Web API's junto con los primeros módulos de la aplicación.

Aplicación Web (Web API)

Actividades - Historia 5

Tarea	
Número: 1	Numero de historia: 5
Nombre de la tarea: Desarrollo de web api autenticación	
Tipo de tarea: Desarrollo	Iteración Asignada: 2
Fecha inicio: 13/02/2017	Fecha fin: 15/02/2017
Programador responsable: Luis Marcalla	
Descripción: Este web api está encargado de autenticar a un usuario como es un estudiante. Cuando no es exitoso devuelve false, caso contrario responde con un objeto donde incluirá todos los datos del estudiante dentro de la institución (datos personales, paralelo, licencia). Debido a la naturaleza de la consulta, esta web api utilizara el método post, un verbo de HTTP.	

Tabla 4.23: Actividad 1 - Historia 5 - Desarrollo de Web API's

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 5
Nombre de la tarea: Desarrollo de web api académico	
Tipo de tarea: Desarrollo	Iteración Asignada: 2
Fecha inicio: 16/02/2017	Fecha fin: 17/02/2017
Programador responsable: Luis Marcalla	
Descripción: Una vez el estudiante este autenticado puede acceder a siguientes web api como este, el web api académico recibe como parámetro la cedula del estudiante y devuelve un query con las materias y notas dependiendo del paralelo y la correspondiente categoría de licencia. Para las respectivas consultas se usa linq propio de MVC y c# dentro de .net	

Tabla 4.24: Actividad 2 - Historia 5 - Desarrollo de Web API's

Elaborado por: Investigador

Tarea	
Número: 3	Numero de historia: 5
Nombre de la tarea: Desarrollo de web api administrativo	
Tipo de tarea: Desarrollo	Iteración Asignada: 2
Fecha inicio: 20/02/2017	Fecha fin: 22/02/2017
Programador responsable: Luis Marcalla	
Descripción: Dicho web api recibe un parámetro como es la cedula del estudiante y envía como resultado un query donde contendrá todos los mensajes enviados por parte de los usuarios Inspector y Secretaria. También la consulta será desarrollada en linq que usa los modelos para hacer las consultas.	

Tabla 4.25: Actividad 3 - Historia 5 - Desarrollo de Web API's

Elaborado por: Investigador

Tarea	
Número: 4	Numero de historia: 5
Nombre de la tarea: Desarrollo de web api Eventos	
Tipo de tarea: Desarrollo	Iteración Asignada: 2
Fecha inicio: 23/02/2017	Fecha fin: 24/02/2017
Programador responsable: Luis Marcalla	
Descripción: Esta Web Api por ser general no necesita recibir parámetros, todos los estudiantes podrán recibir una respuesta exitosa cuando soliciten información a este web api. El verbo correspondiente de http es el get	

Tabla 4.26: Actividad 4 - Historia 5 - Desarrollo de Web API's

Elaborado por: Investigador

Las actividades relacionadas al desarrollo de la aplicación móvil a continuación.

Aplicación Móvil

Actividades - Historia 6

Tarea	
Número: 1	Numero de historia: 6
Nombre de la tarea: Diseño de layout de login	
Tipo de tarea: Diseño	Iteración Asignada: 2
Fecha inicio: 27/02/2017	Fecha fin: 28/02/2017
Programador responsable: Luis Marcalla	
Descripción: Toda la aplicación será basada en Material Design, se deberá tener en cuenta las técnicas y parámetros de este nuevo paradigma al momento de diseñar cada layout. El login contendrá los colores característicos de la escuela de formación y el sello de la institución en la parte superior.	

Tabla 4.27: Actividad 1 - Historia 6 - Desarrollo de Web API's

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 6
Nombre de la tarea: Desarrollo de la clase login	
Tipo de tarea: Desarrollo	Iteración Asignada: 2
Fecha inicio: 01/03/2017	Fecha fin: 02/03/2017
Programador responsable: Luis Marcalla	
Descripción: La parte lógica, será encargada de validar al usuario estudiante y redirigirlo hacia el menú principal; retrofit una librería que realiza las consultas al web api por medio del post solicitará información de que si el estudiante está registrado o no. Si la respuesta es exitosa la clase login recibirá un objeto con los datos del estudiante.	

Tabla 4.28: Actividad 2 - Historia 6 - Desarrollo de Web API's

Elaborado por: Investigador

Tarea	
Número: 3	Numero de historia: 6
Nombre de la tarea: Desarrollo de la clase preferences	
Tipo de tarea: Desarrollo	Iteración Asignada: 2
Fecha inicio: 03/03/2017	Fecha fin: 06/03/2017
Programador responsable: Luis Marcalla	
Descripción: Esta clase actuara inmediateamente después de la clase login quien después de autenticar al estudiante guardara el perfil académico del estudiante para una posterior consulta si la necesidad de realizar un post al web api.	

Tabla 4.29: Actividad 3 - Historia 6 - Desarrollo de Web API's

Elaborado por: Investigador

Tarea	
Número: 4	Numero de historia: 6
Nombre de la tarea: Desarrollo de la clase preferences	
Tipo de tarea: Desarrollo	Iteración Asignada: 2
Fecha inicio: 07/03/2017	Fecha fin: 08/03/2017
Programador responsable: Luis Marcalla	
Descripción: Esta clase actuara inmediateamente después de la clase login quien después de autenticar al estudiante guardara el perfil académico del estudiante para una posterior consulta si la necesidad de realizar un post al web api.	

Tabla 4.30: Actividad 4 - Historia 6 - Desarrollo de Web API's

Elaborado por: Investigador

Las actividades de la historia número 7 a continuación.

Actividades - Historia 7

Tarea	
Número: 1	Numero de historia: 7
Nombre de la tarea: Diseño de la barra de botones	
Tipo de tarea: Diseño	Iteración Asignada: 2
Fecha inicio: 13/03/2017	Fecha fin: 15/03/2017
Programador responsable: Luis Marcalla	
Descripción: Para el diseño de la barra de botón se debe tener en cuenta las métricas y técnicas de Material Design para una mejor visualización	

Tabla 4.31: Actividad 1 - Historia 7 - Implementación de menú

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 7
Nombre de la tarea: Fragment para la barra de botones	
Tipo de tarea: Diseño/Desarrollo	Iteración Asignada: 2
Fecha inicio: 16/03/2017	Fecha fin: 17/03/2017
Programador responsable: Luis Marcalla	
Descripción: Para el funcionamiento de la barra de botones se desarrolla las clases necesarias, importación de imágenes y fragment necesarios para redirigir al usuario a través de la aplicación.	

Tabla 4.32: Actividad 2 - Historia 7 - Implementación de menú

Elaborado por: Investigador

La actividad de la historia número 8 a continuación.

Actividades - Historia 8

Tarea	
Número: 1	Numero de historia: 8
Nombre de la tarea: Diseño de layout escuela	
Tipo de tarea: Diseño	Iteración Asignada: 2
Fecha inicio: 20/03/2017	Fecha fin: 24/03/2017
Programador responsable: Luis Marcalla	
Descripción: Esta pantalla mostrara el aspecto ideológico de la institución educativa.	

Tabla 4.33: Actividad 1 - Historia 8 - Información de escuela

Elaborado por: Investigador

Resultados

A continuación se presentan los resultados de la segunda iteración.

Las Web API's elementales para la comunicación de la aplicación móvil con su panel de administración para la consulta de información. La Web API autenticación realiza la función de validar al estudiante dentro de la aplicación móvil cuando este envía sus credenciales a través del login, en caso de que el estudiante es identificado esta API responde con un objeto donde constan los datos personales del estudiante. La información devuelta se guarda en el smartphone por medio de uso de la *SharedPreferences* de Android para ser visualizados posteriormente en el módulo de perfil de la aplicación móvil. A continuación la Web API autenticación.

- **Código de Web API para la autenticación de estudiante usando C# y LINQ para consulta en base de datos (ASP.NET). Tarea 1 de historia 5.**

```
1 // POST: api/Autenticacion
2 [ResponseType(typeof(PerfilEstudiante))]
3 public PerfilEstudiante PostEstudiante(User user){
4     String paralelo;
5     EstudentHelper obj = new EstudentHelper();
6
```

```

7 var profilestudent = (from studentDb in db.Estudiante
8   from paraleloDb in db.Paralelo
9   from tipo_licenciaDb in db.Tipo_Licencia
10  where studentDb.Codigo_Paralelo == paraleloDb.Codigo
11  where paraleloDb.Codigo_Licencia == tipo_licenciaDb.Codigo
12  where studentDb.Cedula == user.Cedula
13  where studentDb.Email==user.Email
14  select new PerfilEstudiante{
15    Cedula = studentDb.Cedula ,
16    Nombres = studentDb.Nombres ,
17    Apellido_Paterno = studentDb.Apellido_Paterno ,
18    Apellido_Materno = studentDb.Apellido_Materno ,
19    Email = studentDb.Email ,
20    Paralelo = paraleloDb.Nombre ,
21    Tipo_Licencia = tipo_licenciaDb.Categoria ,
22    Modalidad = paraleloDb.Modalidad}).FirstOrDefault();
23 if (profilestudent == null){
24   return null; }
25 return profilestudent;
26 }

```

Para verificar la operación correcta de la anterior Web API se utilizó un cliente para web services basados en REST, Postman es una extensión de Chrome que nos ayuda en esta tarea.

A continuación una prueba exitosa del Web API de autenticación.

- **Prueba de Web API de autenticación en cliente REST (Postman).**
Tarea 1.1 de historia 5.

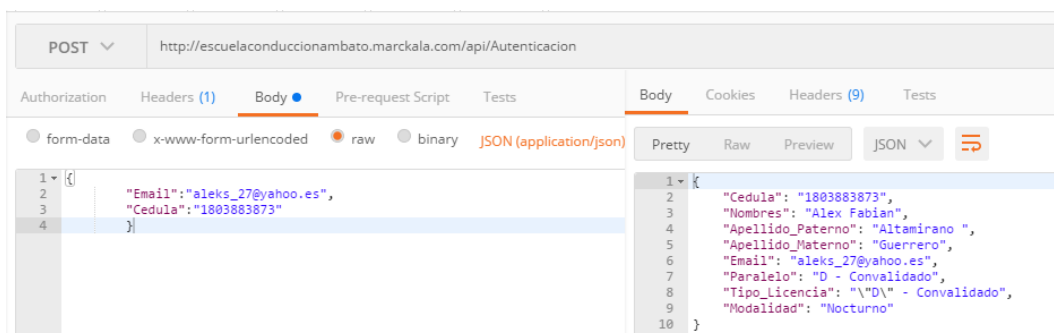


Figura 4.14: Web API autenticación - Postman
Elaborado por: Investigador

La aplicación móvil luego de que un estudiante se ha identificado, hace un request a la siguiente Web API que devuelve una lista con los módulos y sus notas respectivas.

La Web API como las demás hace uso de LINQ para realizar operaciones en la base de datos.

