



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

DESARROLLAR UN SOFTWARE EMPRESARIAL PARA MONITOREAR AVANCES EN EL
DESARROLLO DE PROYECTOS DE SOFTWARE.

Trabajo de Titulación presentado en conformidad con los requisitos establecidos
para optar por el título de Ingeniero en Sistemas de Computación e Informática.

Profesor Guía

MSc. Santiago Ramiro Villarreal Narvárez

Autor

Rafael Eduardo Falconí Salazar

Año

2016

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con el estudiante, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”

Santiago Ramiro Villarreal Narváz

Master en Ciencias, Tecnología y Salud mención Informática

CI. 1713980074

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.

Rafael Eduardo Falconí Salazar

CI. 171799368-5

AGRADECIMIENTOS

Agradezco a mi familia por darme la oportunidad de estudiar la carrera de Ingeniería en Sistemas de Computación e Informática y por darme la fortaleza de acabarla.

DEDICATORIA

Dedico este trabajo a mi Madre,
Padre y Hermanos, que siempre me
apoyaron a lo largo de esta carrera

Resumen

El proyecto de titulación tiene como objetivo el desarrollo de un prototipo de sistema para el control de proyectos de la empresa *Nahia soluciones S.A.*

En la primera fase del proyecto se recolecta datos del manejo de los proyectos dentro de la empresa *Nahia Soluciones S.A.* Se recolectaron los requerimientos funcionales y no funcionales.

En la segunda fase se plantea la solución al problema, se crea una plantilla base para el control de los proyectos. Junto con los diagramas de base de datos, análisis de requerimiento, diagramas de arquitectura para la aplicación web y móvil.

En la tercera fase comienza el desarrollo de la aplicación móvil y web. Una vez concluido el desarrollo se implementa el sistema dentro del servidor de la empresa.

En la cuarta fase se realiza pruebas al sistema para comprobar si cumple con los requerimientos y funcionalidades descritas en la fase dos.

Abstract

The purpose of this project is to develop a prototype system to help with control and organization of the projects of *Nahia Solutions S.A. Company*.

The first phase of the Project is focused in collecting information about the company's project control. The functional and non-functional requirements were collected.

In the second phase of the project, a solution is proposed, by creating a base template to control projects, database diagrams, required analysis and the architecture diagram for the web and mobile APP.

In the third phase the development of the mobile and web APP will start. Once the development is finished, the system is implemented on the company's server.

On the fourth phase the system is tested to check if it meets the requirements and functionality described in phase two

ÍNDICE

Introducción	1
Antecedentes	1
Problema:	1
Alcance:	2
Justificación:	2
Objetivos:	3
Objetivo General:	3
Objetivos Específicos:	3
1 Marco Teórico	4
1.1 Proyecto	4
1.2 Gestión de proyecto	4
1.2.1 Gestión del riesgo	4
1.2.1.1 Identificación del riesgo	4
1.2.1.2 Análisis de riesgo:	5
1.2.1.3 Planeación de riesgo:	5
1.2.1.4 Monitoreo del riesgo:	6
1.2.2 Gestión del personal	6
1.2.2.1 Motivación del personal	6
1.2.3 Trabajo en Equipo	7
1.2.3.1 Selección de los miembros del grupo	7
1.2.3.2 Organización del grupo	7
1.2.3.3 Comunicaciones grupales	7
1.3 Control de proyecto	8
1.3.1 Proceso de software	8
1.3.1.1 Especificación de requerimientos	8
1.3.2 UML	9
1.3.3 Caso de uso	9
1.3.4 Especificación de casos de uso	11

1.3.5 Modelo entidad Relación	12
1.3.4.1 Entidad	12
1.3.4.2 Atributos	12
1.3.4.3 Relación.....	12
1.3.6 Metodología del desarrollo de software	13
1.3.6.1 Metodología ágil	13
1.3.6.2 Metodología tradicional.....	14
1.3.7 Tabla comparativa de metodologías de desarrollo	16
1.4 Aplicación Web	17
1.5 Arquitectura MVC	17
1.6 Phonegap.....	19
1.7 Aplicación Móvil	19
1.8 Android.....	20
1.9 Gestor de Versionamiento.....	21
1.10 Pruebas de desempeño	22
2 Diseño de solución	23
2.1 Datos de proyectos anteriores	23
2.2 Diseño de plantilla base para control de proyectos. Como alternativa de solución.	26
2.2.1 Plantilla base datos del proyecto.	26
2.2.2 Plantilla base de etapas.....	27
2.3 Análisis de Requerimientos.....	27
2.3.1 Aplicación Web	27
2.3.1.1 Requerimientos funcionales:	27
2.3.1.2 Requerimientos no funcionales:	29
2.3.2 Aplicación Móvil	29
2.3.2.1 Requerimientos funcionales	29
2.3.2.2 Requerimientos no funcionales	30
2.4 Diagrama de Base de Datos.....	31
2.4.1 Diagrama Entidad Relación	31

2.4.2 Diagrama Relacional.....	33
2.5 Arquitectura Aplicación Web.....	35
2.6 Arquitectura Aplicación Móvil	36
3.- Desarrollo de la aplicación	38
3.1 Casos de uso.....	38
3.1.1 Casos de uso aplicación web	38
3.1.2 Casos de uso aplicación móvil.....	44
3.2 Diagrama de secuencia MVC.....	48
3.2.1 Portal de administrador.....	48
3.2.2 Portal del desarrollador.....	49
3.3 Desarrollo de aplicación web.....	50
3.4 Desarrollo de Web Service para aplicación móvil	50
3.5 desarrollo de la aplicación móvil.....	50
4 Pruebas.....	50
4.2 Pruebas de caja negra.....	53
5 Conclusiones y Recomendaciones	54
5.1 Conclusiones	54
5.2 Recomendaciones	55
REFERENCIAS	56
ANEXOS	59

Índice de Figura

Figura 1 Casos de uso	10
Figura 2 Ejemplo de caso de uso específico	11
Figura 3 Ejemplo de modelo entidad relación	12
Figura 4 Ciclo de vida metodología RUP	15
Figura 5 diagrama del modelo MVC	18
Figura 6 Arquitectura phonegap	19
Figura 7 Arquitectura del sistema operativo Android	20
Figura 8 Mapa de navegación de la aplicación Claudia Capelo	23
Figura 9 Recopilación de requerimientos aplicación de servicios eléctricos. ...	24
Figura 10 División de etapas de colegio Los pinos.	25
Figura 11 Diagrama propuesto entidad relación.....	31
Figura 12 Diagrama modelo entidad relación por cambios de la empresa Nahia Solutions S.A.	32
Figura 13 primer diagrama relacional.....	33
Figura 14 Diagrama relacional por cambios por la empresa Nahia Solutions. .	34
Figura 15 Arquitectura de la aplicación web.....	35
Figura 16 Arquitectura de aplicación móvil.....	36
Figura 17 Casos de uso aplicación web.....	38
Figura 18 casos de uso aplicación móvil.....	44
Figura 19 Diagrama de secuencia MVC portal web del administrador.....	48
Figura 20 Diagrama MVC aplicación portal del desarrollador.	49

Índice de Tablas

Tabla 1 Comparativa metodologías ágiles y metodologías tradicionales	16
Tabla 2 Plantilla base de datos del proyecto.....	26
Tabla 3 Plantilla base de etapas	27
Tabla 4 Requerimientos funcionales aplicación móvil	29
Tabla 5 Requerimientos no funcionales	30
Tabla 6 Caso de uso específico de aplicación web.....	38
Tabla 7 Casos de uso específicos aplicación móvil.	44

Introducción

Antecedentes

Se ha visto a través de la historia como ha surgido la necesidad de gestionar proyectos y de controlarlos almacenando la información sobre los mismos. Conocer su alcance, el problema a resolver, los requisitos, las etapas y recursos que se tiene a disposición. En la actualidad esto sigue siendo una necesidad y es cada vez mayor, debido a que los proyectos se han vuelto más complejos y la información que se almacena es de mayor importancia. (Haughey, 2013)

Sin importar el cliente al que se dirige el proyecto sea este dirigido hacia una persona, a grupo o hacia una empresa la información del mismo siempre debe ser almacenada de manera metódica y ordenada. En la actualidad, gracias a la tecnología se ha simplificado la recolección y almacenamientos de los datos de los proyectos. Sin embargo, siempre es importante contar con un software que facilite este proceso, ya que hace más eficientes las tareas como generar los reportes personalizados y retroalimentar a la empresa las falencias de los proyectos y también sus fortalezas.

Nahia Soluciones S.A es una empresa que se dedica al desarrollo de aplicaciones móviles, y software para publicidad inteligente. Su sede se encuentra en la ciudad de Quito, Ecuador.

La empresa *Nahia Soluciones S.A* brinda su servicio para distintas plataformas como IOS o Android para pantallas táctiles o aplicaciones móviles.

Problema:

Desde sus inicios *Nahia Soluciones S.A* ha utilizado su propia metodología para la gestión y control de proyectos de software. Cuentan con hojas escritas a mano, donde se coloca cada proyecto con sus etapas de desarrollo y su descripción. Es aquí donde se identificó una falencia que puede optimizarse mediante una aplicación web donde se almacenarán los datos de los proyectos.

Alcance:

El alcance de este trabajo de titulación es desarrollar un prototipo de sistema de control y seguimiento para proyectos de software, mediante un MVC de PHP conectado a una base de datos MySQL en un servidor web para el área de desarrollo de la empresa *Nahia Soluciones S.A.* Al igual que una aplicación móvil para el seguimiento del avance de los proyectos, para el sistema operativo Android mediante el lenguaje phonegap.

Dentro del desarrollo del proyecto se creará una plantilla base para el control y gestión de los proyectos de software de la empresa *Nahia Solutions S.A.*

En el primer módulo se permitirá asignar a cada proyecto de software una descripción y su respectivo alcance. En el segundo módulo se permitirá asignar a cada proyecto los requisitos del mismo. El tercer módulo permitirá asignar las fases de desarrollo, su descripción, alcance y personal asignado. El cuarto módulo permitirá ingresar observaciones del control en cada fase del proyecto. El quinto módulo permitirá asignar un código fuente de cada etapa del proyecto y el código fuente final. Creando un repositorio de código y versionamiento. Finalmente, en el sexto módulo se permitirá desplegar reportes ágiles con filtros: nombre de proyecto, cliente, fase de desarrollo del proyecto, personal asignado tanto para la aplicación móvil como para la aplicación web.

Para el desarrollo del proyecto se utilizará la metodología RUP, ésta permite un mayor control en el desarrollo ya que se basa en un proceso de pruebas y retroalimentación para comprobar el cumplimiento de los objetivos.

Justificación:

Como se ha podido observar el área de desarrollo de la empresa *Nahia Soluciones S.A* no cuenta con un software que le permita gestionar sus proyectos mediante su metodología y tener un control sobre ellos. Lo que lleva a que no se tenga un control y una gestión óptima, ya que no se guardan datos importantes de los proyectos, como alcance, descripción y requisitos. Estos datos son indispensables para el control y gestión de un proyecto. (González, 2014)

La gestión de un proyecto es indispensable dentro de una empresa, ya que para poder competir en un mercado exigente se tiene que tener un control de los procesos, de la calidad del producto, del personal y sobre todo cumplir con metas y tiempos con los clientes. (studio, s.f.)

La aplicación que se propone implementar, permitirá a la empresa tener mejor control de sus proyectos de software al tener almacenado datos importantes y relevantes de los mismos, para poder hacer retroalimentación con ellos. Al igual que permitirá tener una mejor comunicación dentro del área de desarrollo ya que contarán con la información necesaria y tener un repositorio de código fuente y versionamiento.

Objetivos:

Objetivo General:

Desarrollar un sistema web de manejo de proyectos para el área de desarrollo de la empresa *Nahia Soluciones S.A* utilizando un MVC de PHP y MySQL.

Objetivos Específicos:

- Generar un repositorio de versionamiento y código fuente.
- Crear módulos para asignar a los proyectos de software: requerimientos, alcance, las etapas, lo que conlleva las etapas, el personal, tiempo estimado de ejecución.
- Poder observar el avance de los proyectos en teléfonos con el sistema operativo Android.

1 Marco Teórico

En este capítulo se describirán los diferentes términos que se utilizarán en esta tesis así como los que usa la empresa *Nahia Solutions S.A.* en el desarrollo de sus trabajos. Se empezará por los que aplican preferentemente a *Nahia Solutions S.A.*:

1.1 Proyecto

Un proyecto es la unión de actividades que lleva a cabo una persona o conjunto de personas para poder llegar a un objetivo. Este es único, tiene un tiempo definido con principio y fin. (Project Management Institut, 2015)

1.2 Gestión de proyecto

1.3

Es cuando la persona o el equipo utilizan sus conocimientos, habilidades, herramientas y técnicas en las actividades del proyecto para poder cumplir los requerimientos de éstos. (Project Management Institut, s.f.)

1.2.1 Gestión del riesgo

La gestión del riesgo es una de las labores más importantes que tiene que realizar el administrador del proyecto. La gestión del riesgo es anticiparse a los riesgos que el proyecto pudiera llegar a tener y tomar decisiones para evitarlos o que tengan el menor impacto posible. (Sommerville, 2011, p. 594)

Las etapas de la gestión del riesgo son:

1.2.1.1 Identificación del riesgo

Primera etapa de gestión de riesgo. En esta etapa se identifican los riesgos al desarrollo de software o a la compañía que lo desarrolla. La identificación de los riesgos puede ser llevada a cabo por un equipo o por el líder del proyecto. Existen los siguientes tipos de riesgos:

- Riesgos tecnológicos: Son los riesgos producidos por el software o hardware utilizados para el desarrollo del sistema.
- Riesgos personales: Son los riesgos producidos por el personal dentro del equipo de desarrollo.
- Riesgos organizacionales: Son los riesgos producidos por el entorno de la compañía que desarrolla el software.
- Riesgos de herramientas: Son los riesgos producidos por las herramientas del software o del software de soporte utilizado para el desarrollo del sistema.
- Riesgos de requerimientos: Son los riesgos producidos por cambios a los requerimientos pedidos por el cliente.
- Riesgos de estimación: Son los riesgos producidos por las estimaciones de recursos para la elaboración del sistema. (Sommerville, 2011, p. 598)

1.2.1.2 Análisis de riesgo:

Segunda etapa de la Gestión del riesgo. En esta etapa se analiza la probabilidad y gravedad de los riesgos. Para realizar esto se debe utilizar el juicio propio y la experiencia obtenida durante los proyectos anteriores y los problemas de éstos. Para esto se tienen que asignar los por cientos de la probabilidad de que ocurra el riesgo y los efectos de éstos, como por ejemplo: catastróficos, graves, tolerables o insignificantes. (Sommerville, 2011, pp. 598-599)

1.2.1.3 Planeación de riesgo:

Tercera etapa de la Gestión del riesgo. En esta etapa se desarrolla un plan para manejar a cada uno de los riesgos. Para realizar esto se tiene que plantear la acción a llevar a cabo para minimizar el impacto del riesgo si este llega a suceder. Existen tres tipos acciones que se pueden llevar a cabo:

- Estrategias de evitación: Esta estrategia es reducir la probabilidad de que el riesgo se lleve a cabo.

- Estrategias de minimización: Esta estrategia es reducir el impacto del riesgo sucede.
- Planes de contingencia: Esta estrategia es tener un plan cuando se produzca lo peor por un riesgo.