■ **Código de Web API para la consulta de calificaciones (ASP.NET).
Tarea 2 de historia 5.**

```
1 using System.Collections.Generic;
2 using System.Data;
3 using System.Data.Entity;
4 using System.Data.Entity.Infrastructure;
5 using System.Linq; using System.Net;
6 using System.Net.Http; using System.Web.Http;
7 using System.Web.Http.Description;
8 using EscuelaConduccion.Models;
9 namespace EscuelaConduccion.Controllers.API {
10 public class CalificacionController : ApiController {
11     private escuelaDBEntities db = new escuelaDBEntities();
12     [ResponseType(typeof(Calificaciones))]
13     public List<Calificaciones> GetCalificacion(int id){
14         var calificacion = (from calificaciondb in db.Calificacion
15                             where calificaciondb.Codigo_Estudiante == id.ToString()
16                                 select new Calificaciones{
17                                     NombreModulo = (from moduloDb in db.Modulo where
18                                                         moduloDb.Codigo ==
19                                                         calificaciondb.
20                                                         Codigo_Modulo select moduloDb.Nombre).FirstOrDefault
21                                                         (),
22                                     Nota1 = calificaciondb.Nota1.ToString(),
23                                     Nota2 = calificaciondb.Nota2.ToString(),
24                                     Nota3 = calificaciondb.Nota3.ToString(),
25                                     Nota4 = calificaciondb.Nota4.ToString(),
26                                     Promedio = calificaciondb.Promedio.ToString(),
27                                     Supletorio = calificaciondb.Supletorio.ToString(),
28                                     NotaFinal = calificaciondb.Nota_Final.ToString(),
29                                     Estado = calificaciondb.Estado }).ToList();
30         if (calificacion == null){
31             return null;
32         }
33         return calificacion;
34     }
35 }
```

Se define un layout como una estructura visual para una interfaz de usuario, es decir, aquello que hace de intermediario entre el terminal móvil y el usuario. Un elemento de una interfaz de usuario (widgets o layouts) se puede declarar desde un fichero XML o en tiempo de ejecución de la aplicación.

- **Layout de login de la aplicación móvil. Tarea 1 de historia 6.**

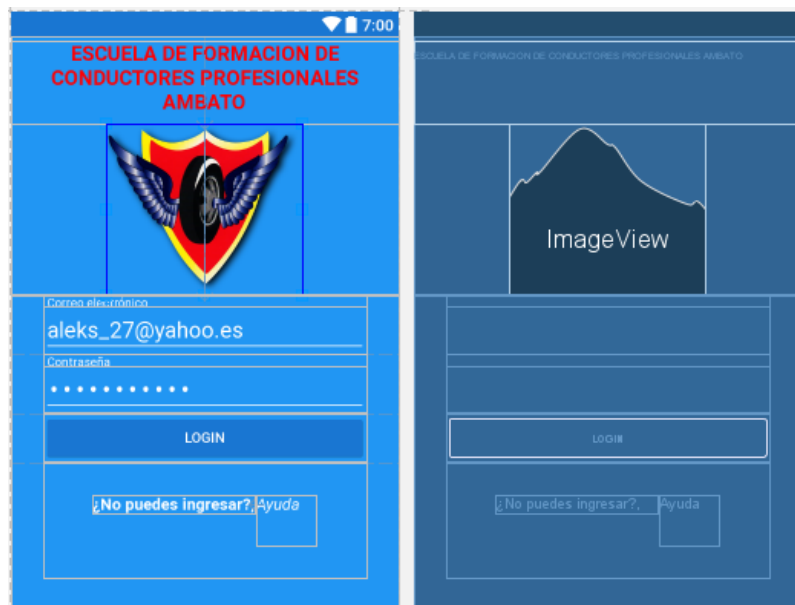


Figura 4.15: Diseño de login - Aplicación móvil
Elaborado por: Investigador

Las interfaces en java especifican métodos para ser implementadas por otras clases que las necesiten. Retrofit es un cliente REST para Java que usa los métodos HTTP (GET, POST, PUT). A continuación se muestra una interfaz hecha en java para Android que especifica métodos de petición a un Web API (web service basado en REST) usando librerías de Retrofit, dependiendo de su naturaleza hará esta solicitud usando un verbo o método del protocolo HTTP, GET o POST.

- **Interfaz implementada para uso de Retrofit (Android). Tarea 2 de historia 6.**

```

1 public interface autentificacion {
2
3     public static final String BASE_URL = "http://escuelaconduccionambato
        .marckala.com/";
4
5     @GET("api/Calificacion/{cedula}") Call<List<Calificaciones>>
        GetCalifications(@Path("cedula") String cedula);

```



```

6
7     @GET("api/Publicacion") Call<ArrayList<Calendario>> GetPublicacion();
8
9     @GET("api/Notificacion/{cedula}") Call<List<Notificaciones>>
        GetNotificaciones(@Path("cedula") String cedula);
10
11    @POST("api/Autenticacion") Call<PerfilEstudiante> login(@Body User user
        );
12
        }

```

Para hacer las respectivas peticiones a las Web API's se implementaron métodos propios de la librería de Retrofit junto con la ayuda de la interfaz anteriormente mostrada, cada método hace un "Call" o llamado a una lista u objeto para lo cual envía los parámetros respectivos. En el siguiente método se envían la cédula de identidad y el correo electrónico del estudiante y se le responde con un objeto que contiene los datos personales del estudiante cuando esta registrado en la aplicación web. Todo este intercambio de datos se lo realiza en formato JSON (JavaScript Object Notation) por la naturaleza de los web services.

■ Código en Retrofit para la autenticación de usuario en la aplicación móvil (Android).- Tarea 2.1 de historia 6.

```

1  import retrofit2.Call;
2  import retrofit2.Callback;
3  import retrofit2.Response;
4  import retrofit2.Retrofit;
5  import retrofit2.converter.gson.GsonConverterFactory;
6  import android.support.design.widget.TextInputLayout;
7  import android.support.v4.app.ActivityOptionsCompat;
8  import android.support.v7.app.AppCompatActivity;
9  import retrofit2.Call;
10 import retrofit2.Callback;
11 import retrofit2.Response;
12 import retrofit2.Retrofit;
13 import retrofit2.converter.gson.GsonConverterFactory;
14 import android.support.design.widget.TextInputLayout;
15 import android.support.v4.app.ActivityOptionsCompat;
16 import android.support.v7.app.AppCompatActivity;
17
18 Call<PerfilEstudiante> loginCall = Autenticacion.login(new User(email, password));
19 loginCall.enqueue(new Callback<PerfilEstudiante>() {
20 @Override
21 public void onResponse(Call<PerfilEstudiante> call, Response<PerfilEstudiante>
        response) { // Mostrar progreso
22 if(response.isSuccessful()){
23 if(!response.body()==null){
24 SessionPrefsProfile.getLoginActivity().this).saveStudent(response.body());
25 Intent intent = new Intent(getApplicationContext(), ContainerActivity.
        class); if (Build.VERSION.
        SDK_INT >= Build.VERSION_CODES.LOLLIPOP){

```

```

26         Slide slide = new Slide();
27         slide.setDuration(4000);
28         getWindow().setExitTransition(slide);           startActivity(
                intent, ActivityOptions.makeSceneTransitionAnimation(
                    LoginActivity.this).toBundle());       }else
29         {startActivity(intent);}
30         showProgress(false);
31     }else {
32         showProgress(false);
33     Toast.makeText(getApplicationContext(), "Estudiante no encontrado", Toast.
        LENGTH_LONG).show();                               }return;
34     }
35 }else{
36     String error = "Ha ocurrido un error. Contacte al administrador";
37     if (response.errorBody().contentType().subtype().equals("json")) {
        ApiError apiError =
38         ApiError.fromResponseBody(response.errorBody());
        error = apiError.getMessage();
39     }
40     else {
41         try {
42             Log.d("LoginActivity", response.errorBody().string())
43             ;
44         } catch (IOException e) {
45             e.printStackTrace(); }
46     }
47     showProgress(false);
48     showLoginError(error);
49     return;
50 }
51
52 @Override
53 public void onFailure(Call<PerfilEstudiante> call, Throwable t) {
54     showProgress(false);
55     Toast.makeText(getApplicationContext(), "Error en la consulta", Toast.LENGTH_LONG
56     ).show();

```

La aplicación ASP.NET, así como las Web API necesarias para la posterior consulta de la aplicación móvil se desarrollaron hasta la tarea número 4 de la historia de usuario 5. Para la prueba de las Web API's se utilizó un cliente REST como se mostró anteriormente.

Después de la finalización de las tareas de esta actividad se prosiguió a integrar los layouts , las clases de java y los métodos de la librería Retrofit. Arrojando como resultado los módulos siguientes.

- Módulos desarrollados de la aplicación móvil (Android).

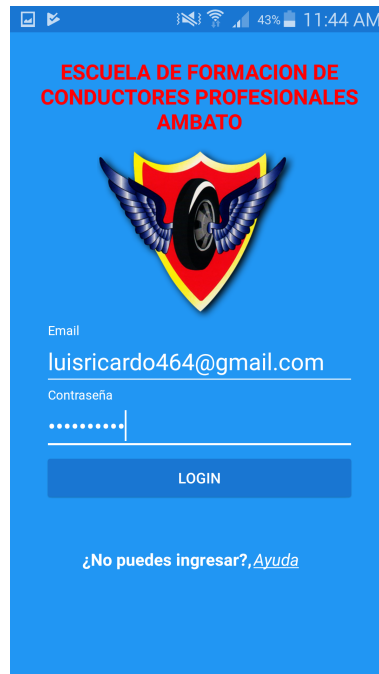


Figura 4.16: Acceso a la aplicación - Aplicación móvil
Elaborado por: Investigador

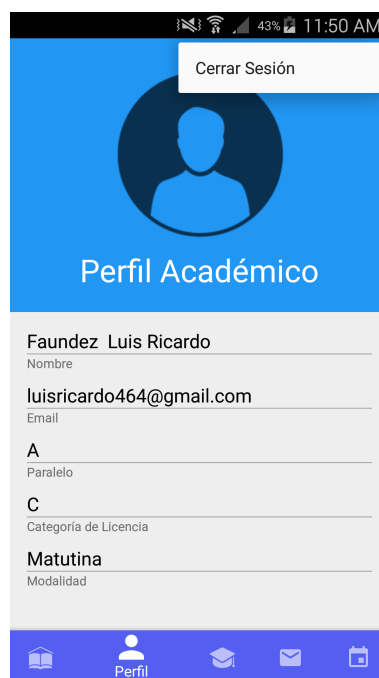


Figura 4.17: Ver datos personales - Aplicación móvil
Elaborado por: Investigador

4.4.3. Iteración tercera

En la tercera iteración se desarrollaron los módulos académico, administrativo y eventos de la aplicación móvil que corresponden a las siguientes actividades.

Actividades - Historia 9

Tarea	
Número: 1	Numero de historia: 9
Nombre de la tarea: Diseño de layout perfil académico	
Tipo de tarea: Diseño	Iteración Asignada: 3
Fecha inicio: 27/03/2017	Fecha fin: 31/03/2017
Programador responsable: Luis Marcalla	
Descripción: En esta pantalla mostrar datos informativos del estudiante ya iniciado sesión.	

Tabla 4.34: Actividad 1 - Historia 9 - Ver datos personales

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 9
Nombre de la tarea: Desarrollo de la clase perfil académico	
Tipo de tarea: Desarrollo	Iteración Asignada: 3
Fecha inicio: 03/04/2017	Fecha fin: 05/04/2017
Programador responsable: Luis Marcalla	
Descripción: La clase dispondrá de los métodos para a la clase de preferencias donde se guardaron los datos anteriormente en la tarea de autenticación del estudiante.	

Tabla 4.35: Actividad 2 - Historia 9 - Ver datos personales

Elaborado por: Investigador

Actividades - Historia 10

Tarea	
Número: 1	Numero de historia: 10
Nombre de la tarea: Diseño de layout académico	
Tipo de tarea: Diseño	Iteración Asignada: 3
Fecha inicio: 10/04/2017	Fecha fin: 14/04/2017
Programador responsable: Luis Marcalla	
Descripción: En esta pantalla se visualiza un cuadro de notas del estudiante con módulos y notas, debido a que el cuadro ocupara más espacio del disponibles es crucial usara un Horizontal Scroll para obtener una vista para los estudiantes.	