(Sommerville, 2011, pp. 600-601)

1.2.1.4 Monitoreo del riesgo:

Cuarta etapa de la Gestión del riesgo. En esta etapa se lleva a cabo el proceso de comprobar que no han surgido cambios los riesgos. (Sommerville, 2011, p. 602)

1.2.2 Gestión del personal

En la gestión de proyecto las personas son el activo más importante. Por lo que es importante asignar al personal responsabilidades dependiendo de sus habilidades y experiencia. Existen cuatro factores críticos en la gestión del personal:

- Consistencia: Debe existir un trato similar entre todo el personal.
- Respeto: El personal tiene distintas habilidades y eso se tiene que respetar.
- Inclusión: El personal es más efectivo cuando se siente escuchados y que sus opiniones se toma en cuenta. Esto se logra en un ambiente donde todas las opiniones son consideradas.
- Honestidad: Se tiene que ser honesto con las limitaciones del personal.

(Sommerville, 2011, pp. 602-603)

1.2.2.1 Motivación del personal

Se debe motivar a las personas que trabajan dentro del proyecto, para que estas contribuyan con lo mejor de sus habilidades a este. La motivación

consiste en organizar el trabajo y el ambiente laboral para que el personal logre desempeñarse efectivamente como sea posible. (Sommerville, 2011, pp. 603-604)

1.2.3 Trabajo en Equipo

En el desarrollo de software cuando el equipo de trabajo es grande se debe dividir en grupos más pequeños. Cada grupo es responsable de un parte del sistema final. Cada grupo no debe tener más de diez integrantes. Para conformar cada grupo tiene que tomarse en cuenta el equilibrio perfecto entre habilidades técnicas, experiencia y personalidades. (Sommerville, 2011, p. 604)

1.2.3.1 Selección de los miembros del grupo

El administrador o líder del proyecto tiene que crear un grupo unido y organizarlo para que puedan trabajar de manera efectiva. Esto implica que se tiene que buscar el equilibrio entre habilidades técnicas y responsabilidad. Esto se logra mediante complementar los conocimientos y habilidades del equipo de desarrollo. (Sommerville, 2011, p. 609)

1.2.3.2 Organización del grupo

El momento de organizar un equipo se debe tener en cuenta cómo dicha organización influirá en la toma de decisiones internamente. (Sommerville, 2011, pp. 610-611)

1.2.3.3 Comunicaciones grupales

Es fundamental que los miembros de los grupos se comuniquen de forma efectiva para poder resolver los problemas que surjan por cambios en el sistema o en el desarrollo del mismo. (Sommerville, 2011, pp. 613-614)

Ahora se desarrollarán los términos que se aplicarán principalmente pero no exclusivamente en la solución del proyecto de tesis.

1.3 Control de proyecto

El control de proyecto consiste en mantener alineado al proyecto con los objetivos y sus requerimientos. (SOMMERVILLE, 2005, p. 86)

1.3.1 Proceso de software

Es la secuencia de actividades para la elaboración de software. Tiene cuatro actividades fundamentales:

1. Especificación del software: Se delimita la funcionalidad del software y las restricciones de sus operaciones.
2. Desarrollo del software: se diseña y programa el software.
3. Validación del software: se comprueba el correcto funcionamiento del software.
4. Evolución del software: se modifica el software para acoplarse a los cambios de requerimientos del cliente o del mercado.

(SOMMERVILLE, 2005)

1.3.1.1 Especificación de requerimientos

Es el proceso donde se escribe un documento con los requerimientos del usuario y del sistema (SOMMERVILLE, 2005, p. 108)

1.3.1.1.1 Requerimientos funcionales

Describen los servicios que el sistema debe proveer, como debería reaccionar el sistema a situaciones específicas y particulares, así como lo que no debe hacer el sistema. (SOMMERVILLE, 2005, p. 110)

1.3.1.1.2 Requerimientos no funcionales

Son las restricciones de los servicios o funciones ofrecidas por el sistema. (SOMMERVILLE, 2005, p. 111)

1.3.2 UML

Es un lenguaje de modelado para sistemas de Software. Es utilizado por analistas funcionales para definir que hace el programa sin tener que programarlo. Para esto utilizan los siguientes diagramas y algunos más:

- Diagramas de casos de usos.
- Diagramas de clases.
- Diagramas de secuencia.
- Diagramas de colaboración.
- Diagramas de estado.

(Object Management Group, Inc, 2015), (Krall, s.f.)

1.3.3 Caso de uso

Los diagramas de caso de uso identifican a los actores implicados en una interacción con el sistema y nombre el tipo de interacción, utilizando el lenguaje UML. Se puede apreciar un ejemplo en la figura 1. (SOMMERVILLE, 2005, pp. 154-155)

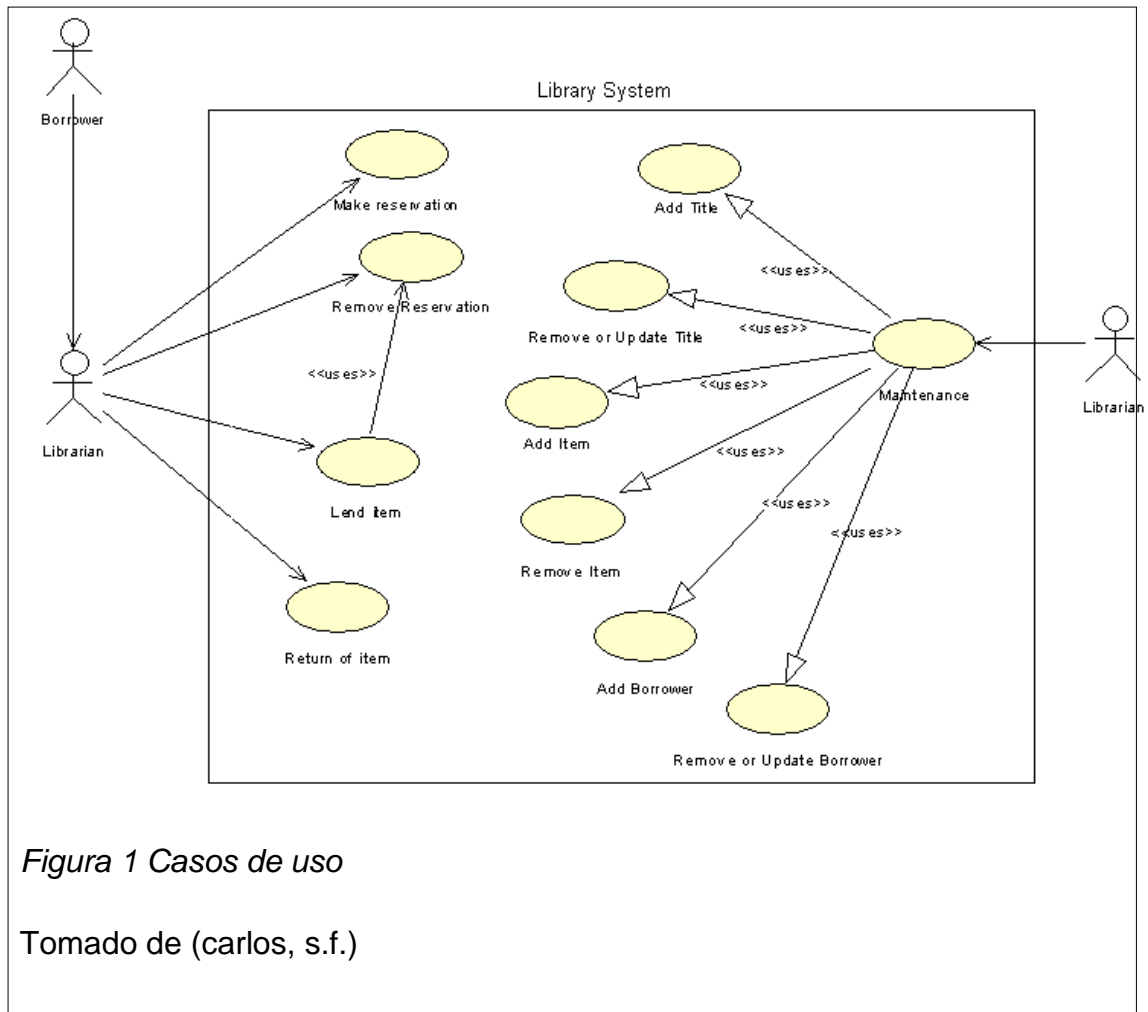


Figura 1 Casos de uso

Tomado de (carlos, s.f.)

1.3.4 Especificación de casos de uso.

Los casos de uso específicos detallan la información de un caso de uso tradicional. En la información incluye el nombre del caso de uso, los actores, la descripción, flujo básico, flujo alternativo, pre-condiciones, post-condiciones y varias más. Un ejemplo de un caso de uso específico se puede ver en la figura 2.(IBM, s.f.)

Nombre:	Registrar como socio / CU-1	
Actor:	Socio/Empleado	
Descripción:	Describe el proceso de realizar el Registro de Socios del Club	
Flujo Principal:	Eventos ACTOR	Eventos SISTEMA
	1. Verifica los documentos necesarios para la realización del registro respectivo	
	2. Activa función de registro de Socio	2. Muestra pantalla de registro de datos
	3. Realiza registro de datos del socio	3. Muestra pantalla de Registro y confirmación si los datos fueron correctamente registrados
	4. Almacenar los datos registrados.	4. El Sistema Con firma el Almacenamiento Datos
	5. Se emite el carnet al socio	5. El Sistema emite el carnet respectivo
Alternativa:	1. Se pide complementar documentación para su posterior registro en el sistema	
	2. Activa función de registro de Socio	2. Muestra pantalla de registro de datos
	3. Realiza registro de datos del socio	3. Verificar si se registraron los datos completos en caso de no ser así volver a introducir los que faltan
	4. Almacenar los datos registrados	4. El Sistema Con firma el Almacenamiento de Datos
Precondición:	El empleado ha sido habilitado para realizar registros de socios. La persona se informa sobre los documentos necesarios para ser habilitado como socio y los lleva.	
Poscondición:	El socio se encuentra registrado y obtiene su carnet de registro. Y está disponible para la solicitud de préstamo de videos.	
Presunción:	La base de datos de registro de socios y videos está disponible.	

Figura 2 Ejemplo de caso de uso específico

Tomada de (Izana, s.f.)

1.3.5 Modelo entidad Relación

El modelo entidad relación es un método para diseñar esquemas para base de datos relacionales. Se puede observar un ejemplo en la figura 3.

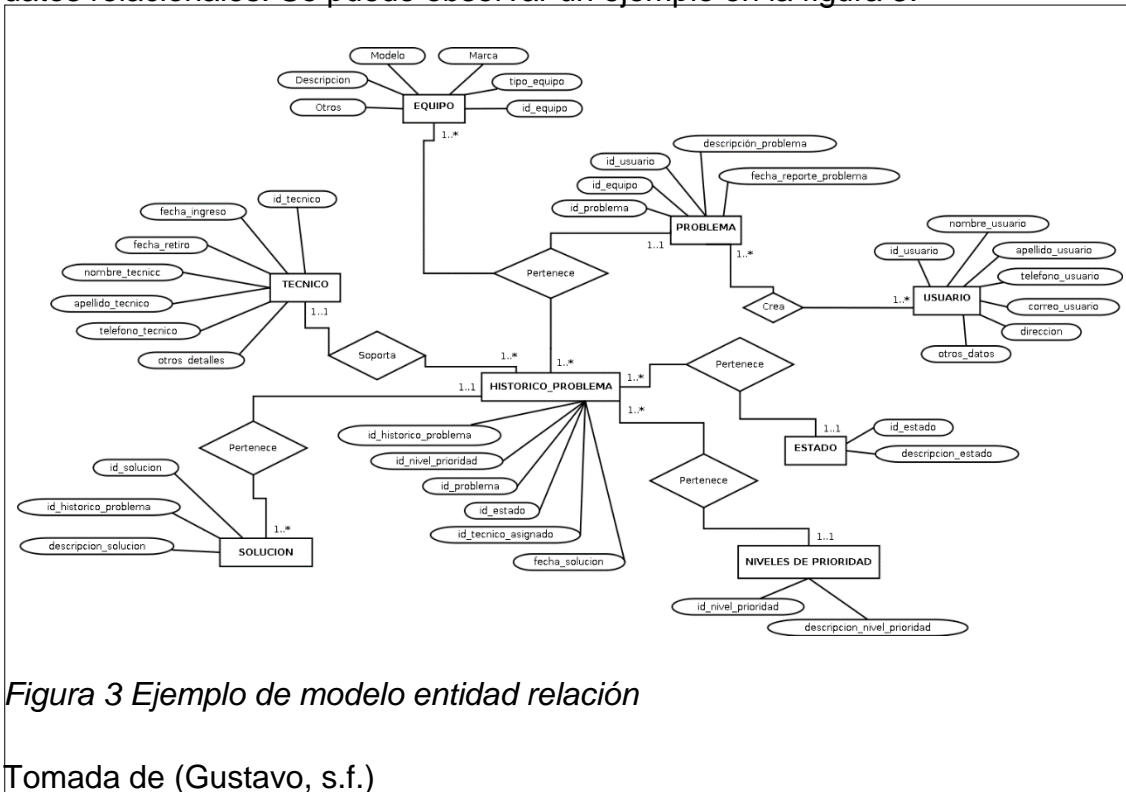


Figura 3 Ejemplo de modelo entidad relación

Tomada de (Gustavo, s.f.)

1.3.4.1 Entidad

La entidad representa cosas u objetos que se pueden diferenciar claramente entre sí.

1.3.4.2 Atributos

Los atributos identifican las características de la entidad.

1.3.4.3 Relación

Es el vínculo que define una dependencia entre varias entidades.

1.3.6 Metodología del desarrollo de software

La metodología es la representación simplificada de procesos para el desarrollo de software. (SOMMERVILLE, 2005, p. 358)

1.3.6.1 Metodología ágil

Permiten el incremento de especificaciones, desarrollo y entrega del software.

(SOMMERVILLE, 2005, pp. 358-361)

1.3.6.1.1 Programación extrema

Programación extrema o PX es una metodología ágil de desarrollo de software que cuenta con las siguientes prácticas básicas:

- **Equipo Completo:** El equipo está conformado por todas las personas que estén involucradas en el proyecto.
- **Planificación:** Se crean historias de usuario y se planifica en qué orden se hacen y las mini versiones.
- **Test del cliente:** El cliente propone pruebas para validar las mini-versiones.
- **Versiones pequeñas:** También conocidas como mini-versiones son versiones que se puedan producir en pocas semanas y sobre todo que el cliente pueda probarlas.
- **Diseño simple:** Mantener el código siempre sencillo.
- **Pareja de programadores:** Dos programadores por un ordenador y se intercambian las parejas una vez cada día.
- **Desarrollo guiado por las pruebas automáticas:** Se deben hacer pruebas automáticas y se deben realizar con la mayor frecuencia posible.
- **Integración continua:** Siempre se tiene que tener un ejecutable que funcione y cuando se tenga una nueva funcionalidad debe recopilarse y probar que funcione.

- El código es de todos: Todos pueden leer y escribir cualquier parte del código.
- Normas de codificación: Todos deben de codificar mediante un estándar cualquiera que este sea.
- Metáforas: Se tiene que utilizar una frase o un nombre para definir la funcionalidad de cada parte del código.
- Ritmo sostenible: Se tiene que mantener un ritmo continuo a lo largo del proyecto.