Tabla 4.36: Actividad 1 - Historia 10 - Ver historial académico

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 10
Nombre de la tarea: Desarrollo de clase académico	
Tipo de tarea: Diseño	Iteración Asignada: 3
Fecha inicio: 17/04/2017	Fecha fin: 21/04/2017
Programador responsable: Luis Marcalla	
Descripción: Esta clase dispondrá de los métodos necesarios para hacer llamado al web api académico por medio del método o verbo get luego de lo cual se mostrara elegantemente en el layout académico dentro de un listview personalizado.	

Tabla 4.37: Actividad 2 - Historia 10 - Ver historial académico

Elaborado por: Investigador

Las actividades de historia de usuario número 11 a continuación.

Actividades - Historia 11

Tarea	
Número: 1	Numero de historia: 11
Nombre de la tarea: Diseño de layout Administrativo	
Tipo de tarea: Diseño	Iteración Asignada: 3
Fecha inicio: 24/04/2017	Fecha fin: 27/04/2017
Programador responsable: Luis Marcalla	
Descripción: En esta pantalla se visualiza una lista de todas notificaciones enviadas por los administrativos; un listview personalizado mejorara la presentación para esta pantalla.	

Tabla 4.38: Actividad 1 - Historia 11 - Ver notificaciones

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 11
Nombre de la tarea: Desarrollo de clase administrativo	
Tipo de tarea: Desarrollo	Iteración Asignada: 3
Fecha inicio: 28/04/2017	Fecha fin: 05/05/2017
Programador responsable: Luis Marcalla	
Descripción: Esta clase dispondrá de los métodos necesarios para hacer llamado al web api administrativo por medio del método o verbo get luego de lo cual se mostrara elegantemente en el layout administrativo dentro de un listview personalizado.	

Tabla 4.39: Actividad 2 - Historia 11 - Ver notificaciones

Elaborado por: Investigador

Las actividades de historia de usuario número 12 a continuación.

Actividades - Historia 12

Tarea	
Número: 1	Numero de historia: 12
Nombre de la tarea: Diseño de layout eventos	
Tipo de tarea: Diseño	Iteración Asignada: 3
Fecha inicio: 08/05/2017	Fecha fin: 12/05/2017
Programador responsable: Luis Marcalla	
Descripción: En esta pantalla se visualiza "imágenes comunicativas" es necesario usar widgets propios de material design como es el RecyclerView. Los RecyclerView mostraran las imagen en forma de tarjetas fáciles de entender.	

Tabla 4.40: Actividad 1 - Historia 12 - Ver evento

Elaborado por: Investigador

Tarea	
Número: 2	Numero de historia: 12
Nombre de la tarea: Desarrollo de clase Eventos	
Tipo de tarea: Desarrollo	Iteración Asignada: 3
Fecha inicio: 15/05/2017	Fecha fin: 19/05/2017
Programador responsable: Luis Marcalla	
Descripción: Esta clase dispondrá de los métodos necesarios para hacer llamado al web api publicación por medio del método o verbo get luego de lo cual se mostrara en forma de tarjetas con la ayuda del RecyclerView presente en el layout Eventos.	

Tabla 4.41: Actividad 2 - Historia 12 - Ver evento

Elaborado por: Investigador

Resultados

A continuación se presentan los resultados de la tercera iteración.

El adaptater o adaptador es una clase especializada de java que reciben un conjunto de datos para mostrarlos en componentes visuales de forma ordenada y organizada. Los CardViews son componentes de Material Design para el sistema operativo Android, este muestra la información en forma de tarjetas de presentación de forma elegante. En el módulo eventos de la aplicación móvil cada nuevo registro se presentará en un CardView.

▪ **Código de Adaptador del CardView para módulo Eventos (Android). Tarea 2 de historia 12.**

```

1 public class AdapterRecyclerViewCalendar extends RecyclerView.Adapter<
    AdapterRecyclerViewCalendar.CalendarViewHolder> {
2     private ArrayList<Calendario> Calendarios;
3     private int resource;
4     private Activity activity;
5     public AdapterRecyclerViewCalendar(ArrayList<Calendario>
        Calendarios, int resource, Activity activity) {
            this.Calendarios = Calendarios;
6         this.resource = resource;
7         this.activity = activity;    }
8     @Override
9     public CalendarViewHolder onCreateViewHolder(ViewGroup parent, int
        viewType) {
10        View view = LayoutInflater.from(parent.getContext()).inflate(
            resource, parent, false);
11        return new CalendarViewHolder(view);    }
12
13     @Override
14     public void onBindViewHolder(CalendarViewHolder holder, int
        position) {
15        final Calendario Calendario = Calendarios.get(position);
16        holder.timeCard.setText(Calendario.getTime());
17        holder.titleCard.setText(Calendario.getNombre());
18        Picasso.with(activity).load("http://luismarcalla-001-site1.itempurl
            .com/Content_Uploads/"+Calendario.getImagen()).into(holder.
            CalendarioCard);
19        Picasso.with(activity).load(Calendario.getImagen()).into(holder.
            CalendarioCard);
20        holder.CalendarioCard.setOnClickListener(new View.OnClickListener()
            {
21
22            @Override
23            public void onClick(View view) {
24                Intent intent = new Intent(activity, VisorActivity.class);

```



```

25         intent.putExtra("url", Calendario.getImagen());
26         if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP)
27             {
28                 Slide slide = new Slide();
29                 slide.setDuration(1000);
30                 activity.getWindow().setExitTransition(slide);
31                 activity.startActivity(intent,
32                     ActivityOptionsCompat.makeSceneTransitionAnimation(
33                         activity, view, activity.getString(R.string.
34                             transitionname_Calendario)).toBundle());
35             }
36         }else {
37             activity.startActivity(intent);
38         }
39     }
40     @Override
41     public int getItemCount() {
42         return Calendarios.size(); }
43
44     public class CalendarViewHolder extends RecyclerView.ViewHolder{
45         private ImageView CalendarioCard;
46         private TextView titleCard;
47         public CalendarViewHolder(View itemView) {
48             super(itemView);
49             CalendarioCard = (ImageView) itemView.findViewById(R.id.
50                 Iv_Imagen);
51             titleCard = (TextView) itemView.findViewById(R.id.
52                 tv_title);
53         }
54     }

```

En la presente iteración se desarrollaron las layout y fragment de los módulos académico, administrativo y eventos para la aplicación móvil que posteriormente fueron integrados y probados en la aplicación móvil. La actividad principal de la aplicación Container muestra los fragments gracias al método *getSupportFragmentManager* de la clase *FragmentManager*, se hace uso de un switch para llevar este cometido.

```

1     switch (tabId){
2         case R.id.tab_school:
3             SchoolFragment homeFragment = new SchoolFragment();
4             getSupportFragmentManager().beginTransaction().replace(R.id.container,homeFragment)
5                 .setTransition(FragmentManager.TRANSIT_FRAGMENT_FADE)
6                 .addToBackStack(null).commit();
7         break;
8     ...

```

- Captura de proyecto en Android Studio 2.3.2

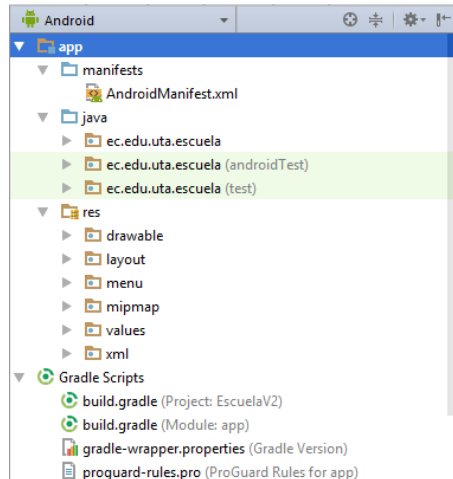


Figura 4.18: Proyecto escuela en android studio
Desarrollado por: Investigador

- Módulos desarrollados de la aplicación móvil (Android).

MODULO	N1	N2	N3	N4
Educación Vial	17.0	16.0	18.0	20.0
Leyes de tránsito	16.0	17.0	15.0	20.0
Mecanica Básica	18.0	17.0	15.0	14.0
Primeros Auxilios	16.8	17.9	14.5	16.2
Computación	17.0	18.0	16.0	15.0
Educación Ambiental	16.0	18.0	17.0	16.5
Relaciones Humanas	18.0	17.5	16.9	14.5
Atención al Cliente	16.0	18.9	20.0	12.5
Geografía Urbana	14.8	14.9	16.5	18.0
Conducción Vehicular	16.8	14.5	15.0	16.0
Práctica de Conducción	16.0	17.0	18.0	15.6

N1: Trabajo en equipo /4: Promedio
 N2: Estudio de casos SP: Supletorio
 N3: Prueba práctica NF: Nota Final
 N4: Prueba teórica

Figura 4.19: Módulo académico - Aplicación móvil
Elaborado por: Investigador



Figura 4.20: Módulo administrativo - Aplicación móvil
Elaborado por: Investigador



Figura 4.21: Módulo eventos - Aplicación móvil
Elaborado por: Investigador

4.5. Fase IV: Producción

La aplicación móvil no presentó inconvenientes al momento de hacer peticiones a un nuevo dominio, solo se requirió cambios en unas cuantas líneas de código. Posteriormente se procedió a subir el instalador de la aplicación móvil a la tienda de aplicaciones de Android donde previamente se creó una cuenta de desarrollador y se canceló el valor respectivo por el alojamiento.

4.5.1. Pruebas de aceptación

Las pruebas de aceptación son un requisito para cumplir con los objetivos de las historias de usuario, las pruebas dan la oportunidad de saber si lo implementado es lo que en realidad se deseaba. Como anteriormente se mostró las pruebas de aceptación están divididas para la aplicación web y la aplicación móvil.

4.5.1.1. Aplicación web (Administración)

Las pruebas a las funciones de la aplicación web fueron efectuadas para cada tipo de usuario habilitado para acceder, la aplicación respondió en concordancia a los datos ingresados. A continuación las pruebas efectuadas a la aplicación web.

Prueba de aceptación	
Número: 1	Historia de usuario #1, Acceso al sistema web
Nombre: Acceso al sistema web con datos correctos	
Descripción: Se ingresará al aplicativo proporcionando el correo electrónico y contraseña, se identificar el rol que juego el usuario ingresado, para el caso de los docentes se debe inicializar los cursos a los que imparte clases.	
Condición de ejecución: Conexión a internet Servicio de base de datos corriendo Usuario debe estar registrado con su rol respectivo en la base de datos	
Entrada: El usuario digita correo electrónico y contraseña La capa de presentación valida los datos esperados	
Resultado esperado: El navegador redirige al home de la aplicación, dependiendo del rol de cada usuario y habilita o restringe algunas funciones	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.22: Prueba 1 - Acceso al sistema web
Elaborado por: Investigador

Prueba de aceptación	
Número: 4	Historia de usuario #2, Mantenimiento de tablas
Nombre: Actualización de las tablas del sistema	
Descripción: El usuario después de haber iniciado sesión puede realizar operaciones sobre la base de datos, esto quiere decir que realiza las operaciones CRUD sobre cualquier tabla.	
Condición de ejecución: Conexión a internet Estar con sesión activa de usuario Administrador Conexión a base de datos Servicio de base de datos levantado	
Entrada: EL usuario digitan correo electrónico y contraseña El sistema muestra el home con todas las opciones para Administrador Se despliega la pestaña de mantenimiento y se selecciona una entidad de la base El usuario puede modificar datos con una interfaz fácil de entender.	
Resultado esperado: Las acciones de los controladores realizan a cabalidad las acciones que el usuario solicita con ayuda de los modelos y vistas. Base de datos alterada Web api devuelven nuevos datos	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.23: Prueba 2 - Mantenimiento de tablas
Elaborado por: Investigador

Prueba de aceptación	
Número: 5	Historia de usuario #3, Ingreso de notas
Nombre: Actualización de lista de calificaciones	
Descripción: El usuario docente puede acceder al sistema para la actualización de las calificaciones de sus paralelos a cargo.	
Condición de ejecución: - Conexión a internet - Estar con sesión activa de usuario Docente - Conexión a base de datos - Servicio de base de datos levantado - Asignación de paralelo(s) por parte del usuario Administrador	
Entrada: EL usuario digitan correo electrónico y contraseña - El sistema muestra el home con la pestaña de paralelos La pestaña paralelo despliega todos los paralelos asignados - Accede a una lista de calificaciones al presionar un paralelo - El docente puede modificar las notas de cada uno de los estudiantes presionando la opción de modificar - En caso de que el estudiante no haya alcanzado el puntaje necesario para aprobar la materia también está la casilla para la nota de supletorio - El docente cierra sesión.	
Resultado esperado: Las listas de calificaciones fueron actualizados exitosamente. Base de datos alterada El web api de calificaciones para la consulta por parte de la aplicación móvil arroja los datos actualizados	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.24: Prueba 3 - Ingreso de notas
Elaborado por: Investigador

Prueba de aceptación	
Número: 6	Historia de usuario #4, Envío de notificaciones
Nombre: Ingreso de nuevas notificaciones	
Descripción: Un usuario con el rol de Administrativo puede enviar una notificación a un estudiante cualquiera. El estudiante recibe este mensaje de manera inmediatamente.	
Condición de ejecución: - Conexión a internet -El usuario debe estar registrado en la base de datos - Estar con sesión activa de usuario Administrativo - Conexión a base de datos - Servicio de base de datos levantado	
Entrada: EL usuario digitan correo electrónico y contraseña - El sistema muestra el home con la pestaña de notificaciones habilitada - Se muestra una lista para elegir el paralelo al que pertenece el estudiante - Se elige el estudiante y se muestra una lista de notificaciones enviadas por todos los usuarios de rol Administrativo - El usuario ingresado solo puede alterar una notificación si solo él fue quien la creo para eso las validaciones respectivas - Se puede ingresar una nueva notificación y enviar al momento La nueva notificación de muestra en la lista de notificaciones pendientes del estudiante seleccionado Si se quiere enviar una nueva notificación se realizan los mismos pasos para cualquier estudiante El usuario sale del sistema	
Resultado esperado: Base de datos alterada exitoso La lista de notificaciones pendientes de determinado estudiante se va incrementando Web api de notificaciones devuelve la nueva lista actualizada con los datos correctos y lista para ser recibida por la aplicación móvil.	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.25: Prueba 4 - Envío de notificaciones

Elaborado por: Investigador

Prueba de aceptación	
Número: 7	Historia de usuario #5, Desarrollo de Web Api
Nombre: Web api responde a los métodos HTTP	
Descripción: Los webs service responden a cada uno de los métodos HTTP (Post, Get, Put), el web api de autenticación válida a un determinado estudiante para posteriormente enviarle toda su información (notas, notificaciones y eventos).	
Condición de ejecución: Conexión a internet -El usuario debe estar registrado en la base de datos Conexión a base de datos -Servicio de base de datos levantado Web api de autenticación desplegado en el servidor web	
Entrada: El usuario estudiante teclea sus credenciales en la pantalla de login Se envía los datos en formato JSON por método POST hacia el servidor web El servidor responde negativamente -La app muestra un mensaje informándole al usuario que los datos ingresados no corresponden a ningún estudiante matriculado.	
Resultado esperado: La aplicación móvil verifica si el solicitantes está registrado.	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.26: Prueba 5 - Desarrollado de Web API

Elaborado por: Investigador

4.5.1.2. Aplicación móvil

Las pruebas de aceptación realizadas a la aplicación móvil se efectuaron arrojando los resultados esperados. A continuación las pruebas de aceptación.

Prueba de aceptación	
Número: 8	Historia de usuario #6, Acceso a la aplicación
Nombre: Autenticación fallida	
Descripción: Un usuario no registrado tipea sus credenciales para poder ingresar al contenido de la aplicación móvil.	
Condición de ejecución: - App instala en el Smartphone -Conexión a internet -El usuario no debe estar registrado en la base de datos -Conexión a base de datos Servicio de base de datos levantado	
Entrada: - El usuario tipea sus credenciales en la pantalla de login - Envía sus datos esperando una respuesta - El servidor envía como resultado una consulta fallida Se deniega el ingreso al usuario	
Resultado esperado: Los Web Api responden correctamente a las solicitudes de autenticación de usuarios no identificados.	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.27: Prueba 6 - Acceso a la aplicación móvil
Elaborado por: Investigador

Prueba de aceptación	
Número: 10	Historia de usuario #7, Implementación de menú
Nombre: Navegación fácil de usar	
Descripción: El usuario estudiante una vez identificado navegar por las pantallas de la app de forma sencilla.	
Condición de ejecución: - App instala en el Smartphone - Conexión a internet - Conexión a base de datos - Servicio de base de datos levantado - Estudiante autenticado e iniciado sesión	
Entrada: El usuario tipea sus credenciales en la pantalla de <u>login</u> -Envía sus datos esperando una respuesta - El servidor envía como resultado una consulta exitosa -La app redirige al usuario al home de la aplicación -En el home se muestra una barra de botones en la parte inferior -El usuario iniciado sesión puede navegar por las pantallas con tan solo unos clics	
Resultado esperado: Las transiciones de pantallas durante la navegación no presenta errores ni deformaciones inesperadas.	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.28: Prueba 7 - Implementación de menú
Elaborado por: Investigador

Prueba de aceptación	
Número: 11	Historia de usuario #8, Escuela
Nombre: Visualización de aspecto ideológico de la Institución	
Descripción: La primera pantalla muestra una introducción en lo que se refiere a la Institución Educativa.	
Condición de ejecución: App instala en el smartphone Conexión a internet Servicio de base de datos levantado de parte del servidor web Estudiante autenticado e iniciado sesión	
Entrada: Estudiante registrado iniciado sesión La primera pantalla de la aplicación presenta en una interfaz intuitiva y bien presentada la misión y visión de la Institución así como también datos informativos.	
Resultado esperado: La información se muestra de manera elegante y fácil de entender. Se puede deslizar verticalmente la pantalla mostrando toda la información disponible	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.29: Prueba 8 - Información de escuela
Elaborado por: Investigador

Prueba de aceptación	
Número: 12	Historia de usuario #9, Perfil Académico
Nombre: Visualización de datos informativos de estudiante	
Descripción: Se muestra los datos de un estudiante con respecto a su matrícula dentro de la Institución.	
Condición de ejecución: App instala en el smartphone Conexión a internet Servicio de base de datos levantado de parte del servidor web Estudiante autenticado e iniciado sesión	
Entrada: Estudiante registrado iniciado sesión El estudiante da clic en la barra de botones en la opción Perfil. La aplicación muestra su datos personales así como también la licencia por la que opta y el paralelo al que pertenece.	
Resultado esperado: La pantalla muestra de forma clara los datos del estudiante	
Evaluación de la prueba: Prueba satisfactoria	

Figura 4.30: Prueba 9 - Ver datos personales
Elaborado por: Investigador

4.5.2. Implantación

Para la puesta en marcha del proyecto y beneficio de la noble institución educativa se a determinado que la aplicación desarrollada en ASP.NET (Panel de Administración) este disponible en un servidor IIS (servidor web de ASP.NET), listo para atender las solicitudes del personal administrativo de la institución . La aplicación web una vez alojada en este hosting respondió claramente a cada operación que se realizaba para el ingreso de datos relevantes que después serías solicitas por la aplicación móvil. Las Web API's también fueron probadas con ayuda de una extensión de Chrome para peticiones a web services basados en REST.

La aplicación móvil se aloja en la tienda de aplicaciones de Android (Play Store) facilitando a los estudiantes el poder acceder a ella sin ningún contratiempo. Para poder subir el instalador de la aplicación móvil (APK) a la Play Store se requirió firmar la aplicación móvil, la firma serviría como identificador único de la aplicación a la vez que crea una barrera de seguridad para el desarrollador y la tienda de aplicaciones para futuras actualizaciones.

El proceso de firma de la aplicación se realizó en línea de comandos de Windows con ayuda del SDK de Java. A continuación unas capturas del proceso.

```
Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.15063]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\WINDOWS\system32>cd C:\Program Files\Java\jdk1.8.0_121\bin

C:\Program Files\Java\jdk1.8.0_121\bin>keytool -genkey -v -keystore fisei.keystore -alias fisei_alias -keyalg RSA -keysize 2048 -validity 10000
Introduzca la contraseña del almacén de claves:
Volver a escribir la contraseña nueva:
¿Cuáles son su nombre y su apellido?
  [Unknown]: Luis Marcalla
¿Cuál es el nombre de su unidad de organización?
  [Unknown]: UTA
¿Cuál es el nombre de su organización?
  [Unknown]: fisei
¿Cuál es el nombre de su ciudad o localidad?
  [Unknown]: Ambato
¿Cuál es el nombre de su estado o provincia?
  [Unknown]: Tungurahua
¿Cuál es el código de país de dos letras de la unidad?
  [Unknown]: EC
¿Es correcto CN=Luis Marcalla, OU=UTA, O=fisei, L=Ambato, ST=Tungurahua, C=EC?
  [no]: si

Generando par de claves RSA de 2.048 bits para certificado autofirmado (SHA256withRSA) con una validez de 10.000 días
para: CN=Luis Marcalla, OU=UTA, O=fisei, L=Ambato, ST=Tungurahua, C=EC
Introduzca la contraseña de clave para <fisei_alias>
  (INTRO si es la misma contraseña que la del almacén de claves):
Volver a escribir la contraseña nueva:
[Almacenando fisei.keystore]

C:\Program Files\Java\jdk1.8.0_121\bin>
```

Figura 4.31: Creación de firma - Firma de APK
Desarrollado por: Investigador

```

Administrador: Símbolo del sistema - jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore fisei.keystore app-release.apk fisei_alias
Microsoft Windows [Versión 10.0.15063]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

C:\WINDOWS\system32>cd C:\Program Files\Java\jdk1.8.0_121\bin
C:\Program Files\Java\jdk1.8.0_121\bin>
C:\Program Files\Java\jdk1.8.0_121\bin>jarsigner -verbose -sigalg SHA1withRSA -digestalg SHA1 -keystore fisei.keystore app-release.apk fisei_alias
Enter Passphrase for keystore:

```

Figura 4.32: Ejecución de firma - Firma de APK
Desarrollado por: Investigador

```

Administrador: Símbolo del sistema
X.509, CN=Luis Marcalla, OU=UTA, O=fisei, L=Ambato, ST=Tungurahua, C=EC
[certificate is valid from 07/06/17 13:25 to 23/10/44 13:25]
[CertPath not validated: Path does not chain with any of the trust anchors]
sm 880 Fri Nov 30 00:00:00 COT 1979 res/xml/pref_headers.xml
X.509, CN=Luis Marcalla, OU=UTA, O=fisei, L=Ambato, ST=Tungurahua, C=EC
[certificate is valid from 07/06/17 13:25 to 23/10/44 13:25]
[CertPath not validated: Path does not chain with any of the trust anchors]
sm 976 Fri Nov 30 00:00:00 COT 1979 res/xml/pref_notification.xml
X.509, CN=Luis Marcalla, OU=UTA, O=fisei, L=Ambato, ST=Tungurahua, C=EC
[certificate is valid from 07/06/17 13:25 to 23/10/44 13:25]
[CertPath not validated: Path does not chain with any of the trust anchors]
sm 320188 Fri Nov 30 00:00:00 COT 1979 resources.arsc
X.509, CN=Luis Marcalla, OU=UTA, O=fisei, L=Ambato, ST=Tungurahua, C=EC
[certificate is valid from 07/06/17 13:25 to 23/10/44 13:25]
[CertPath not validated: Path does not chain with any of the trust anchors]

s = signature was verified
m = entry is listed in manifest
k = at least one certificate was found in keystore
i = at least one certificate was found in identity scope

Signed by "CN=Luis Marcalla, OU=UTA, O=fisei, L=Ambato, ST=Tungurahua, C=EC"
Digest algorithm: SHA1
Signature algorithm: SHA1withRSA, 2048-bit key

jar verified.

Warning:
This jar contains entries whose certificate chain is not validated.
This jar contains signatures that does not include a timestamp. Without a timestamp, users may not be able to validate this jar after the signer certificate's expiration date (2044-10-23) or after any future revocation date.

C:\Program Files\Java\jdk1.8.0_121\bin>

```

Figura 4.33: Verificación de firma - Firma de APK
Desarrollado por: Investigador

4.6. Fase V: Mantenimiento

La metodología XP estipula que cada versión del sistema, resultado de cada iteración debe ser operativa y funcional, dándole valor al negocio. El equipo de desarrollo debe procurar el buen funcionamiento de la versión reciente y al mismo tiempo desarrollar la siguiente iteración, de ser necesario aumentar nuevas historias de usuario o nuevos requerimientos cuando lo requiera el propietario del software

En el caso de la realización del presente proyecto después de la entrega de la última iteración el representante de la institución educativa mencionó que el agregar una funcionalidad más a la parte administrativa beneficiaría a la institución, luego de estructurar a detalle la nueva funcionalidad se continuó a implementarla.