(Boliviana, s.f.)

1.3.6.2 Metodología tradicional

La metodología tradicional sigue por procesos rígidos, controlados y secuenciales. No permite saltarlos o retroceder hasta finalizarlos todos por completo. (SOMMERVILLE, 2005, p. 358)

1.3.6.2.1 RUP

Es una metodología tradicional que utiliza el UML. Permite varias interacciones en sus fases para que éstas queden completas y de la mejor manera. Las cuatro fases de RUP son:

1. Fases de Inicio: En esta fase se define el alcance del proyecto, los riesgos y análisis de requerimientos.
2. Fase de elaboración: En esta fase se definen los casos de uso para definir la arquitectura del sistema.
3. Fase de desarrollo: En esta fase se comienza el desarrollo del software se definen y aclaran los últimos requerimientos.
4. Fase de Cierre: En esta fase se asegura que el software esté disponible para los usuarios finales, se ajustan los errores y defectos encontrados en las pruebas de aceptación, capacitar a los usuarios y proveer el soporte técnico. Se verifica que el software cumpla con las especificaciones entregadas.

(wikispaces, s.f.)

Se puede apreciar el tiempo y la sucesión de estas fases en la figura 4/

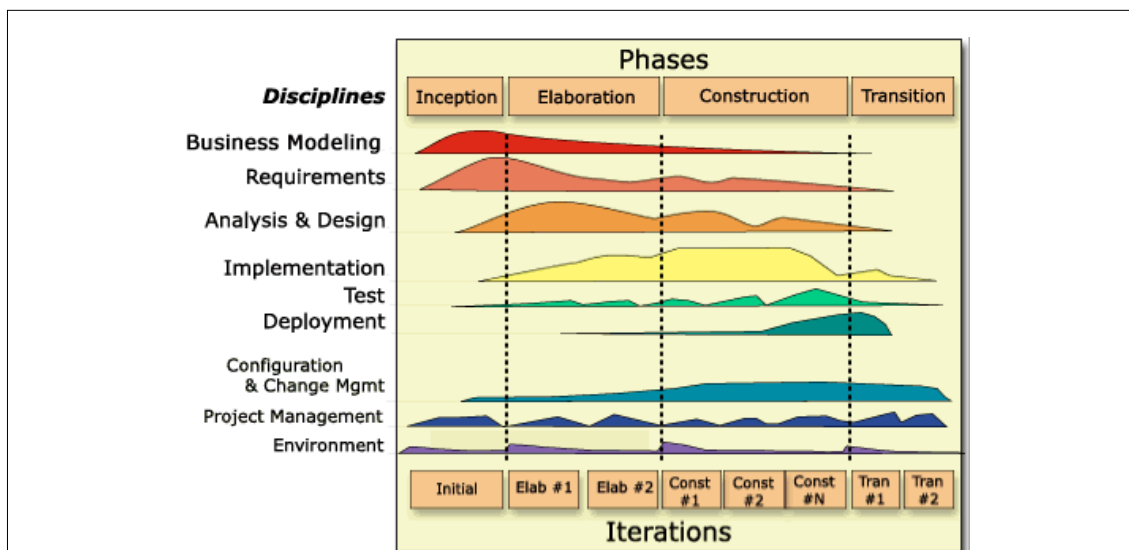


Figura 4 Ciclo de vida metodológica RUP

Tomada de (wikispaces, s.f.)

1.3.7 Tabla comparativa de metodologías de desarrollo

Como se puede apreciar en la Tabla 1 se puede apreciar las diferencias entre las metodologías tradicionales y las metodologías ágiles.

Tabla 1 Comparativa metodologías ágiles y metodologías tradicionales

MÉTODOS ÁGILES	MÉTODOS TRADICIONALES
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.
Impuestas internamente (por el equipo).	Impuestas externamente.
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas.
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos.	Más artefactos.
Pocos roles	Más roles.
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos.

Tomada de(wikispaces, s.f.)

1.4 Aplicación Web

Una aplicación web es un software que se encuentra dentro de un servidor web que es ejecutado en un navegador y que tiene la ventaja de ser multiplataforma. Los datos y la información son almacenados en los servidores y son desplegados dependiendo de las peticiones del usuario. (terms, 2014), (libre, s.f.), (group, s.f.)

1.5 Arquitectura MVC

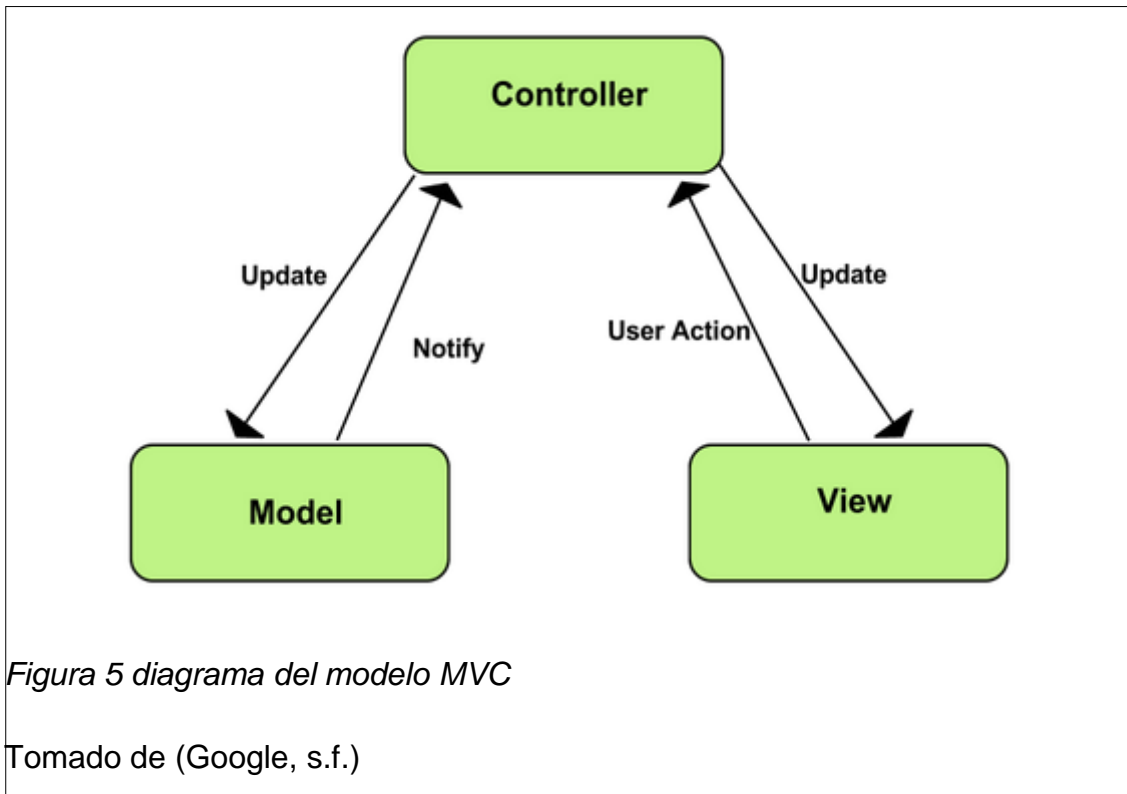
La arquitectura MVC es una arquitectura basada en la programación en capas. Está compuesta por tres componentes; modelo, vista y controlador.

Capa Modelo: en esta capa es donde los datos del objeto son almacenados. Al igual que los datos son manejados, ya sea para modificarlos, crearlos, editarlos o eliminarlos. Esto a petición de la vista o del controlador.

Capa Vista: Es la interface gráfica que interactúa con el usuario. Comúnmente contiene elementos HTML, JavaScript CSS.

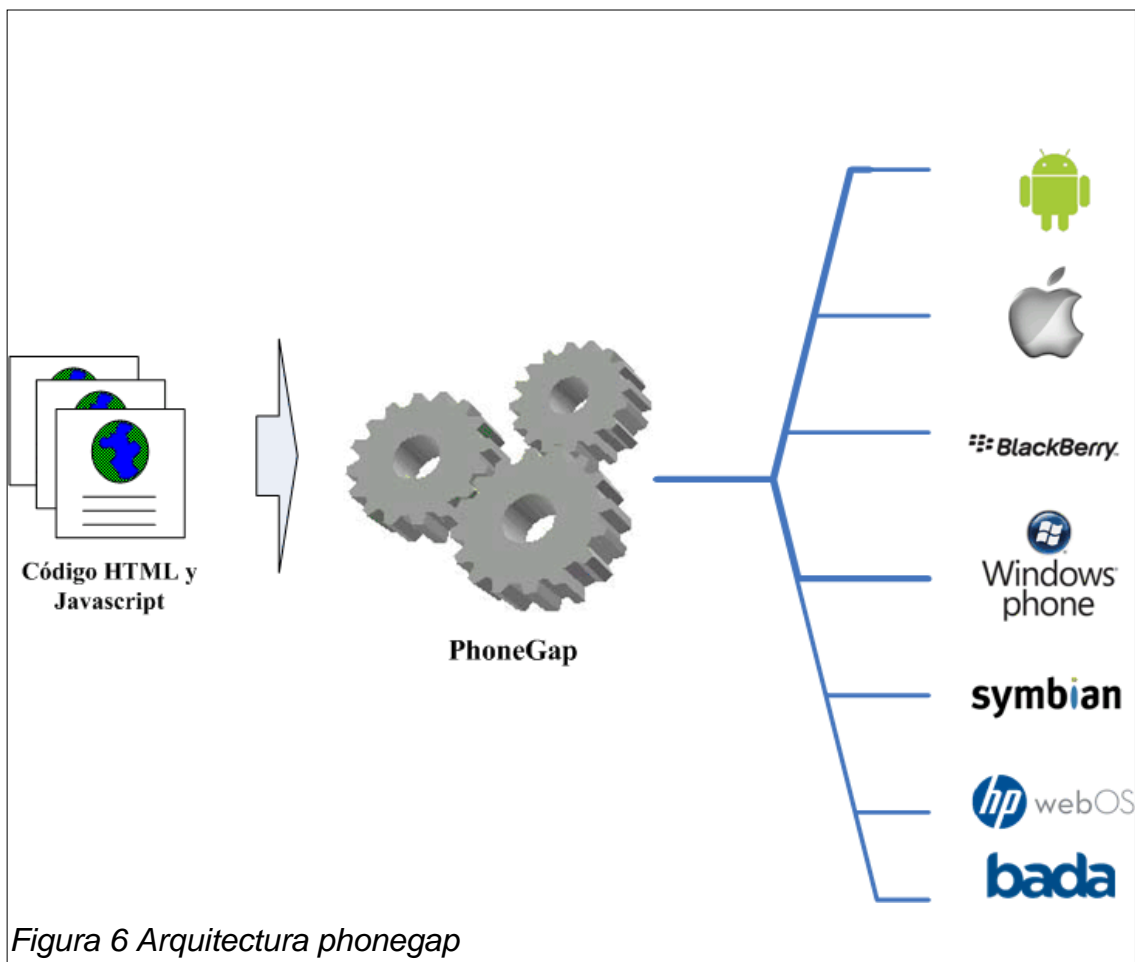
Capa Controlador: El controlador es el encargado de la interacción de las capas de vista y modelo. Esto ya que cuando se modifica la vista actualiza el modelo y viceversa. (Google, s.f.), (Point, s.f.)

La interacción de estas capas se puede apreciar en la figura 5



1.6 Phonegap

Phonegap es un framework de Apache Cordova, desarrollado por Niobi y adquirido por Adobe en el 2011. Phonegap es utilizado para desarrollo de aplicaciones móviles para los siguientes sistemas operativos: Android, iOS, Windows Phone, BlackBerry OS, Web OS, Symbian o Bada. Esto significa que Phonegap es multiplataforma. Utiliza HTML5, CSS3 y Javascript (jQuery Mobile) como elementos para el desarrollo de las aplicaciones.



1.7 Aplicación Móvil

Una aplicación móvil es un programa que es ejecutado en un dispositivo móvil como teléfonos móviles, tablets etc. Existen dos tipos de aplicaciones móviles:

- APP Nativas: este tipo de aplicaciones son las cuales son desarrolladas específicamente para un sistema operativo móvil mediante el software

development kit o SDK oficial de este. Por esta razón si se desea que la aplicación móvil sea para varios sistemas operativos tiene que ser desarrollado para cada uno de éstos.

- Aplicaciones Híbridas: Estas aplicaciones son desarrolladas con lenguajes y compones web como HTML, Javasript y CSS, pero con la ventaja de poder utilizar y acceder al hardware del dispositivo móvil e interactuar con éste.

(AlertaenLinea, s.f.), (Talent, s.f.)

1.8 Android

Android es un sistema operativo basado en el núcleo de Linux, diseñado por Google. Android controla e interactúa con el hardware del dispositivo mediante Java. El código fuente de Android es abierto. Algunas características de Android son:

1. Utiliza como motor de base de datos SQLite.
2. Soporta pantalla táctil multitouch.
3. Máquina virtual Dalvik.

Mientras que la arquitectura del sistema operativo se puede observar en la figura 7.



Figura 7 Arquitectura del sistema operativo Android.

Tomado de (EazyTuts, s.f.)

- Aplicaciones: Aquí es donde se encuentran las aplicaciones que vienen por defecto en el sistema operativo y las cuales son instaladas por el usuario.
- Framework de Aplicaciones: Aquí es donde las aplicaciones de Android son ejecutadas y manejadas.
- Librerías: Aquí se encuentran las librerías que utiliza Android, las cuales fueron escritas utilizando los lenguajes C/C++.
- Tiempo de ejecución de Android: Esta al mismo nivel que las librerías utilizadas por Android. Aquí se almacena las librerías base de Java y la máquina virtual Dalvik.
- Núcleo Linux: El núcleo utilizado es el de Linux 2.6. Aquí se encuentran los drivers para que el hardware pueda ser utilizado por cualquier aplicación que lo requiera.

En la siguiente lista podremos observar las versiones de Android que han sido lanzadas.

- Android 1.6 Donut.
- Android 2.0 Eclair.
- Android 2.2 Froyo.
- Android 2.3 Gingerbread.
- Android 3.0 Honeycomb.
- Android 4.0 Ice cream Sandwich.
- Android 4.1 Jelly Bean.
- Android 4.1 KikKat
- Android 5.0 Lollipop.

1.9 Gestor de Versionamiento

Es un software que permite registrar los cambios que se realizan sobre un archivo o conjunto de estos a lo largo del tiempo. Esto permite recuperar el código fuente, contenido multimedia, etc. En la versión que el usuario necesite.

Existen tres tipos de sistemas de controles; sistema de control de versiones local, sistemas de control de versiones centralizados, sistemas de control de versiones distribuidos. (Scott Chacon, Juan Murua, Alejandro Fernández, Javier Eguiluz, s.f.)

1.10 Pruebas de desempeño

Las pruebas de desempeño sirven para evaluar el comportamiento de la aplicación en condiciones extremas como altos niveles de usuarios que lo utilizan simultáneamente. Permitiendo ajustar y mejorar el rendimiento de la aplicación. (e-quallity, s.f.)

En resumen:

- La empresa *Nahia Solutions S.A.* tiene claro en su gestión lo que es un proyecto y lo que implica ejecutarlo y administrar sus riesgos.
- La empresa *Nahia Solutions S.A.* no incorpora de manera sistemática los conceptos de control del desarrollo de los proyectos que ejecuta, en cada una de sus etapas.

2 Diseño de solución

En este capítulo se expondrá cómo la empresa *Nahia Solutions* gestiona sus proyectos y, a partir de esto, se planteará una solución para mejorar este proceso incorporando la idea de control.

2.1 Datos de proyectos anteriores

Los pasos que sigue *Nahia Solutions S.A.* en la gestión de sus proyectos se puede mostrar así:

Los mapas de navegación de las posibles aplicaciones móviles son recogidos en una hoja A4, en la que se incorpora la información básica de la interacción de las eventuales diferentes pantallas. En dichas hojas resalta a simple vista que no constan datos importantes como medidas, código de colores y proporciones de cada elemento de las pantallas. Un ejemplo de lo dicho es la figura 8

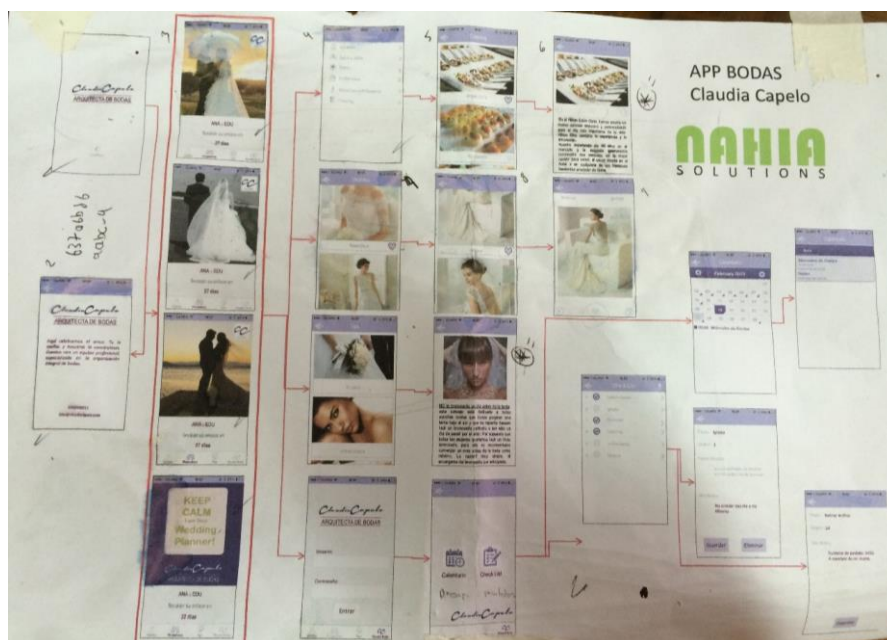


Figura 8 Mapa de navegación de la aplicación Claudia Capelo.

Los requerimientos de los clientes son apuntados en una hoja A3 sin aparente orden o estándar, quedando evidenciadas varias inconsistencias. De igual manera no se puede diseñar una solución efectiva para el problema ya que, por ejemplo, no se pueden detallar las funciones del sistema, ni definir su alcance y funcionalidades. Un ejemplo de esto es la figura 9.

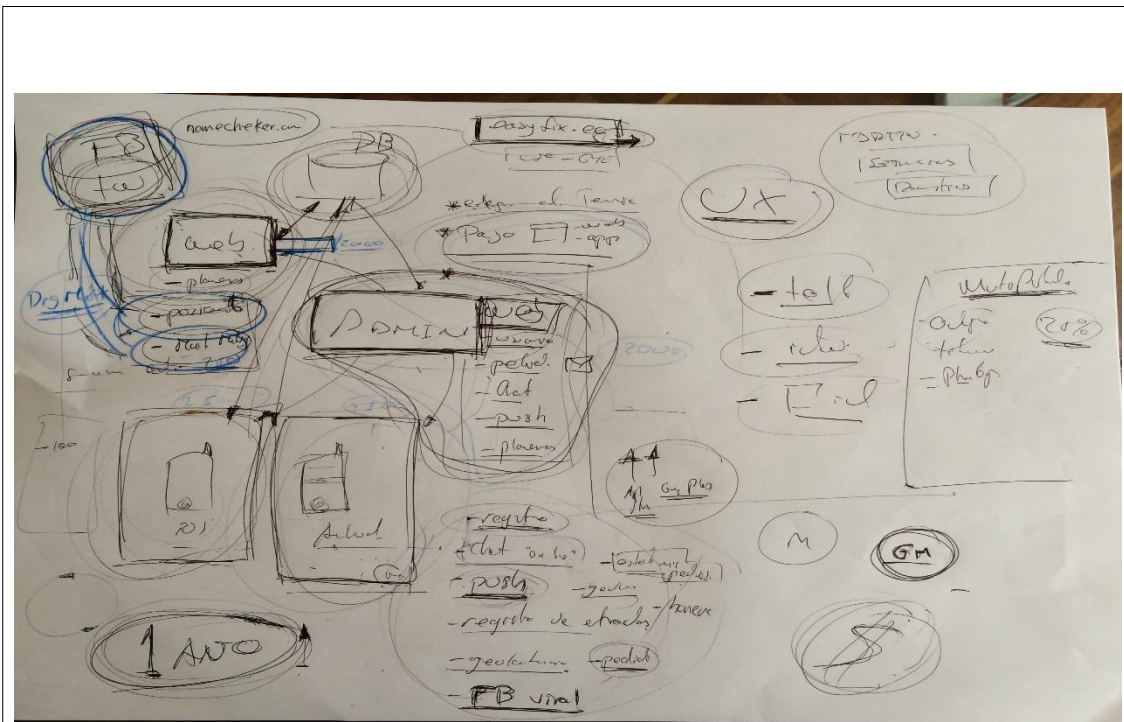


Figura 9 Recopilación de requerimientos aplicación de servicios eléctricos.

La empresa además utiliza hojas A4 en las que divide el proyecto en etapas, lo que conlleva, el tiempo en el que se las tiene que cumplir, pero no quienes están a cargo de éstas. Esto crea una inconsistencia ya que no se detalla a profundidad el alcance de cada etapa ni quienes están a su cargo, o qué presentables tienen que existir. En la figura 10 se puede ver un ejemplo de esto.

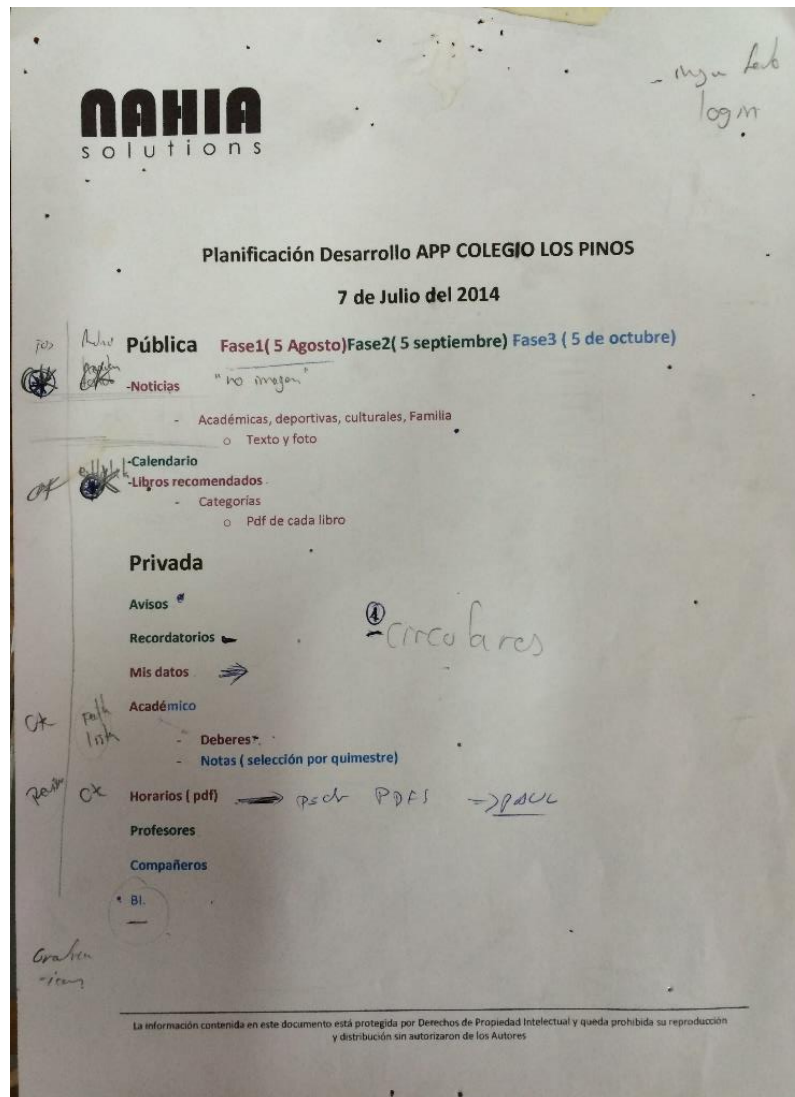


Figura 10 División de etapas de colegio Los pinos.

De todo esto se puede apreciar que la compañía *Nahia Solutions S.A.* no tiene un estándar para el control de sus proyectos y no se guarda un registro de sus avances o inconvenientes.

2.2 Diseño de plantilla base para control de proyectos. Como alternativa de solución.

Al ver las falencias de la compañía *Nahia Solutions S.A.* en el control de avances de sus proyectos se planteó una plantilla base para poder controlar esta tarea con mayor eficiencia y guardando los datos, a partir de los conceptos detallados en el capítulo anterior.

2.2.1 Plantilla base datos del proyecto.

En las tablas 2 y 3 se puede observar la plantilla que se propuso para el control de avance de proyecto según las necesidades de la empresa *Nahia Solutions S.A.*

Tabla 2 Plantilla base de datos del proyecto.

Nombre	
Cliente	
Alcance	
Plataforma	
Requerimientos Funcionales	
Requerimientos No Funcionales	
Tiempo de desarrollo	
Número de etapas	
Responsables/Cargos	

2.2.2 Plantilla base de etapas.

Tabla 3 Plantilla base de etapas

Número de etapa	
Alcance	
Plataformas	
Observaciones	
Responsables/Cargos	
Código/versión/plataforma	

2.3 Análisis de Requerimientos

Una vez aprobada la plantilla base se procedió a levantar los requerimientos del sistema para utilizar la misma, de la siguiente forma:

2.3.1 Aplicación Web

2.3.1.1 Requerimientos funcionales:

Número	Requerimiento	Funcionalidad
RF1	Creación de usuario	El sistema permitirá la creación de usuarios y los almacenará en la base de datos. Con su id, nombre, correo y contraseña.
RF2	Login	El sistema tendrá un formulario para el ingreso, el cual será comprobado con los usuarios almacenados en la base de datos
RF3	Modificación de usuarios	El sistema permitirá la modificación de los datos de los usuarios almacenados en la base de datos.
RF4	Eliminación de usuarios	El sistema permitirá la eliminación de los usuarios almacenados en la base de datos.
RF5	Creación de proyectos	El sistema permitirá la creación de proyectos y los almacenará en la base de

		datos. Con su id, nombre, usuarios asignados, requerimientos, tiempo, etapas. Además, creará una carpeta con el nombre del proyecto
RF6	Modificación del proyecto	El sistema permitirá la modificación de los datos de los proyectos almacenados en la base de datos.
RF7	Eliminación de usuarios	El sistema permitirá la eliminación de los proyectos almacenados en la base de datos.
RF8	Subir versión de código	El sistema permitirá subir el código del proyecto a su carpeta con su respectivo número de versión. Almacenará en la base de datos el path del código, número de versión, el proyecto al que pertenece y a que plataforma pertenece.
RF9	Listado de versiones	El sistema permitirá ver el número de versión del código, a que proyecto y plataforma pertenece.
RF10	Descargar Código	El sistema permitirá descargar la versión del código que se necesite.
RF11	Reporte	El sistema permitirá desplegar reportes de los proyectos mediante los siguientes filtros: nombre de proyecto, cliente, fase de desarrollo del proyecto, personal asignado tanto para la aplicación móvil como para la aplicación web.

2.3.1.2 Requerimientos no funcionales:

Número	Requerimiento	Funcionalidad
RFN1	Multiplataforma	El sistema funcionará en los sistemas operativos; Windows, IOS
RFN2	Diseño	El sistema se acoplará a las diferentes resoluciones de pantalla.
RFN3	Cifrado de contraseñas	Las contraseñas serán almacenadas mediante el cifrado md5.
RFN4	Datos desplegados	Los datos serán desplegados según el usuario que iniciado sesión
RFN5	Disponibilidad	El sistema funcionará las 24 horas, 7 días a la semana mientras los servidores donde se encuentre estén funcionando correctamente y estén disponibles.
RFN7	Portabilidad	El sistema se encontrará disponible si el equipo tiene navegador web (Google Chrome, Firefox) e internet.
RFN8	Reportes	Los reportes desplegados serán gráficos en forma de barras.

2.3.2 Aplicación Móvil

2.3.2.1 Requerimientos funcionales

Tabla 4 Requerimientos funcionales aplicación móvil

Número	Requerimiento	Funcionalidad
RF1	Login	La aplicación móvil tendrá módulo con un formulario para el ingreso, el cual será comprobado con los usuarios

		almacenados en la base de datos
RF2	Reportes	La aplicación móvil tendrá un módulo que permitirá desplegar reportes de los proyectos mediante los siguientes filtros: nombre de proyecto, cliente, fase de desarrollo del proyecto, personal asignado tanto para la aplicación móvil como para la aplicación web.
RF3	Web service	La aplicación consultará los datos de los proyectos mediante web services.
RFN7	Portabilidad	La aplicación móvil se encontrara disponible si el equipo como sistema operativo los anteriores ya mencionados y acceso al internet.