El nuevo requerimiento giraba alrededor de la función de enviar notificaciones, hasta el momento un par de administrativos (Inspector/Secretaria) podían enviar

mensajes. Si bien esta acción de enviar una notificación a un estudiante determinado era correcto pero no abarcaba a los demás administrativos que son también parte activa de la Escuela de Capacitación. Los nuevos administrativos a implementarse fueron el Director Administrativo de la honorable institución encargado de la representación legal de la institución y el instructor vial que es el representante de la ANT dentro de la organización y es encargado de informar acerca de las rutas a trazarse para la prácticas de conducción.

4.7. Fase VI: Muerte del proyecto

Última fase de la metodología seleccionada para la realización del presente proyecto, sucede cuando el cliente no tiene más historias a ser incluidas en el sistema por lo que se prosigue a la documentación.

Después de la última entrega y los cambios solicitados por parte del representante de la institución tanto la aplicación web y la aplicación móvil quedaron a disposición de los estudiantes matriculados y personal administrativo. Cabe recalcar que en esa ocasión estaba en vigencia 8 paralelos, 4 paralelos para licencia tipo D y cuatro paralelos tipo E.

CAPÍTULO 5

Conclusiones y recomendaciones

5.1. CONCLUSIONES

- La aplicación móvil implantada bajo la plataforma Android para la gestión académica y administrativa en la “Escuela de Formación y Capacitación de Conductores Profesionales Ambato” ayuda al personal de la comunidad educativa a realizar tareas del proceso académico/administrativo en forma ágil e independiente del lugar donde se encuentren, optimizando tiempo y recursos.
- Bootstrap es un marco de trabajo de fácil implementación para desarrollar sitios web adaptativos, debido a que su utilización no requiere conocimientos amplios en materia de estilos CSS y además dispone de una extensa documentación acerca de la implementación de sus componentes (Jumbotron, Dropdowns, Glyphicons, etc.).
- La comunicación entre distintas plataformas de desarrollo de software ha dejado de ser un inconveniente, debido a la aparición de los servicios web los cuales usan estándares y protocolos muy bien definidos, en el caso del presente proyecto se uso servicios a través de Web API's, basados en REST, para el envío y recepción de información entre una aplicación web (ASP.NET) y una aplicación móvil (Android).
- La “Escuela de Formación y Capacitación de Conductores Profesionales Ambato” será pionera en integrar sus servicios a la plataforma Android, el impacto de esto promoverá el uso de la TIC's en la sociedad a la vez que llamará la atención de las personas que optan por obtener una licencia de conductor profesional.

5.1.1. RECOMENDACIONES

- Se recomienda la utilización de la metodología de desarrollo ágil Xtreme Programming (XP) para medianos y pequeños proyectos, debido a que los requerimientos o historias de usuario que se establecen en la etapa inicial no son estáticos, sino más bien son flexibles y se los puede ir modificando de acuerdo a las necesidades del proyecto, durante el transcurso de cada iteración.
- Para el desarrollo de aplicaciones web se recomienda usar el patrón de arquitectura MVC por las ventajas que tiene su implementación, la separación de la capa de datos, y la lógica de negocio de la interfaz de usuario permiten la reutilización de código y hacen posible un mantenimiento adecuado de la aplicación.
- Se aconseja implementar un servicio de notificaciones para el módulo académico de la aplicación móvil, el cual estará encargado de notificar al estudiante si su historial académico se actualizó, esto sin que la aplicación se este ejecutando en primer plano.
- Se sugiere el empleo de técnicas de marketing dirigidas a promocionar la aplicación desarrollada hacia los usuarios finales.

Bibliografía

- [1] E. M. T. Llanganate, “Aplicación móvil utilizando plataforma android para mejorar la calidad del servicio de consulta de información de consumo eléctrico de la eeasa en la empresa besixplus cia. ltda.” Disponible en: <http://repo.uta.edu.ec/handle/123456789/4955>, Junio 2013.
- [2] O. E. A. Mayorga, “Aplicación móvil bajo la plataforma android para la gestión de calificaciones en la unidad educativa augusto nicolás martínez.” Disponible en: <http://repo.uta.edu.ec/handle/123456789/10999>, Junio 2015.
- [3] S. Z. Z. Johana Nataly Picon Cajamarca, “Diseño e implementación de una aplicación móvil para trabajo operativo de los vendedores de la empresa agrota cia. ltda. universidad politecnica salesiana.” Disponible en: <dspace.ups.edu.ec/bitstream/123456789/7815/1/UPS-CT004666.pdf>, Diciembre 2014.
- [4] Q. P. D. Fernando, “Diseño e implementación de una aplicación para dispositivos android en el marco del proyecto pequeñas y pequeños científicos de la universidad politécnica salesiana.” Disponible en: <http://dspace.ups.edu.ec/bitstream/123456789/3298/1/UPS-CT002537.pdf>, Diciembre 2010.
- [5] J. D. Morillo, “Introducción a los dispositivos móviles.” Disponible en: [https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_\(Modulo_2\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Modulo_2).pdf), 2010.
- [6] K. M. Polanco and J. L. B. Tai, “Red de revistas científicas de américa latina y el caribe, españa y portugal.” Disponible en: <http://www.redalyc.org/pdf/782/78219156004.pdf>, 2012.
- [7] J. Stalaman, “Sistema operativo ios.” Disponible en: <https://es.slideshare.net/jatzen01/sistema-operativo-ios-41898500>, Noviembre 2014.

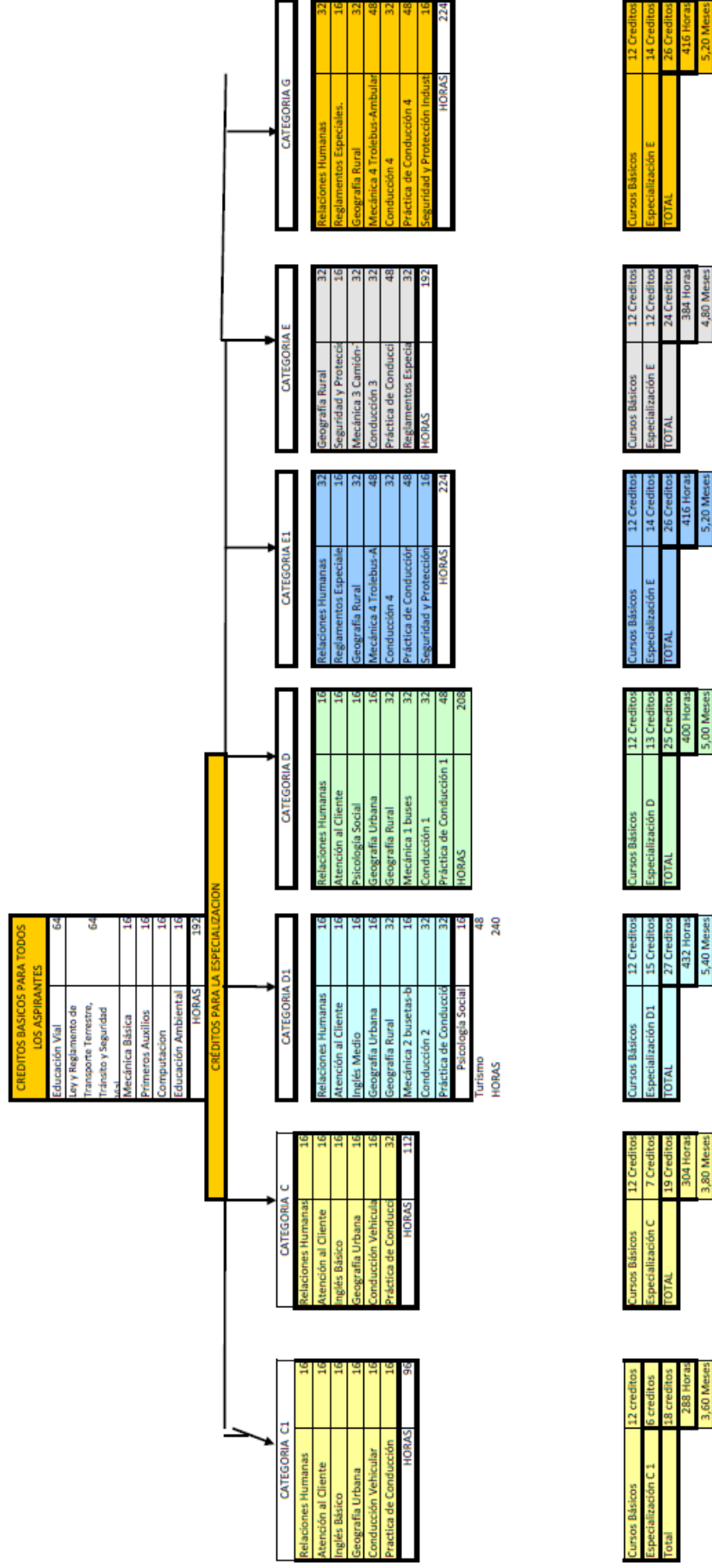
- [8] V. Saldaña, “Sistemas operativos móviles.” Disponible en: <http://sistoperativos98.blogspot.com/2016/07/sistemas-operativos-moviles.html>, Junio 2016.
- [9] J. A. Tudela, “Desarrollo de aplicaciones para dispositivos móviles sobre la plataforma android de google.” Disponible en: http://e-archivo.uc3m.es/bitstream/handle/10016/6506/PFC_Jaime_Aranaz_Tudela_2010116132629.pdf;jsessionid=F8C345E45C63C51E29072B9EA75C81A8?sequence=1, Enero 2009.
- [10] U. C. I. de Madrid, “Arquitectura android.” Disponible en: <https://sites.google.com/site/swcuc3m/home/android/generalidades/2-2-arquitectura-de-android>.
- [11] J. T. Gironés, *El gran libro de Android*. Marcombo, 2012.
- [12] SOCIALMOOD, “¿qué es el diseño responsive?.” Disponible en: <https://www.40defiebre.com/que-es/disenio-responsive/>, Mayo 2015.
- [13] w3schools.com, “Responsive web design - introducción.” Disponible en: https://www.w3schools.com/css/css_rwd_intro.asp.
- [14] J. Solis, “¿qué es bootstrap y cómo funciona en el diseño web?.” Disponible en: <http://www.arweb.com/chucherias/editorial/%C2%BFque-es-bootstrap-y-como-funciona-en-el-diseno-web.htm>, Noviembre 2014.
- [15] A. Leiva, “Patrones de diseño de software.” Disponible en: <https://devexperto.com/patrones-de-diseno-software/>, Julio 2016.
- [16] E. Gamma, “Introducción a los patrones de diseño.” Disponible en: <http://codecriticon.com/introduccion-patrones-diseno/>, Noviembre 2015.
- [17] EcuRed, “Patrones de diseño y arquitectura.” Disponible en: https://www.ecured.cu/Patrones_de_dise%C3%B1o_y_arquitectura, Julio 2017.
- [18] U. Hernández, “Mvc (model, view, controller) explicado.” Disponible en: <https://codigofacilito.com/articulos/mvc-model-view-controller-explicado>, 2015.
- [19] B. Alveo, “Patrones de diseño.” Disponible en: <http://patronesdiseno.blogspot.com/2013/07/modelo-vista-controlador-mvc.html>, 2013.