2.3.2.2 Requerimientos no funcionales

Tabla 5 Requerimientos no funcionales

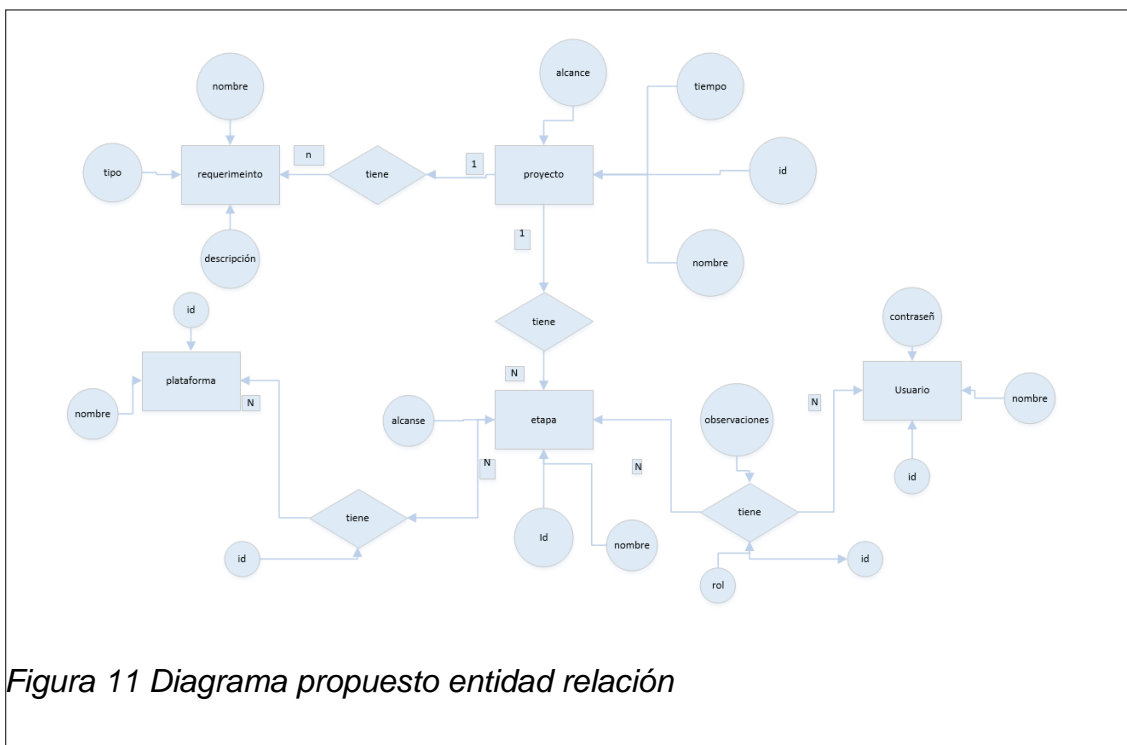
Número	Requerimiento	Funcionalidad
RFN1	Plataforma	El sistema funcionará en los sistemas operativos; Android 2.3 en adelante.
RFN2	Diseño	El sistema se acoplará a las diferentes resoluciones de pantalla.
RFN3	Reportes	Los reportes desplegados serán gráficos en forma de barras.
RFN4	Disponibilidad	El sistema funcionará las 24 horas, 7 días a la semana mientras los servidores donde se encuentre estén funcionando correctamente y estén disponibles.

2.4 Diagrama de Base de Datos

El diagrama de base de datos se creó mediante los requerimientos de la empresa *Nahia Solutions S.A.* y con la plantilla base para el control de proyectos propuesta.

2.4.1 Diagrama Entidad Relación

En la figura 11 se puede apreciar el modelo entidad relación planteado inicialmente para el sistema de control de proyectos.



La figura 12 se puede apreciar el modelo entidad relación final a petición del gerente de la empresa *Nahia Solutions S.A.* debido a que el personal del área de desarrollo es menor y no se piensa incrementar por lo menos en 1 año.

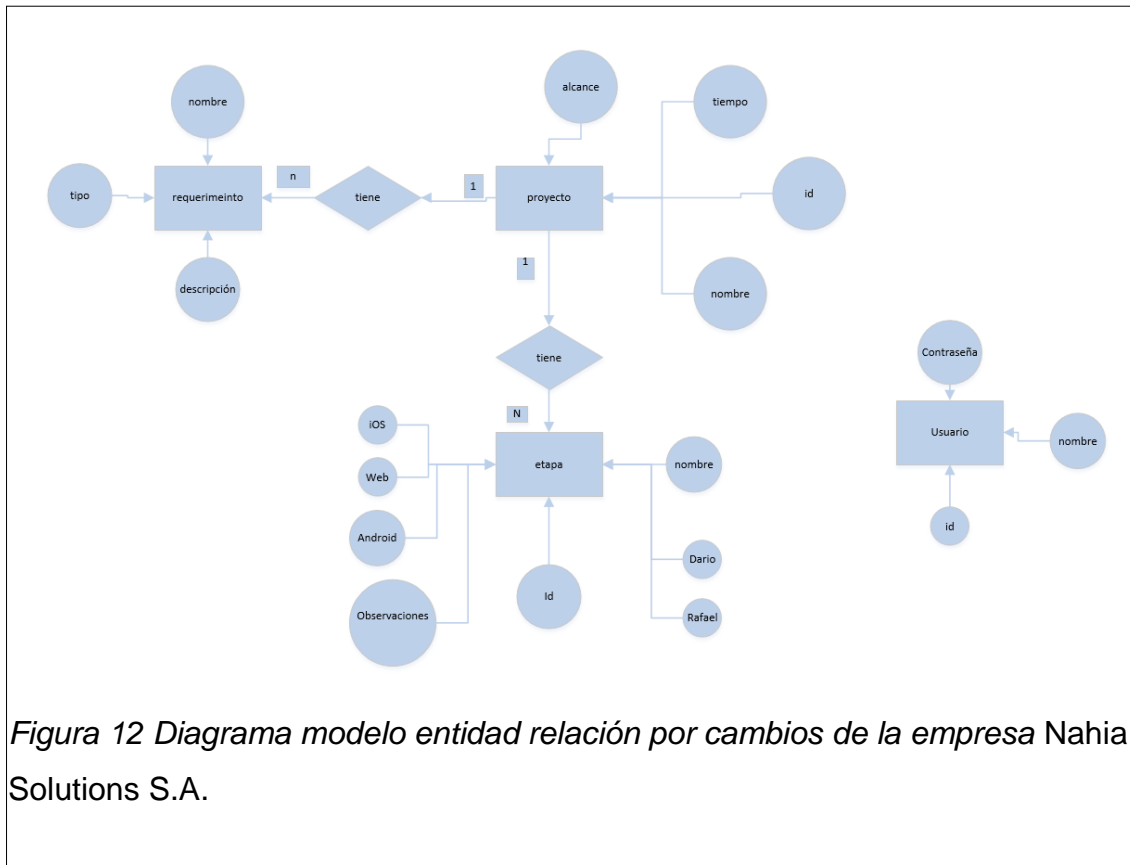


Figura 12 Diagrama modelo entidad relación por cambios de la empresa Nahia Solutions S.A.

2.4.2 Diagrama Relacional

En la figura 13 se puede apreciar el modelo relacional a partir del modelo entidad planteada inicialmente relación previamente realizado.

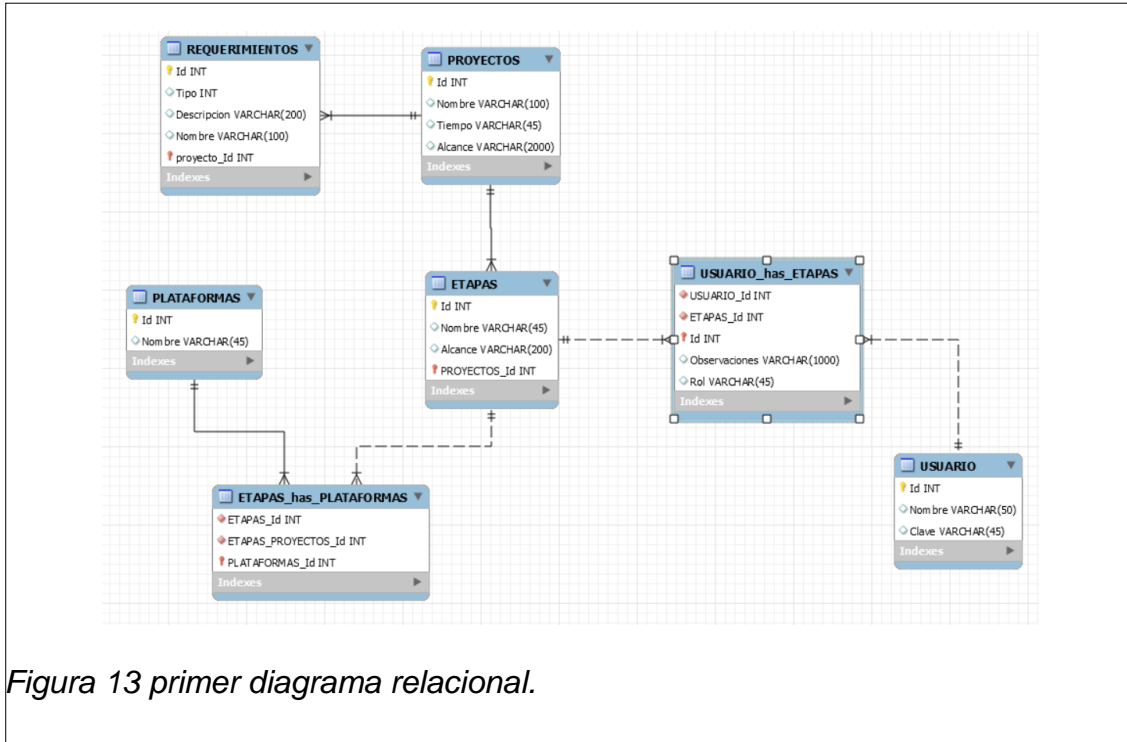


Figura 13 primer diagrama relacional.

En la figura 14 se puede ver el modelo relacional a partir del modelo entidad final por los requerimientos del gerente de la empresa *Nahia Solutions S.A.*

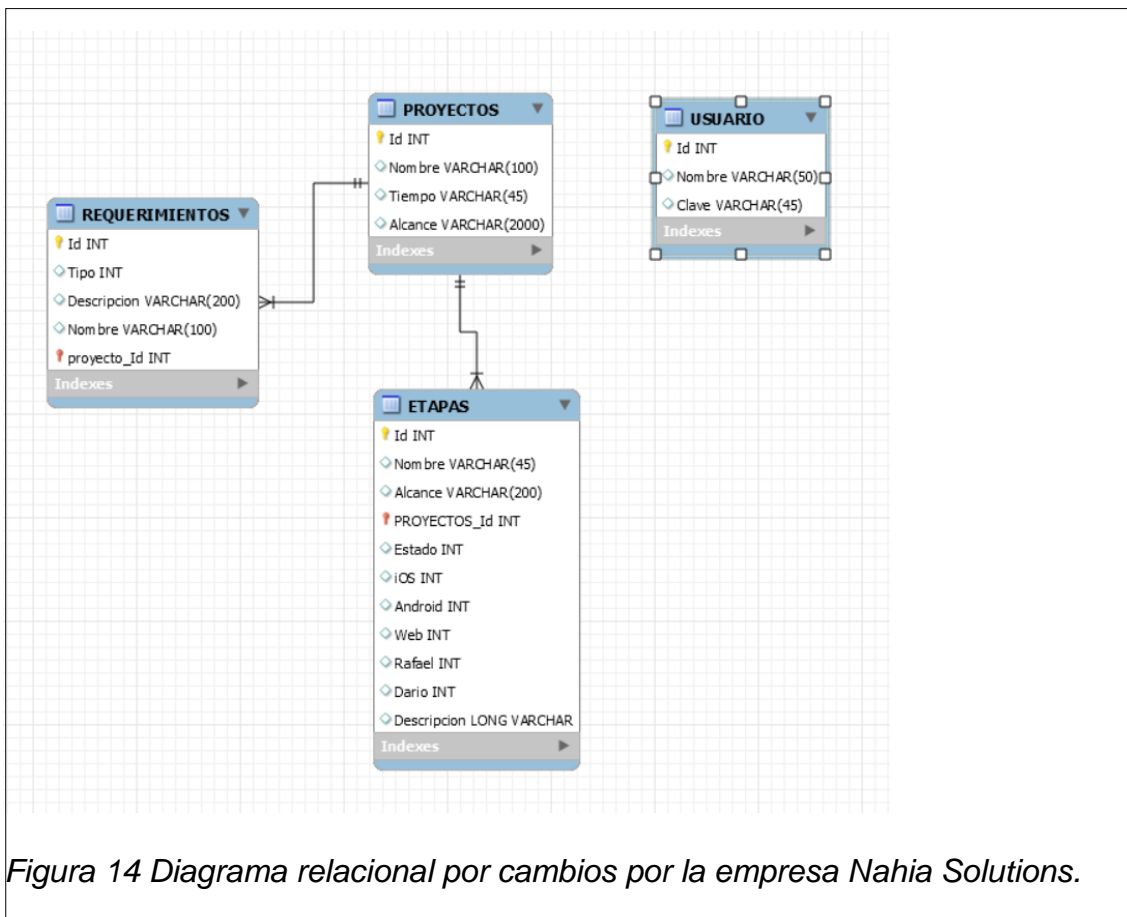


Figura 14 Diagrama relacional por cambios por la empresa Nahia Solutions.

2.5 Arquitectura Aplicación Web

En la figura 15 se podrá apreciar cómo está estructurada la aplicación web y que del lado del cliente estará el navegador web para poder utilizar la aplicación. En el lado del servidor web estarán cuatro capas: la primera constará de la interface gráfica que contará con elementos HTML, Javascript y CSS3; en la segunda se llamarán a los datos desde la primera capa para poder editarlos, crearlos, modificarlos o eliminarlos según lo necesite el usuario; en la tercera capa se encontrarán los datos de conexión con el repositorio de datos, en este caso una base de datos; y, en la cuarta capa, se permitirá la comunicación entre las tres anteriores y se implementará la seguridad.

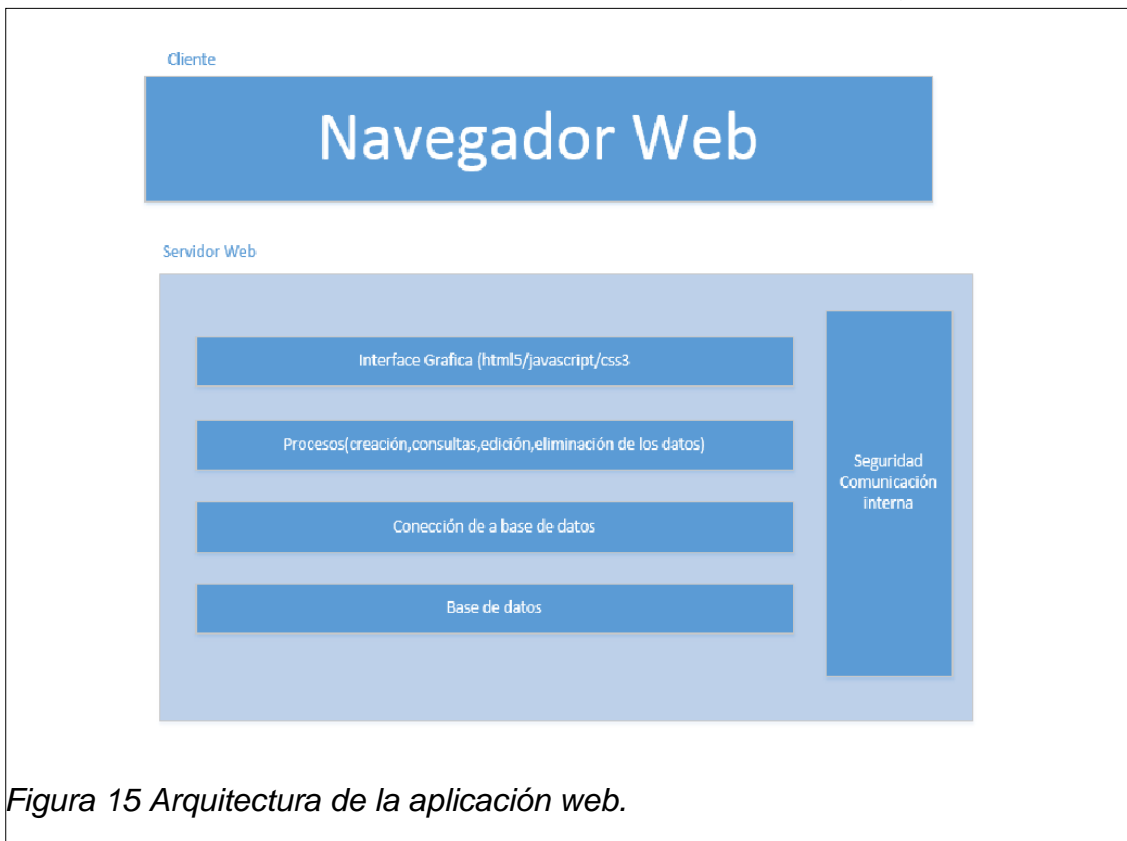


Figura 15 Arquitectura de la aplicación web.