- [20] M. Peñafiel, “Aplicaciones web.” Disponible en: <https://es.scribd.com/doc/136052164/APLICACIONES-WEB-pdf>, Julio 2013.
- [21] P. Computacional, “Visual csharp.” Disponible en: <http://informaticabachilleratoitea.blogspot.com/p/visual-c.html>, 2013.
- [22] E. P. Bautista, “Programación web con asp.net.” Disponible en: <http://aplicaciones-web-lenguajes-programaci.blogspot.com/2011/12/aspaspnet.html>, 2016.
- [23] Microsoft, “Información general sobre asp.net mvc.” Disponible en: [https://msdn.microsoft.com/es-es/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/es-es/library/dd381412(v=vs.108).aspx), 2017.
- [24] J. QUIJANO, “Asp.net web api, framework orientado a la construcción de aplicaciones restful.” Disponible en: <https://www.genbetadev.com/frameworks/asp-net-web-api-framework-orientado-a-la-construccion-de-aplicaciones-restful>, Marzo 2013.
- [25] R. N. Marset, “Rest vs web services.” Disponible en: <file:///C:/Users/Luis/Documents/tesis%20v2/RestVsWebServices.pdf>.
- [26] G. de España, “Servidor iis.” Disponible en: <http://2003server.webcindario.com/iis/definici.htm>, Junio 2015.
- [27] J. Porras, “Historia y definición de microsoft sql server.” Disponible en: <https://prezi.com/absznnvrqpsr/historia-y-definicion-de-microsoft-sql-server/>, Agosto 2011.
- [28] T. y. S. V. Directorio de la Agencia Nacional de Regulación y Control del Transporte, “Reglamento de escuelas de capacitación para conductores profesionales.” Disponible en: <http://www.ant.gob.ec/index.php/transito-7/resoluciones-2015/file/2730-resolucion-no-010-dir-2015-ant-reglamento-de-escuelas-de-capacitacion-para-conductores-profesionales>, Noviembre 2013. Resolución No. 010-DIR-2015-ANT.
- [29] D. E. A. y G. Fitzgerald, *Information system development*. Maidenhead, 2006.
- [30] C. P. Patricio Letelier, “Métodologías ágiles para el desarrollo de software: extreme programming (xp),” *Ciencia y Técnica Administrativa*, vol. 05, no. 26, 2006.

- [31] J. M. BAUTISTA, “Programación extrema (xp) extreme programming (xp).” Disponible en: <http://ingenieriadesoftware.mex.tl/images/18149/PROGRAMACI%C3%93N%20EXTREMA.pdf>, 2017.
- [32] extremeprogramming, “Programación extrema.” Disponible en: <http://www.extremeprogramming.org/>, 2013.
- [33] A. H. Jeffries, R., *Extreme Programming Installed*. Addison-Wesley Longman Publishing Co., 2001.

Anexos y apéndices

■ Malla curricular emitida por la Agencia Nacional de Tránsito para la obtención de Título de Conductor Profesional



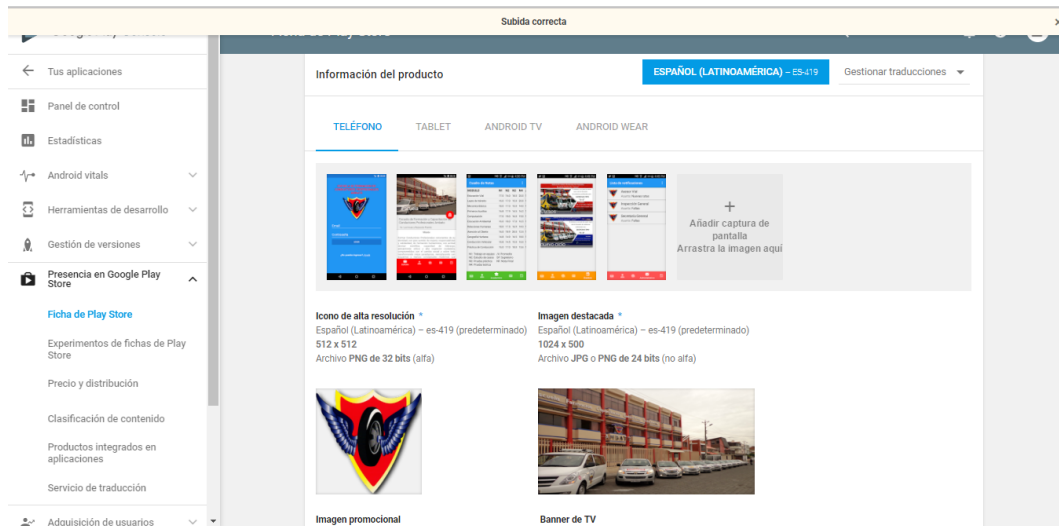
- Exteriores de la ECCP



- Título de Conductor Profesional Tipo “C”



■ Ficha de la aplicación móvil en la Play Store



Manual de usuario

Anexo A

Manual de aplicación móvil

A.1. Introducción

La presente Aplicación Móvil(app) esta desarrollada para la plataforma Android, destinada para los smartphones que tiene instalado Android como S.O. Dispone de funciones personalizadas para el uso exclusivo de la prestigiosa "Escuela de Capacitación y Formación de Conductores Profesionales de la ciudad de Ambato", además posee características innovadoras que con las tecnologías actuales y nuevos paradigmas se pueden lograr. En cuanto a compatibilidad con las distintas versiones de Android, la app puede ejecutarse desde la versión 4.3 de Android (Jelly Bean) alcanzando un 80 % de smartphone disponibles en el mercado.

La app consta de su panel de administración el cual es una aplicación ASP.NET desarrollada bajo la plataforma de Visual Studio, el objeto de este panel de administración es la de inicialización de la información relevante a los procesos académicos y administrativos para una posterior consulta de los mismos por parte de la app. Es importante mencionar que tanto el panel de administración y la app trabajan de manera conjunta, son interdependientes una del otra.

A.2. Instalación

Para acceder a la app el solicitante puede hacerlo directamente de la tienda de aplicaciones de Android (Play Store), basta con tipear «escuela ambato» para que se muestre la app lista para ser instalada. La app no necesita mayores permisos.

En la siguiente página se muestra como se visualiza la app en la Play Store.

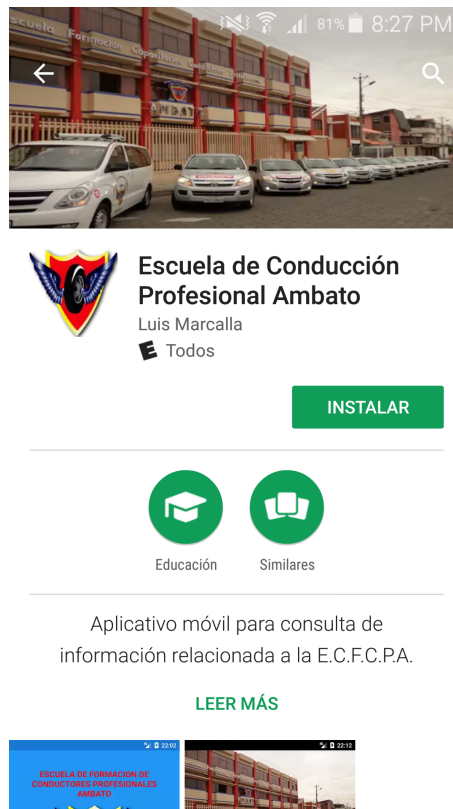


Figura A.1: Instalación - Aplicación móvil
Elaborado por: Investigador

A.3. Guía de uso

A.3.1. Autenticación

La primera pantalla de la aplicación es la de login (autenticación), en ella el estudiante ya matriculado legalmente y registrado en la base de datos podrá acceder sin ningún problema. Esto lo logra escribiendo sus datos (email y cédula).

Una vez identificado el estudiante la aplicación lo redirigirá a la pantalla principal de la aplicación. En la parte inferior se visualizará una barra de botones propio de Material Design (Paradigma de diseño implementado por Google a partir del 2004 para Android), esta le servirá al usuario estudiante para navegar por las distintas pantallas de la aplicación de forma rápida.

La pantalla de login de la aplicación móvil contiene el sello en la parte central y la parte superior el nombre completo de la institución. En la siguiente página se muestra la pantalla de login.

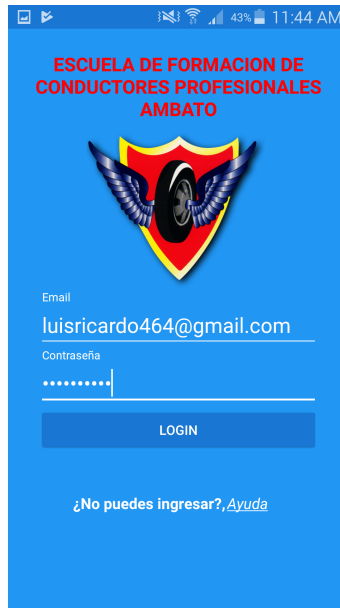


Figura A.2: Login - Aplicación móvil
Elaborado por: Investigador

A.3.2. Módulo Escuela

La pantalla que se visualiza por defecto es la de "Escuela", esta visualiza el aspecto ideológico de la institución como también la localidad de la misma.



Figura A.3: Escuela - Aplicación móvil
Elaborado por: Investigador

A.3.3. Módulo Perfil Académico

La segunda pantalla es la de "Perfil", en esta se muestra información relevante del estudiante en lo concerniente a su matricula dentro de la institución.

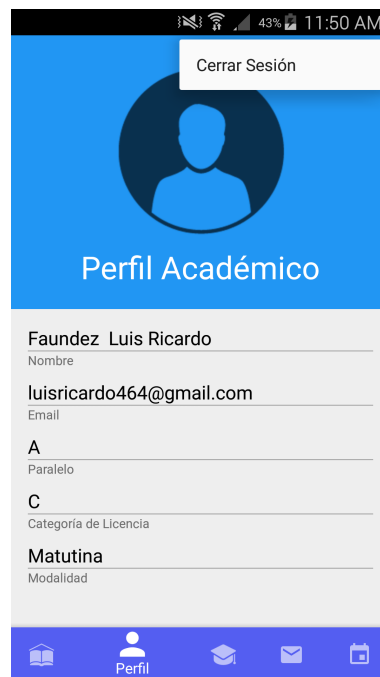


Figura A.4: Perfil académico - Aplicación móvil
Elaborado por: Investigador

A.3.4. Módulo Académico

La Tercera pantalla "Académico", en esta pantalla el estudiante puede saber su puntaje alcanzado en cada uno de los módulos que recibe. Esta función esta inmediatamente disponible al momento de que el Docente respectivo actualice la lista de calificaciones por medio del panel de Administración(Aplicación ASP.NET) más adelante mas información acerca de esta función.