2.6 Arquitectura Aplicación Móvil

En la figura 16 se puede apreciar cómo estará estructurada la aplicación web. Aquí, en el lado del cliente estará la aplicación móvil donde consta la UI que cuenta como HTML/CSS3; en la capa de negocios estarán las peticiones de datos mediante jQuery y Ajax; y, en el lado del servidor web estarán los Web Services, la conexión a la base de datos para obtener los datos pedidos por la capa de negocios de la APP.

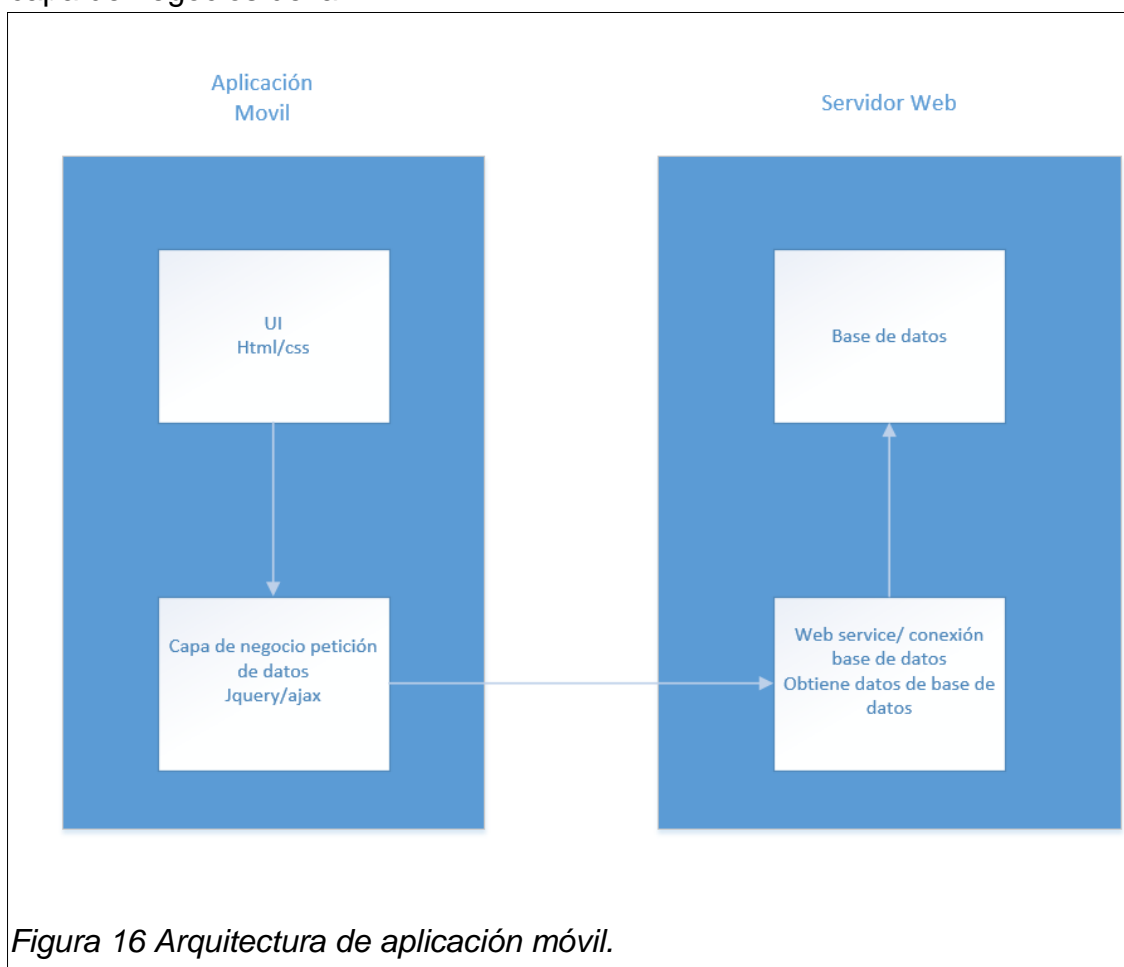


Figura 16 Arquitectura de aplicación móvil.

En resumen:

En la metodología de trabajo de *Nahia Solutions S.A.* destaca el uso de hojas sueltas de trabajo en las que se evidencia la falta de datos como medidas, código de colores y proporciones de cada elemento de las pantallas; además, varias inconsistencias toda vez que no se puede diseñar una solución efectiva para el problema ya que, por ejemplo, no se pueden detallar las funciones del

sistema, ni definir su alcance y funcionalidades; y, no se detalla a profundidad el alcance de cada etapa ni quienes están a su cargo, o qué presentables tienen que existir. De todo esto se puede apreciar que la compañía *Nahia Solutions S.A.* no tiene un estándar para el control de sus proyectos y no se guarda un registro de sus avances o inconvenientes.

Por ello, es claro que la solución a este problema está en seguir con rigor la metodología planteada en el capítulo I tal y como se expresa en las tablas anteriormente desarrolladas.

3.- Desarrollo de la aplicación

En este capítulo se desarrollarán los casos de usos, secuencia y el desarrollo de la aplicación.

3.1 Casos de uso

En las siguientes imágenes se podrán observar los casos de uso para la aplicación web y la aplicación móvil.

3.1.1 Casos de uso aplicación web

En la figura 17 se puede apreciar los casos de uso para la aplicación web

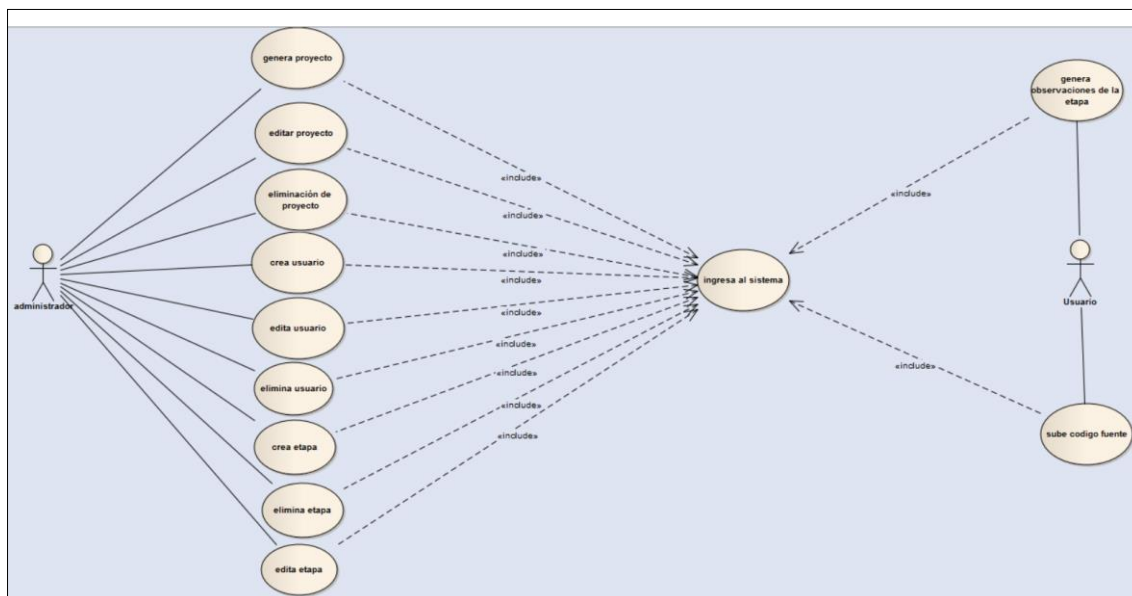


Figura 17 Casos de uso aplicación web.

Tabla 6 Caso de uso específico de aplicación web.

Caso de Uso	Ingresa al sistema
Actor	Administrador/Usuario
Descripción	El usuario/administrador ingresa su nombre de usuario y contraseña para ser validadas e ingresar al sistema.
Flujo básico	El sistema pide al usuario sus credenciales. El usuario

	indica al sistema sus credenciales. El sistema valida que son correctas las credenciales y les permite ingresar a los módulos privados.
Flujos Alternos	El sistema pide al usuario sus credenciales. El usuario indica al sistema sus credenciales. El sistema valida que no son correctas las credenciales y devolviéndoles un mensaje de que sus credenciales no son correctas.
Pre-condición	Tener un usuario asignado en el sistema
Post-condiciones	El sistema desplegará el menú del módulo primario en caso de ser el ingreso exitoso caso contrario seguirá en la página principal de logeo.
Caso de Uso	Generar proyecto
Actor	Administrador
Descripción	El administrador del sistema podrá crear un proyecto con su respectivo nombre, tiempo y alcance.
Flujo básico	El sistema pide al usuario el nombre del proyecto, tiempo y descripción. El usuario indica al sistema los valores de estos campos y le indica al sistema guardar la información.
Flujos Alternos	
Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejara los cambios en la base de datos
Caso de Uso	Edición de proyecto
Actor	Administrador
Descripción	El administrador del sistema podrá editar la información del proyecto
Flujo básico	El sistema pide al usuario el nombre del id, proyecto, tiempo y descripción. El usuario indica al sistema los valores de estos campos y le indica al sistema cambiar la información del proyecto.
Flujos Alternos	

Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejara los cambios en la base de datos
Caso de Uso	Eliminación de proyecto
Actor	Administrador
Descripción	El administrador del sistema podrá eliminar el proyecto
Flujo básico	El sistema pide al usuario el id del proyecto. El usuario indica al sistema los valore de este campo y le indica al sistema eliminar el proyecto y las etapas asociadas a este.
Flujos Alternos	
Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejará los cambios en la base de datos
Caso de Uso	Creación de usuario
Actor	Administrador
Descripción	El administrador del sistema podrá crear usuarios
Flujo básico	El sistema pide al usuario el nombre y contraseña del usuario. El usuario indica al sistema los valores de estos campos y le indica al sistema guardar la información.
Flujos Alternos	
Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejará los cambios en la base de datos
Caso de Uso	Eliminación de usuario
Actor	Administrador
Descripción	El administrador del sistema podrá eliminar usuarios
Flujo básico	El sistema pide al usuario el id del usuario. El usuario indica al sistema el valor de esto campo y le indica al sistema eliminar al usuario.
Flujos Alternos	

Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejará los cambios en la base de datos
Caso de Uso	Edición de usuario
Actor	Administrador
Descripción	El administrador del sistema podrá editar usuarios
Flujo básico	El sistema pide al usuario el id, nombre y contraseña del usuario. El usuario indica al sistema los valores de estos campos y le indica al sistema cambiar la información del usuario.
Flujos Alternos	
Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejará los cambios en la base de datos
Caso de Uso	Creación de etapa
Actor	Administrador
Descripción	El administrador del sistema podrá crear usuarios
Flujo básico	El sistema pide al usuario el nombre, alcance, plataformas, desarrolladores, proyecto al que pertenece, roll del desarrollador, observaciones de la etapa. El usuario indica al sistema los valores de estos campos y le indica al sistema guardar la información de la etapa.
Flujos Alternos	
Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	
Caso de Uso	Edición de etapa
Actor	Administrador
Descripción	El administrador del sistema podrá crear usuarios
Flujo básico	El sistema pide al usuario el id, nombre, alcance, plataformas, desarrolladores, proyecto al que pertenece,

	roll del desarrollador, observaciones de la etapa. El usuario indica al sistema los valores de estos campos y le indica al sistema cambiar la información de la etapa.
Flujos Alternos	
Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejará los cambios en la base de datos
Caso de Uso	Eliminación de etapa
Actor	Administrador
Descripción	El administrador del sistema podrá crear usuarios
Flujo básico	El sistema pide al usuario el id de la etapa. El usuario indica al sistema lo valor de estos campo y le indica al sistema eliminar esta información.
Flujos Alternos	
Pre-condición	El administrador tiene que haber ingresado al sistema
Post-condiciones	Se reflejará los cambios en la base de datos
Caso de Uso	Generación de observación
Actor	Usuario
Descripción	El administrador del sistema podrá crear usuarios
Flujo básico	El sistema pide la observación, id de la etapa. El usuario indica al sistema los valores de estos campos y le indica al sistema cambiar la información de la etapa.
Flujos Alternos	
Pre-condición	El usuario tiene que haber ingresado al sistema
Post-condiciones	Se reflejará los cambios en la base de datos
Caso de Uso	Subir código fuente
Actor	Administrador
Descripción	El administrador del sistema podrá crear usuarios
Flujo básico	El sistema pide un archivo de extensión rar. El usuario

	indica al sistema el archivo y le indica al sistema subir el archivo al servidor.
Flujos Alternos	
Pre-condición	El usuario tiene que haber ingresado al sistema
Post- condiciones	Se reflejará los cambios en la base de datos

3.1.2 Casos de uso aplicación móvil

En la figura 18 se puede apreciar los casos de uso de la aplicación móvil.

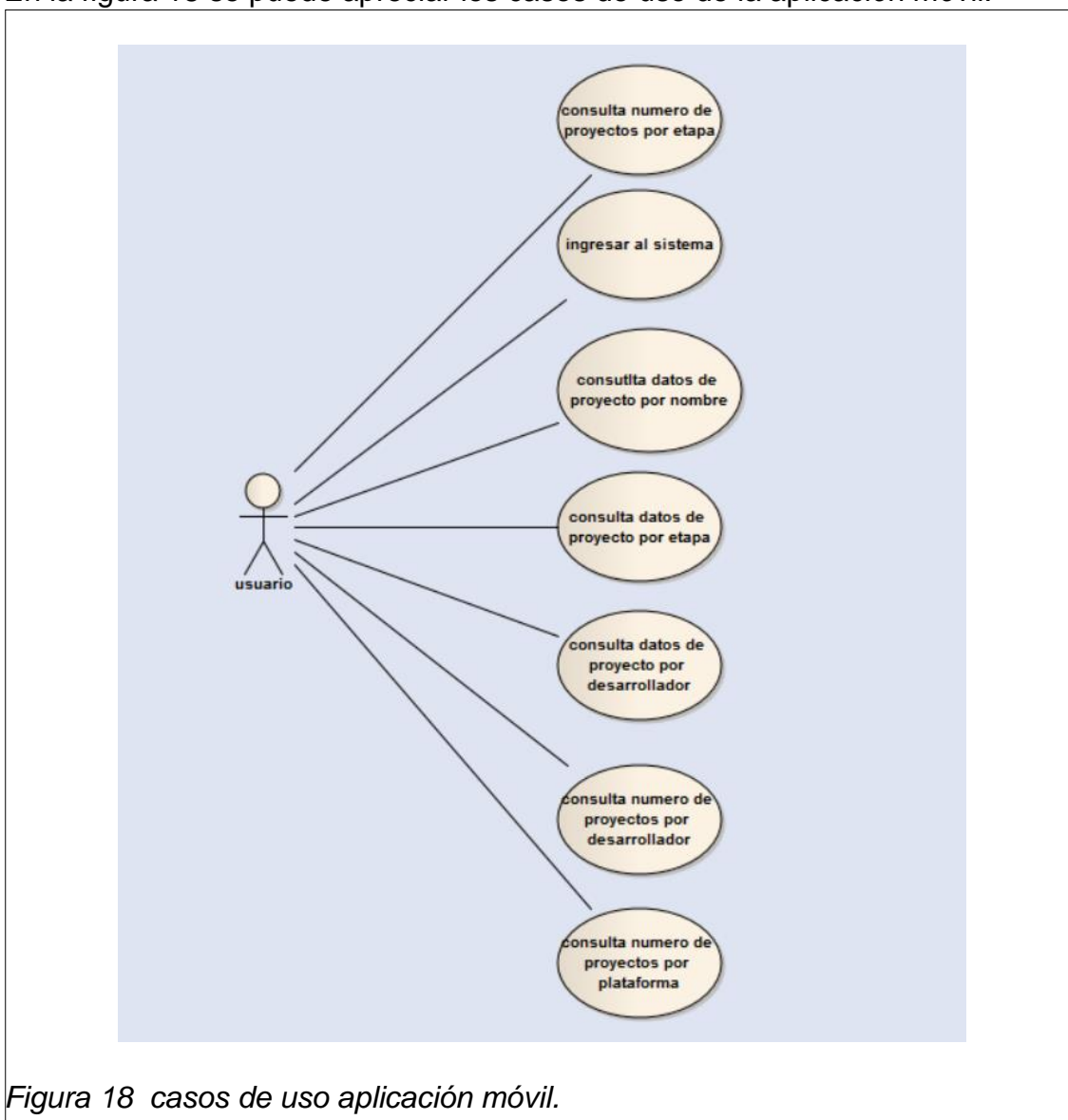


Figura 18 casos de uso aplicación móvil.

Tabla 7 Casos de uso específicos aplicación móvil.

Caso de Uso	Ingresa al sistema
Actor	Usuario
Descripción	El usuario ingresa un número de pin para poder ingresar al sistema.
Flujo básico	El sistema pide al usuario su pin. El usuario indica al sistema sus credenciales. El sistema valida que son

	correctas las credenciales y les permite ingresar a los módulos privados.
Flujos Alternos	El sistema pide al usuario sus credenciales. <u>El usuario indica al sistema sus credenciales. El sistema valida que no son correctas las credenciales y devolviéndoles un mensaje de que sus credenciales no son correctas.</u>
Pre-condición	El usuario tiene que haber ingresado al sistema
Post-condiciones	El sistema desplegará el menú del módulo primario en caso de ser el ingreso exitoso caso contrario seguirá en la página principal de logeo.
Caso de Uso	Consulta datos de proyectos por etapa
Actor	Usuario
Descripción	El usuario busca los proyectos por una etapa en específico
Flujo básico	El sistema pide al usuario el número de etapa. El usuario indica al sistema lo valor de esto campo y le indica al sistema que despliegue la información de los proyectos por la etapa ingresada
Flujos Alternos	El sistema pide al usuario sus credenciales. El usuario indica al sistema sus credenciales. El sistema valida que no son correctas las credenciales y devolviéndoles un mensaje de que sus credenciales no son correctas.
Pre-condición	El usuario tiene que haber ingresado al sistema
Post-condiciones	El sistema desplegara el menú del módulo primario en caso de ser el ingreso exitoso caso contrario seguirá en la página principal de logeo.
Caso de Uso	Consulta datos de proyectos por desarrollador
Actor	Usuario
Descripción	El usuario busca los proyectos por un desarrollador en específico
Flujo básico	El sistema pide al usuario el nombre de desarrollador. El usuario indica al sistema lo valor de esto campo y le

	indica al sistema que despliegue la información de los proyectos por el desarrollador ingresado.
Flujos Alternos	
Pre-condición	El usuario tiene que haber ingresado al sistema
Post-condiciones	El sistema desplegara el menú del módulo primario en caso de ser el ingreso exitoso caso contrario seguirá en la página principal de logeo.
Caso de Uso	Consulta datos de proyectos por plataforma
Actor	Usuario
Descripción	El usuario busca los proyectos por una plataforma en específico
Flujo básico	El sistema pide al usuario el nombre de la plataforma. El usuario indica al sistema lo valor de esto campo y le indica al sistema que despliegue la información de los proyectos por la plataforma ingresado.
Flujos Alternos	
Pre-condición	El usuario tiene que haber ingresado al sistema
Post-condiciones	El sistema desplegara el menú del módulo primario en caso de ser el ingreso exitoso caso contrario seguirá en la página principal de logeo.
Caso de Uso	Consulta número de proyectos por desarrollador
Actor	Usuario
Descripción	El usuario busca el número de proyectos por desarrollador
Flujo básico	El usuario pide al sistema que despliegue el número de proyectos por cada desarrollador.
Flujos Alternos	
Pre-condición	
Post-condiciones	El sistema desplegará el menú del módulo primario en caso de ser el ingreso exitoso, caso contrario seguirá en la página principal de logeo.

Caso de Uso	Consulta número de proyectos por plataforma
Actor	Usuario
Descripción	El usuario busca el número de proyectos por plataforma
Flujo básico	El usuario pide al sistema que despliegue el número de proyectos por cada plataforma.
Flujos Alternos	
Pre-condición	El usuario tiene que haber ingresado al sistema
Post-condiciones	El sistema desplegará el menú del módulo primario en caso de ser el ingreso exitoso caso contrario seguirá en la página principal de logeo.
Caso de Uso	Consulta número de proyectos por etapa
Actor	Usuario
Descripción	El usuario busca el número de proyectos por etapa
Flujo básico	El usuario pide al sistema que despliegue el número de proyectos por cada etapa.
Flujos Alternos	
Pre-condición	El usuario tiene que haber ingresado al sistema
Post-condiciones	El sistema desplegará el menú del módulo primario en caso de ser el ingreso exitoso caso contrario seguirá en la página principal de logeo.

3.2 Diagrama de secuencia MVC

3.2.1 Portal de administrador

En la figura 19 se puede ver el diagrama de secuencia MVC para el portar administrador.

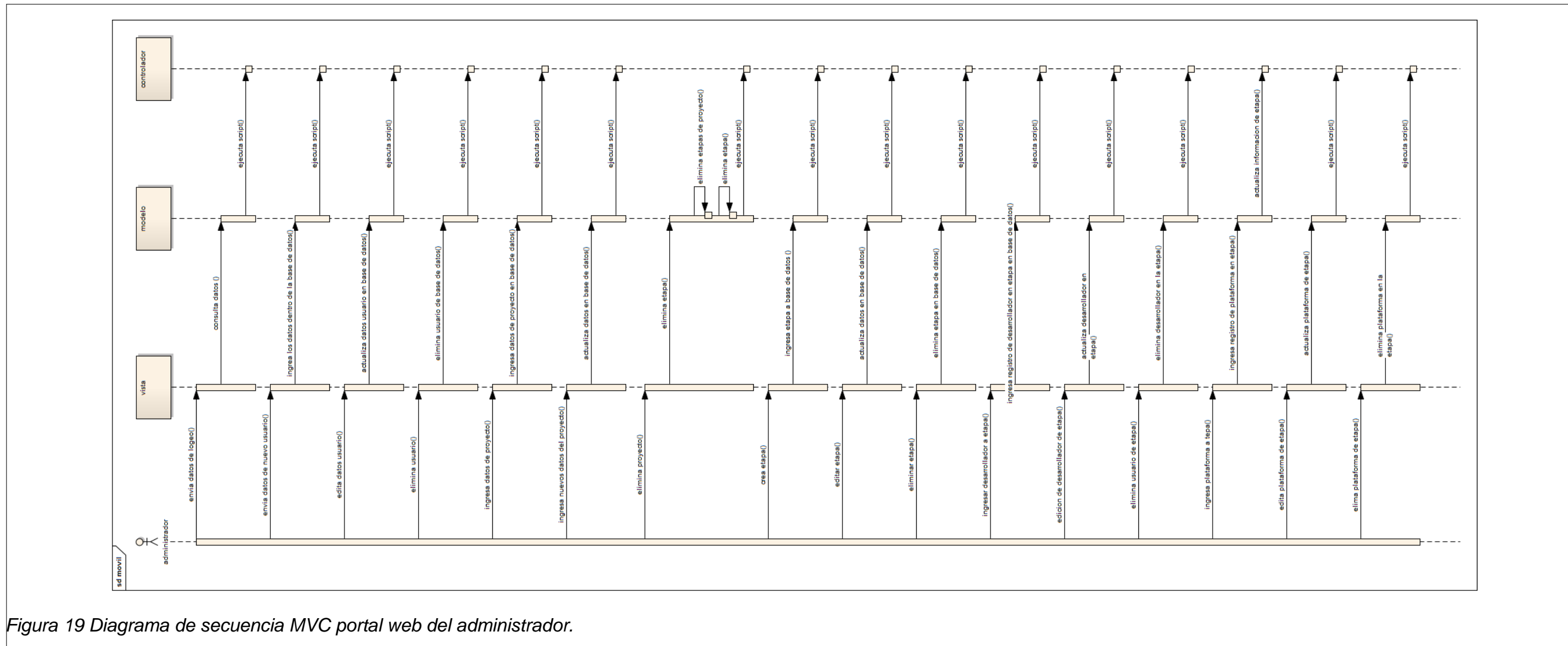


Figura 19 Diagrama de secuencia MVC portal web del administrador.

3.2.2 Portal del desarrollador

En la figura 3.2.2 se puede ver el diagrama de secuencia MVC para el portal del desarrollador.

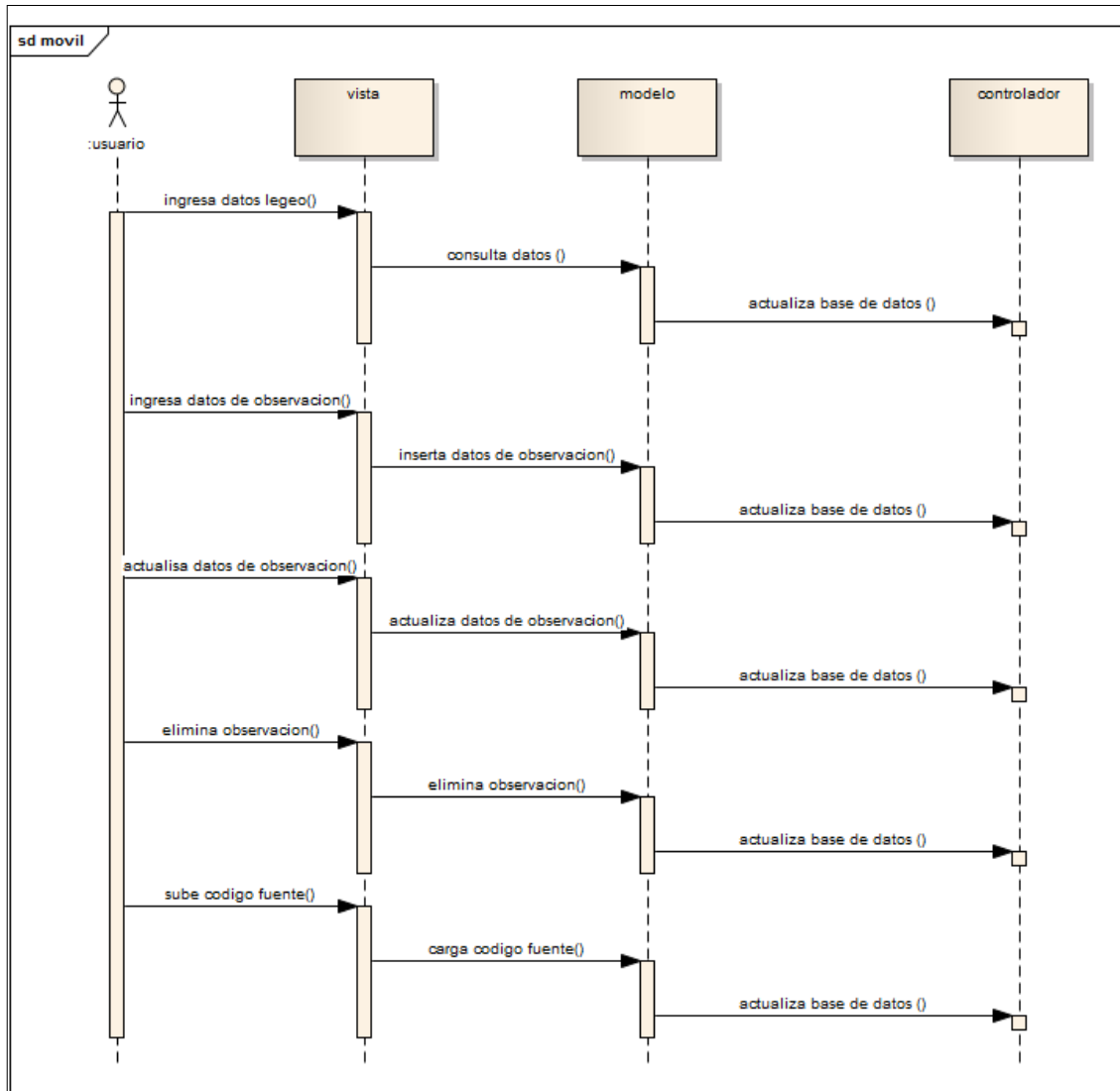


Figura 20 Diagrama MVC aplicación portal del desarrollador.

3.3 Desarrollo de aplicación web

Para el desarrollo de la aplicación se utilizó el framework CodeIgniter del lenguaje PHP y el motor para la base de datos es MySQL. El código fue alojado dentro de un servidor Apache.

3.4 Desarrollo de Web Service para aplicación móvil

Los Web Service fueron desarrollados en el lenguaje PHP devolviendo un objeto json para el consumo de la aplicación móvil, los cuales están alojados en un servidor Apache el mismo donde se encuentra el código de la aplicación web. Los datos son extraídos de la base de datos MySQL que es la misma que utiliza la aplicación web.

3.5 desarrollo de la aplicación móvil

La aplicación móvil fue desarrollada en Apache Cordova con HTML5, jQuery mobile, CSS3 con Bootstrap y Chartjs para el sistema operativa Android. Utilizando la versión 3.5.1 de Apache Cordova.

4 Pruebas

En este capítulo se realizarán pruebas para verificar la robustez y la correcta realización del software.

4.1 Pruebas de caja blanca

En estas pruebas se pondrán entradas controladas para saber si el resultado obtenido por el software es igual al esperado, con diferentes combinaciones para saber si el comportamiento interno del software es el esperado.