Esta pantalla tiene la peculiaridad de poder puede desplazarse horizontalmente, gracias a la inserción *HorizontalScrollView* en su layout. Los atributos de este layout permiten al estudiante poder visualizar todo el cuadro de notas sin inconvenientes.

A continuación el módulo académico.

MODULO	N1	N2	N3	N4
Educación Vial	17.0	16.0	18.0	20.0
Leyes de tránsito	16.0	17.0	15.0	20.0
Mecánica Básica	18.0	17.0	15.0	14.0
Primeros Auxilios	16.8	17.9	14.5	16.2
Computación	17.0	18.0	16.0	15.0
Educación Ambiental	16.0	18.0	17.0	16.5
Relaciones Humanas	18.0	17.5	16.9	14.5
Atención al Cliente	16.0	18.9	20.0	12.5
Geografía Urbana	14.8	14.9	16.5	18.0
Conducción Vehicular	16.8	14.5	15.0	16.0
Práctica de Conducción	16.0	17.0	18.0	15.6

N1: Trabajo en equipo /4: Promedio
 N2: Estudio de casos SP: Supletorio
 N3: Prueba práctica NF: Nota Final
 N4: Prueba teórica

Figura A.5: Académico - Aplicación móvil
Elaborado por: Investigador

A.3.5. Módulo Administrativo

El módulo administrativo consta de un servicio que se ejecuta en segundo plano, el servicio se encarga de notificar al estudiante cuando un mensaje de un administrativo le es recibido. El servicio hace uso de la API *Notification Manager* de la Arquitectura de Android.



Figura A.6: Notificación recibida - Aplicación móvil
Elaborado por: Investigador

La notificación entrante redirige al módulo administrativo, en ella se muestra una lista de notificaciones las mismas que son enviadas directamente por los administrativos de la institución con la ayuda del Panel de Administración de la App.



Figura A.7: Lista de notificaciones - Aplicación móvil
Elaborado por: Investigador

A.3.6. Módulo Eventos

La idea de la realización de esta opción es la de un «Muro Virtual» donde el Administrador subirá con la ayuda del panel de administración nuevos anuncios, comunicados, horarios, etc.

El administrador puede subir un nuevo evento cuando lo crea necesario desde la Aplicación Web, el sistema permite la subida de cualquier tipo de imagen como jpg y png. Los anuncios publicados se visualizarán en forma de tarjetas las cuales al hacerle clic se amplía la imagen para una mejor visualización, continuación un ejemplo de esto en las siguientes figuras.



Figura A.8: Eventos - Aplicación móvil
Elaborado por: Investigador

Anexo B

Manual de administración web

B.1. Introducción

La Administración de la app es realizada con ayuda de una aplicación ASP.NET desarrollada bajo la plataforma de Visual Studio, dicha aplicación web posee propiedades que la hacen de enorme utilidad. Su diseño es adaptativo, cambia hasta adecuarse a la pantalla donde se inicia sesión.

Los tipos de usuarios habilitados para ingresar al sistemas son:

Administrador.- Encargado de el ingreso de datos de los estudiantes, paralelos, docentes, etc. Realiza las relaciones y mantiene consistentes los datos con el fin de que todo el conjunto funcione correctamente.

Administrativo.- Este usuario es variado puede ser secretaria, inspector o cualquiera administrativo debidamente registrado por parte del usuario Administrador.

Docente.-Las obligaciones del usuario es la actualizar el historial académico de sus paralelos a cargo dependiendo del módulo o materia que imparte.

Para acceder a la aplicación web es necesario disponer de un navegador web e ingresar a la URL <http://escuelaconduccionambato.marckala.com/> posteriormente se les solicitará las respectivas credenciales, la aplicación web luego de validar al usuario visualizara las respectivas funciones dependiendo del tipo de usuario o rol que juega en la institución.

En la siguiente página se muestra el acceso al panel de administración de la aplicación móvil.

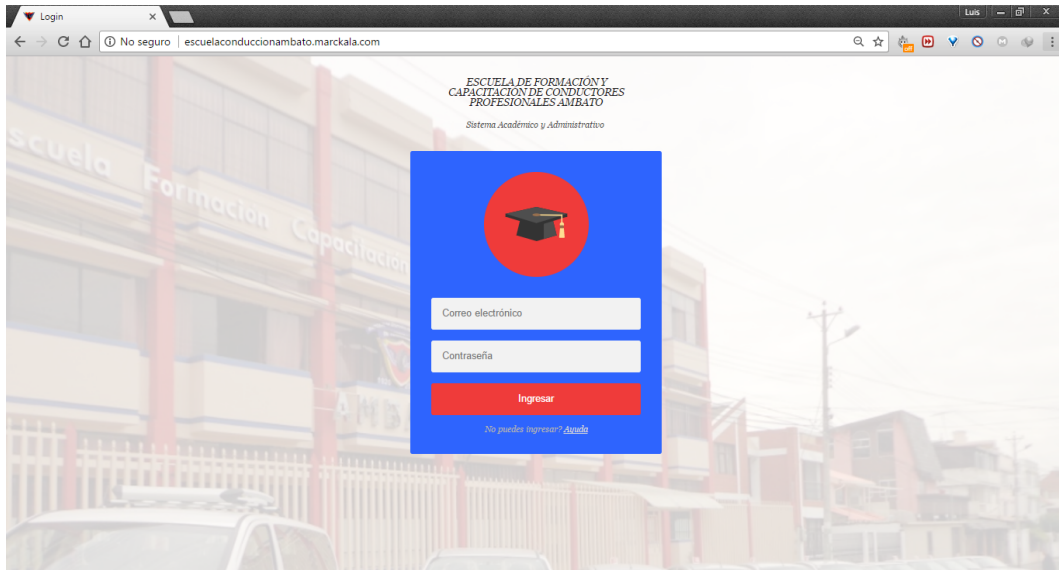


Figura B.1: Login - Aplicación web
Elaborado por: Investigador

B.2. Módulo Administrador

Luego de haberse logeado el usuario Administrador puede acceder a las tablas del sistema donde por medio de una interfaz fácil de usar puede realizar cualquier operación (inserción, actualización, borrado y ver detalles).

A continuación se muestran las entidades de la base de datos que pueden ser modificadas



Figura B.2: Dropdown - Aplicación web
Elaborado por: Investigador

En este ejemplo se toma como ejemplo la entidad Licencias, luego de haberlo presionado nos mostrará lo siguiente:

Nótese que se tiene 3 registros ingresados, tres tipos de licencias a los cuales se puede editar, ver los detalles y borrar respectivamente.

Categoria	Descripción	Meses	Operaciones
C	Para taxis convencionales, ejecutivos, camionetas livianas o mixta hasta 3.500 kg, hasta 8 pasajeros; vehículos de transporte de pasajeros de no más de 25 asientos y los vehículos comprendidos en el tipo B	6	
D	Para servicio de pasajeros (intracantonales, interprovinciales, intra - provinciales, intraregionales y por cuenta propia)	4	
E	Para camiones pesados y extra pesados con o sin remolque de más de 3,5 toneladas, tráiler, volquetas, tanqueros, plataformas públicas, cuenta propia, otros camiones y los vehículos estatales con estas características.	3	

Figura B.3: Lista de categorías de licencias - Aplicación web
Elaborado por: Investigador

Editar Licencia

Categoria

Descripción

Meses

Figura B.4: Edición de categorías de licencias - Aplicación web
Elaborado por: Investigador

Las relaciones primordiales para tener una base consistente se evidencian en la tabla horario donde se asigna , esto va depender de el módulo que el docente imparte, horas disponibles, etc.

Dia	Hora	Docente	Paralelo	Operaciones
Lunes - Viernes	8:00 am - 9:30 am	Gallardo Honorina	A	
Lunes - Viernes	9:45 am - 11:15 am	Aravena Susana Angela	A	
Lunes - Viernes	8:00 am - 9:30 am	Recabal Flor Esilda	A	
Lunes - Viernes	9:45 am - 11:15 am	Vera Samuel Segundo	A	
Lunes - Viernes	8:00 am - 9:30 am	Salgado Juan	A	
Lunes - Viernes	8:00 am - 9:30 am	Rocha Irma Eliana	A	
Lunes - Viernes	9:45 am - 11:15 am	Cuevas Rosa Eulalia	A	
Lunes - Viernes	9:45 am - 11:15 am	Salazar José Felipe	A	
Lunes - Viernes	9:45 am - 11:15 am	Rivera Magaly del Pilar	A	
Lunes - Viernes	8:00 am - 9:30 am	Fredes Juan Honorino	A	
Lunes - Viernes	8:00 am - 9:30 am	Villar Tania Angelica	A	

Figura B.5: Horarios de clases - Aplicación web
Elaborado por: Investigador

Figura B.6: Nuevo horario de clases - Aplicación web
Elaborado por: Investigador

B.3. Módulo Administrativo

El usuario Administrativo que puede ser cualquiera de los administrativos de la institución ya anteriormente registrado por parte del usuario administrador puede acceder al panel de administración. La pantalla que visualizó es la home que le muestra la pestaña «notificaciones» para hacer uso de ella.

La razón de este módulo es la tener la posibilidad de enviar mensajes personalizados a determinado estudiante por parte de un administrativo. La naturaleza del mensaje

gira entorno al tipo de administrativo que envía la notificación como por ejemplo un inspector enviará mensajes acerca de las faltas injustificadas y demás actividades propias de este administrativo.

A continuación un ejemplo.

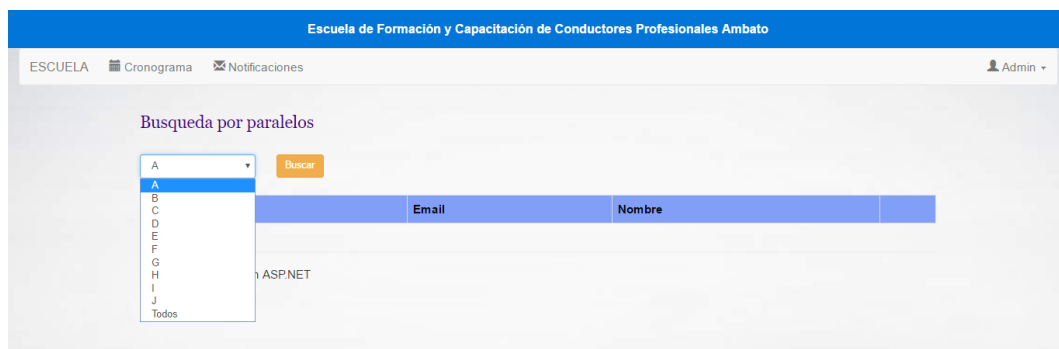


Figura B.7: Búsqueda por paralelo - Aplicación web
Elaborado por: Investigador

The screenshot shows the same web application interface as Figure B.7, but with the search filter set to 'I'. The 'Buscar' button is highlighted. Below the search section, a table displays a list of students. The table has four columns: 'Cedula', 'Nombre', 'Email', and 'Nombre'. Each row represents a student, and there is a small yellow icon in the final column of each row.

Cedula	Nombre	Email	Nombre
1805293592	Acosta Santacruz, Diego Oswaldo	Acostadiego46@Hotmail.Es	I
1804721072	Adame Maisanche, Edgar Vladimir	Edgaradame20@Hotmail.Com	I
1803883873	Altamirano Guerrero, Alex Fabian	Aleks_27@Yahoo.Es	I
1805335377	Altamirano Toro, Juan Sebastian	Altamiranojuansebastian@Gmail.Com	I
1803782515	Andachi Olovacha, Milton Gustavo	Rosi-Paucar@Hotmail.Com	I
1804097630	Arcos Guerrero, Luis Arturo	Luiggycaros@Outlook.Com	I
1804747846	Barona Vaca, Henry Paul	Baronapaul21@Gmail.Com	I
2100658364	Barragan Vega, Richath Bladimir	Richathbarragan1808@Hotmail.Com	I
1802941672	Calero Nuñez, Veronica Jackeline	Verocanu108@Yahoo.Es	I
1250049764	Chimborazo Manobanda, Jaime Homero	Jaimechimborazo226@Gmail.Com	I
1804263802	Espin Montero, Darwin David	Darwinespin1988@Gmail.Com	I

Figura B.8: Estudiantes por paralelo - Aplicación web
Elaborado por: Investigador

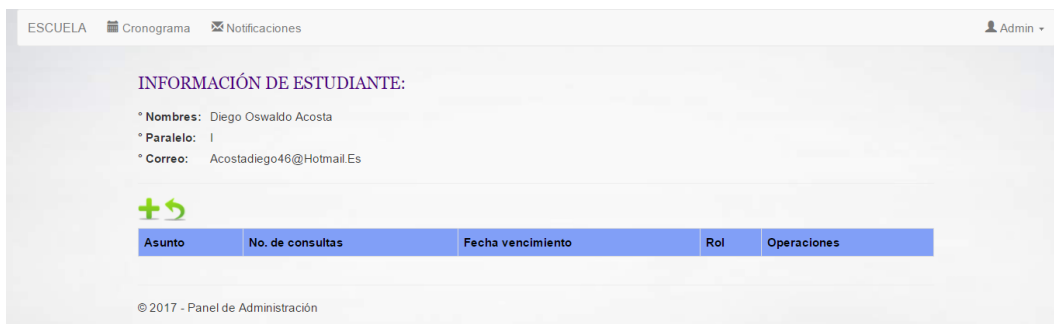


Figura B.9: Notificaciones de estudiante - Aplicación web
Elaborado por: Investigador

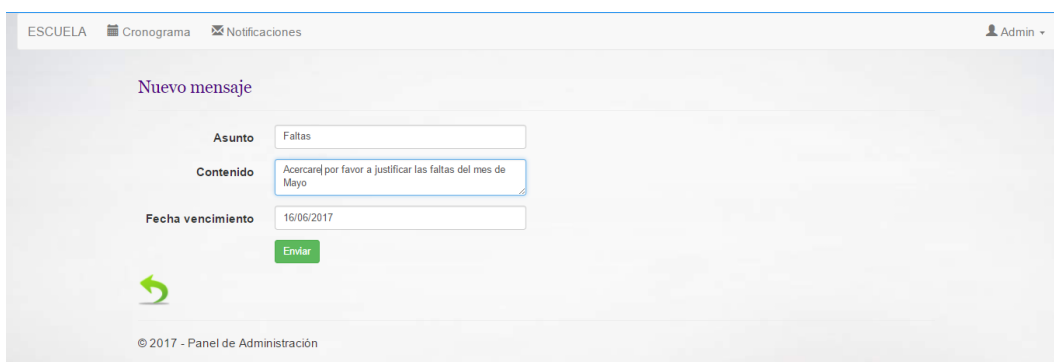


Figura B.10: Nueva notificación de estudiante - Aplicación web
Elaborado por: Investigador

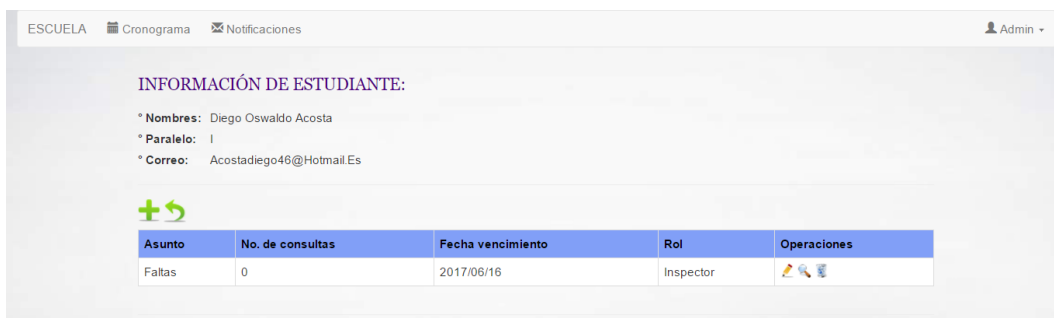


Figura B.11: Notificación ingresada de estudiante - Aplicación web
Elaborado por: Investigador

Después de haber realizado todos estos pasos por parte de determinado administrativo el estudiante puede observar enseguida los detalles del mensaje enviado desde la aplicación móvil en la opción Administrativo, para una mejor vista diríjase al módulo Académico de la guía de uso de la app.

Nota: Para la creación de cada notificación existe un atributo «fecha vence» que sirve para definirse un tiempo de validez a la notificación, después de la fecha la notificación ya no sera recibida por el estudiante por ya estar vencida. Con esto se evita que las notificaciones se envíen permanentemente al un estudiante determinado.

B.3.1. Módulo Docente

El usuario Docente encargado de evaluar académicamente a los estudiantes, después de haberse logueado en la aplicación web se le desplegará home y se le habilitara la opción de paralelos donde puede acceder a la lista de los paralelos que se le han asignado anteriormente por parte del usuario Administrador.

Al dar clic sobre un determinado paralelo puede actualizar la calificación de los estudiantes que conforman el paralelo.

A continuación se muestra como debe de proceder un docente para actualizar el historial de calificaciones de los paralelos a su cargo.



Figura B.12: Paralelos de docente - Aplicación web
Elaborado por: Investigador

Escuela de Formación y Capacitación de Conductores Profesionales Ambato										
ESCUELA	Cronograma	Perfil	Paralelos							Salir
LISTADO DE CALIFICACIONES										
* Paralelo: A										
* Modulo: Educación Vial										
Nombre	Trabajo en equipo	Estudio de casos	Prueba práctica	Prueba teórica	Promedio	Supletorio	Nota final	Estado	Operación	
Alarcon Maria del Carmen	16,00	15,00	14,00	20,00	16,30			Aprobado		
Altamirano Santos Alejandro	16,00	15,00	17,00	18,00	16,50			Aprobado		
Altamirano Jose Eduardo	16,00	15,00	18,00	16,00	16,30			Aprobado		
Carrillo Maria Ester	14,00	20,00	17,00	19,00	17,50			Aprobado		
Castillo Isolina Sofia	10,00	17,00	16,00	18,00	15,30			Suspense		
Cuevas Mercedes del Carmen	15,00	14,00	17,00	15,00	15,30			Suspense		
Diaz Irma del Carmen	10,00	18,00	17,00	16,00	15,30			Suspense		
Encina Isabel Zunilda	15,00	17,00	17,00	19,00	17,00			Aprobado		
Faundez Luis Ricardo	17,00	15,00	14,00	14,00	15,00			Suspense		
Gallardo Irma del Carmen	15,00	10,00	12,00	18,00	13,80			Suspense		

Figura B.13: Listado de calificaciones - Aplicación web
Elaborado por: Investigador

Como se puede observar al momento de seleccionar un paralelo se muestran una lista de calificaciones de los estudiantes de dicho paralelo, el docente al dar clic en el icono de modificación puede alterar el historial académico de un estudiante específico.

En la página siguiente un ejemplo de lo anteriormente explicado.

Escuela de Formación y Capacitación de Conductores Profesionales Ambato	
ESCUELA	Cronograma Perfil Paralelos
Alarcon Pereira, María del Carmen	
Trabajo en equipo	<input type="text" value="16.00"/>
Estudio de casos	<input type="text" value="15.00"/>
Prueba práctica	<input type="text" value="14.00"/>
Prueba teórica	<input type="text" value="20.00"/>
Promedio	<input type="text" value="16.30"/>
Supletorio	<input type="text"/>
Nota final	<input type="text"/>
<input type="button" value="Guardar"/>	

Figura B.14: Actualización de calificaciones - Aplicación web
Elaborado por: Investigador

En el caso de que un estudiante no haya alcanzado el mínimo necesario para aprobar un módulo su estado quedaría en suspenso hasta que se le tome una prueba supletorio, en el sistema el docente también tiene la posibilidad de ingresar esta

nota final la misma que al igual que demás pueden ser consultadas por el estudiante desde su smartphone.

El docente como parte de sus deberes en la institución entrega un informe académico de cada uno de los paralelos a su cargo donde constan todas las notas obtenidas a lo largo del curso. El sistema es capaz de generar un reporte como se muestra en las siguientes imágenes.

Escuela de Formación y Capacitación de Conductores Profesionales Ambato

ESCUELA Cronograma Perfil Paralelos - Salir

LISTADO DE CALIFICACIONES

° Paralelo: D - Convalidado Nocturno
° Modulo: Educación Vial

Descargar en PDF

Nombre	Trabajo en equipo	Estudio de casos	Prueba práctica	Prueba teórica	Promedio	Supletorio	Nota final	Estado	Operación
Altamirano Guerrero Alex Fabian	10,00	10,00	15,10	20,00	13,80			Suspenso	
Arcos Guerrero Luis Arturo	15,00	16,00	18,00	20,00	17,30			Aprobado	
Espin Montero Darwin David	16,00	18,00	15,00	18,00	16,80			Aprobado	
Navas Navarrete Daniel Javier	12,00	18,00	15,00	12,00	14,30			Suspenso	
Ortiz Castro Juan Carlos	18,00	19,00	15,00	17,00	17,30			Aprobado	
Peña Pico Diego Armando	18,00	15,00	19,00	15,00	16,80			Aprobado	
Velastegui Saltos Edison Rafael	19,00	18,00	17,00	15,00	17,30			Aprobado	

Figura B.15: Informe de calificaciones 1 - Aplicación web
Elaborado por: Investigador

Calificaciones 1 / 1

ESCUELA DE FORMACIÓN Y CAPACITACIÓN DE CONDUCTORES PROFESIONALES "AMBATO"

CUADRO GENERAL DE CALIFICACIONES

PARALELO: "D - Convalidado" PERIODO: 2016-2017 JORNADA: Nocturno
MÓDULO: Educación Vial

N1: Trabajo en equipo, N2: Estudio de casos, N3: Prueba práctica, N4: Prueba teórica, /4: Promedio, SP: Supletorio, NF: Nota Final

Nº	Nombres	N1	N2	N3	N4	/4	SP	NF
1	Altamirano Guerrero Alex Fabian	10,0	10,0	15,1	20,0	13,8		
2	Arcos Guerrero Luis Arturo	15,0	16,0	18,0	20,0	17,3		
3	Espin Montero Darwin David	16,0	18,0	15,0	18,0	16,8		
4	Navas Navarrete Daniel Javier	12,0	18,0	15,0	12,0	14,3		
5	Ortiz Castro Juan Carlos	18,0	19,0	15,0	17,0	17,3		
6	Peña Pico Diego Armando	18,0	15,0	19,0	15,0	16,8		
7	Velastegui Saltos Edison Rafael	19,0	18,0	17,0	15,0	17,3		
8	Villacres Andachi Danny Sebastian	18,0	19,0	20,0	14,0	17,8		

DOCENTE: Gallardo Honorina FIRMA: _____
FECHA DE PRESENTACION: _____

Figura B.16: Informe de calificaciones 2 - Aplicación web
Elaborado por: Investigador

Luego de la generación del informe de calificaciones el docente tiene la posibilidad de imprimirla directamente desde el navegador para posteriormente entregarla en Secretaría General.