Tabla 8 pruebas de caja blanca

Usuario	Clave	Software esperado	Software Obtenido	Resultado		
Paul	jh162404	Ingresa al portal administrador	Ingresa al portal administrador	Aprobado		
Dario	nahia20142014	Ingresa al portal desarrollador	Ingresa al portal desarrollador	Aprobado		
Rafael	300276	No ingresa al sistema	No ingresa al sistema	Aprobado		
Nombre	Tiempo	Alcance	Software esperado	Software obtenido	Resultado	
Rocoto	4 días	Replicar aplicación Patikas con nombre rocoto y cambiar los color por azul amarillo y negro	Proyecto creado con datos ingresados	Proyecto creado con datos ingresados	ok	
Nombre	Tiempo	Alcance	Estado	iOS	Android	Web
Primera	2 semanas	Estructura base de la pagina web	No iniciado	No	No	Si
Rafael	Dario	Proyecto	Observaciones	sf esperado	sf obtenido	resultado
Web	No participa	Gabriel Rivera	-	Etapa creada con los datos asignados	Etapa creada con los datos asignados	Ok
Nombre	Descripción	Tipo	Proyecto	Software Esperado	Software Obtenido	Resultado
Carrusel	El carrusel debe ser controlado por el teclado	No funcional	Gabriel Rivera	Creación de requerimiento con los datos ingresado	Creación de requerimiento con los datos ingresado	<u>Ok</u>
Software Esperado	Software Obtenido	Resultado				
Rafael 3	Rafael 3	Aprobado				
Darío 2	Darío 2					

Software esperado		Software Obtenido	Resultado
Primera no iniciada 1		Primera no iniciada 1	Aprobada
Segunda no iniciada 0		Segunda no iniciada 0	
Tercera no iniciada 0		Tercera no iniciada 0	
Primera iniciada 0		Primera iniciada 0	
Segunda iniciada 1		Segunda iniciada 1	
Tercera iniciada 0		Tercera iniciada 0	
Primera Finalizada 1		Primera Finalizada 1	
Segunda Finalizada 0		Segunda Finalizada 0	
Tercera Finalizada		Tercera Finalizada	
Software esperado		Software obtenido	Resultado
Android 0		Android 0	Falla
Web 2		Web 0	
iOS 2		iOS 2	
Software esperado	Software obtenido	Resultado	
Android 0	Android 0	Aprobado	
Web 0	Web 0		
iOS 2	iOS 2		

Como se puede apreciar en el módulo de reporte por plataforma existía una falla que fue resuelta gracias a la pruebas de caja blanca del sistema.

4.2 Pruebas de caja negra

En estas pruebas se pondrán entradas controladas para saber si el resultado obtenido por el software es igual al esperado para ver si el software cumple con los requerimientos establecidos en el capítulo dos.

Tabla 9 pruebas de caja negra

Proyecto	Eliminación esperada	Eliminación del sistema	Resultado
Gabriel Rivera	Proyecto Gabriel Rivera	Proyecto Gabriel Rivera	Aprobada
Proyecto	Edición Esperada	Edición del sistema	Resultado
Flexnet	Cambio en la base de datos según los campos ingresados	Cambio en la base de datos según los campos ingresados	Aprobado
Requerimiento	Eliminación esperada	Eliminación del sistema	Resultado
Paginación	Eliminación del requerimiento Paginación	Eliminación del requerimiento Paginación	Aprobado
Requerimiento	Eliminación esperada	Eliminación del sistema	Resultado
Paginación	Cambio en la base de datos según los campos ingresados	Cambio en la base de datos según los campos ingresados	Aprobado
Requerimiento	Eliminación esperada	Eliminación del sistema	Resultado
Primera	Cambio en la base de datos según los campos ingresados	Cambio en la base de datos según los campos ingresados	Aprobado
Requerimiento	Eliminación esperada	Eliminación del sistema	Resultado
Primera	Eliminación de la etapa Primera	Eliminación de la etapa Primera	Aprobado

Como se puede apreciar con las pruebas de caja blanca y caja negra el sistema cumple con lo establecido en los requerimientos del sistema.

5 Conclusiones y Recomendaciones

Como se puede apreciar, el trabajo de titulación fue realizado mediante las normas de la metodología de desarrollo RUP, con su ciclo de vida.

Por tanto, se pueden puntualizar las siguientes conclusiones y recomendaciones.

5.1 Conclusiones

La empresa *Nahia Solutions S.A* logro gracias al software establecer diferentes etapas de gestión de un proyecto de desarrollo de software lo cual le permitió administrar sus riesgos.

Mediante el diseño de la solución para la empresa *Nahia Solution S.A* reestructuro sus políticas de negocio.

Se evidencio con la implementación de la herramienta de software ligado a la metodología tradicional RUP que se logró una mejor gestión de los proyectos y guardar de una manera ordenada sus avances.

Con un repositorio centralizado de los códigos fuentes de los proyectos, se da un mejor seguimiento de los bugs o inconvenientes de los mismos.

Al definir claramente los alcances de los proyectos y las responsabilidades de los que interactúan con ellos, se puede optimizar la carga para el personal del área del desarrollo para mejorar el uso del tiempo de desarrollo de los proyectos mediante el seguimiento del Director General y del Director de Desarrollo a través de la aplicación móvil.

Al guardar las observaciones de las diferentes etapas de los proyectos, se pueden apreciar con mayor facilidad los errores que se están cometiendo para poder corregirlos y evitarlos para ese y futuros proyectos.

Se cumple los objetivos con la implementación exitosa del sistema. Se logró resolver el problema planteado en el trabajo de titulación logrando corregir la falencia en el control del avance de los proyectos dentro de la empresa *Nahia Solutions S.A.*

5.2 Recomendaciones

Al observar el tipo de proyectos que maneja la empresa *Nahia Solutions S.A.* se puede recomendar lo siguiente:

Debido a las características de los proyectos informáticos se consideraría que el software además de estar ligado a una metodología de desarrollo tradicional RUP, se debería considerar una metodología ágil.

Si la empresa mantiene la idea de dividir sus proyectos por etapas, entonces debería poner una fecha límite para cada una de éstas, además de hitos semanales al personal de la empresa, jerarquizándolos bajo el criterio de prioridad.

A base del recomendación dos, consideraría un sistema de alerta temprana para el cumplimiento de los tiempos establecidos.

REFENECIAS

- AlertaenLinea. (s.f.) AlertaenLinea. Recuperado el 2015 de Septiembre de 2011 de <https://www.alertaadenlinea.gov/articulos/s0018-aplicaciones-m%C3%B3viles-qu%C3%A9-son-y-c%C3%B3mo-funcionan>
- Boliviana (s.f.) Ingenieria de software. Recuperado el 2 de Febrero de 2016 de Obtenido de http://ingenieriadesoftware.mex.tl/52753_XP---Extreme-Programing.html.
- e-quallity. (s.f.). e-quallity.net Recuperado el 24 de Agosto de 2015 de de <http://e-quallity.net/es/servicio/pruebas-de-desempeño>
- González, J. C. (s.f.) Lider de Proyecto. Recuperado el 13 de Noviembre de 2014 Recuperado de http://www.liderdeproyecto.com/articulos/fase_conceptual_metodologia_administracion_proyectos_software.html
- Google (s.f.) https://developer.chrome.com/apps/app_frameworks .Recuperado el 01 de Agosto de 2015 de <https://developer.chrome.com>. https://developer.chrome.com/apps/app_frameworks
- group, p. m. (s.f.) <http://www.pcmag.com>. Recuperado el 26 de Julio de 2015 Recuperado de <http://www.pcmag.com/encyclopedia/term/54272/web-application>
- Haughey, D. (s.f.) Lider De Proyecto. Recuperado el 17 de Noviembre de 2015 de http://www.liderdeproyecto.com/manual/breve_historia_sobre_la_administracion_de_proyectos.html
- <http://aprenderaprogramar.com>. (s.f) aprende a programa Recuperado el 2 de febrero de 2016 de http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=688:ique-es-y-para-que-sirve-uml-versiones-de-uml-lenguaje-unificado-de-modelado-tipos-de-diagramas-uml&catid=46:lenguajes-y-entornos&Itemid=163

IBM.(s.f.) IBM Knowledge Cente. Recuperado el 3 de febrero de 2016
http://www-01.ibm.com/support/knowledgecenter/SSCP65_4.0.3/com.ibm.rational.rmm.help.doc/topics/r_uc_spec_outline.html?lang=es

libre, g. a. (s.f.) <http://www.gcfaprendelibre.org/>. Recuperado el 26 de Julio de 2015 de
http://www.gcfaprendelibre.org/tecnologia/curso/informatica_basica/aplicaciones_web_y_todo_acerca_de_la_nube/1.do

Lourdes, M. d.(s.f.) universidad tecnológica del valle del Mezquital. Recuperado el 17 de febrero de 2015 de
<http://www.utvm.edu.mx/OrganoInformativo/orgJul07/RUP.htm>

UML. (s.f.) UML. Recuperado el 06 de Octubre de 2015 de
<http://www.uml.org/what-is-uml.htm>

Point, T. <http://www.tutorialspoint.com>. Recuperado el 01 de Agosto de 2015 de
http://www.tutorialspoint.com/struts_2/basic_mvc_architecture.htm

Project Management Institut. <http://www.pmi.org>. Recuperado el 15 de Julio de 2015 de
<http://www.pmi.org/About-Us/About-Us-What-is-Project-Management.aspx>

LIBROSWEB (s.f) .acerca del control de versiones. Recuperado el 24 de agosto de 2015 el
http://librosweb.es/libro/pro_git/capitulo_1/acerca_del_control_de_versiones.html

SOMMERVILLE, I. (2005). *Ingñiería del software*. España: Pearson.

Sommerville, I. (2011). *INGENIERÍA DE SOFTWARE*. Naucalpan de Juárez: Pearson.

studio, o. (s.f) odm studio.Recuperado el 17 de Febero de 2016 de
<http://www.odmstudio.com.mx/ws/dossier/articulos-de-interes/por-que-la-administracion-de-proyectos.htm>

Talent, L. (s.f.) los Tres tipos de aplicaciones móviles. Recuperado el 08 de Agosto de 2015 el <http://www.lancetalent.com/blog/tipos-de-aplicaciones-moviles-ventajas-inconvenientes/>

terms, t. (s.f) Web Application. Recuperado el 17 de Agosto de 2015 de http://techterms.com/definition/web_application

wikispaces (s.f). Metodología RUP: Recuperado el 4 de Octubre de 2015 el <http://procesosdesoftware.wikispaces.com/METODOLOGIA+RUP>

wikispaces. (s.f). métodos ágiles vs métodos tradicionales. Recuperado el 2 de Febrero de 2016 el <https://metodologiasagiles.wikispaces.com/metodos+agiles+vs+metodos+tradicionales>

ANEXOS

Anexo 1 Pruebas de caja blanca

The image displays four sequential screenshots of a web application interface, illustrating user login and project management features.

Screenshot 1: Login
The page title is "Login". It features two input fields: the first contains the username "Paul" and the second contains a masked password "*****". A blue "Ingresar" button is positioned below the password field.

Screenshot 2: Project List
The page title is "Proyectos". Below the title, it reads "Lista de los proyectos de la empresa Nahia Solutions". A blue "Crear Proyecto" button is on the left. The main content area shows a table with two project entries:
- Entry 1: "Nombre: Flexnet ecuador" with "Eliminar" and "Modificar" buttons.
- Entry 2: "Nombre: Rocoto" with "Eliminar" and "Modificar" buttons.

Screenshot 3: Login
The page title is "Login". It features two input fields: the first contains the username "Dario" and the second contains a masked password "*****". A blue "Ingresar" button is positioned below the password field.

Screenshot 4: Login
The page title is "Login". It features two input fields: the first contains the username "Rafael" and the second contains a masked password "*****". A blue "Ingresar" button is positioned below the password field.

Screenshot 5: Login
The page title is "Login". It features two input fields: the first is labeled "Usuario" and the second is labeled "Password". A blue "Ingresar" button is positioned below the password field.

Anexo 2 Manual de usuario

Manual de Usuario

1 Introducción.

El presente documento está dirigido a entregar las normas para el uso correcto del sistema de monitoreo de avances en el desarrollo de proyectos de software.

2 Usuarios.

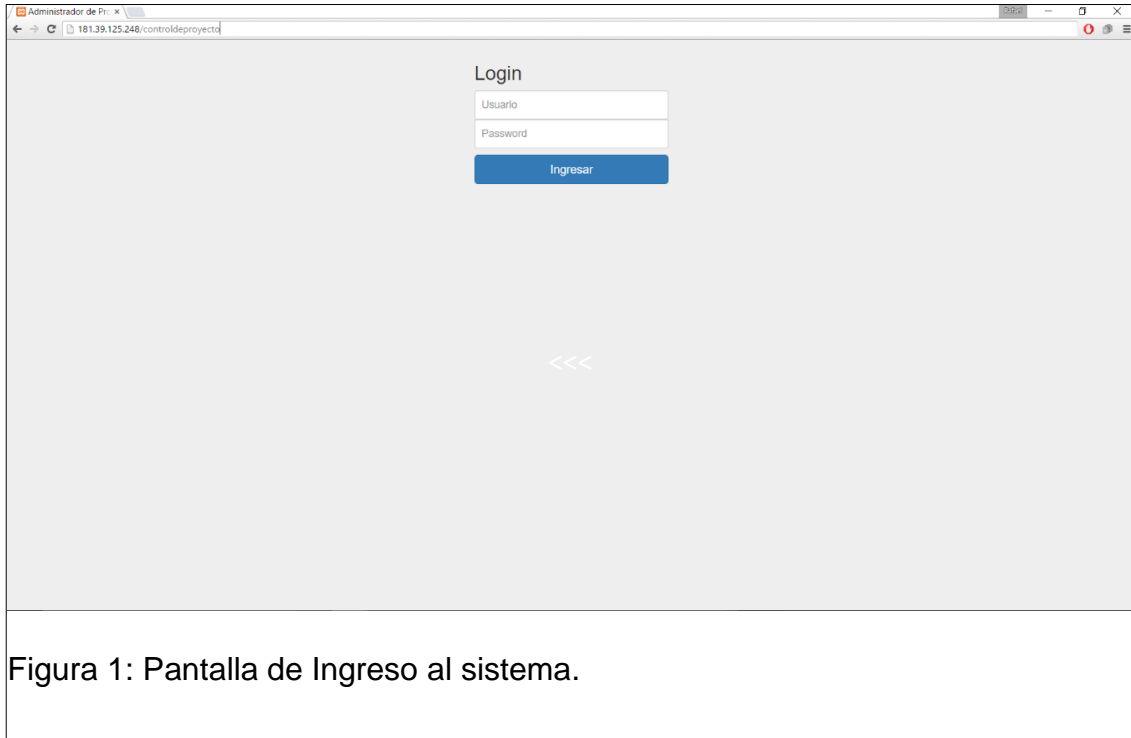
Existen dos diferentes tipos de usuario:

- **Administrador:** este tipo usuario administra el sistema permitiéndole gestionar los proyectos, etapas, requerimiento, ver el repositorio de código fuente, ver los diferentes reportes, cambiar las contraseñas de los usuarios.
- **Desarrollador:** este tipo usuario puede ver únicamente las etapas de los proyectos al que ha sido asignado en las cuales podrá ingresar las observaciones que crea pertinente. Podrá subir el código fuente de las diferentes plataformas al repositorio de cada proyecto.

3 Ingreso al sistema.

El sistema de control para monitorear avances en el desarrollo de proyectos de software es una aplicación web por lo que para acceder al sistema se necesita un navegador web. La dirección URL es <http://181.39.125.248/controldeproyectos>. Una vez visualizada la pantalla se debe ingresar los datos de acceso que son

el nombre de usuario y la contraseña.



Si los datos de ingreso son incorrectos no avanzara de la pantalla de inicio del sistema.

4 Funcionalidades del usuario administrador.

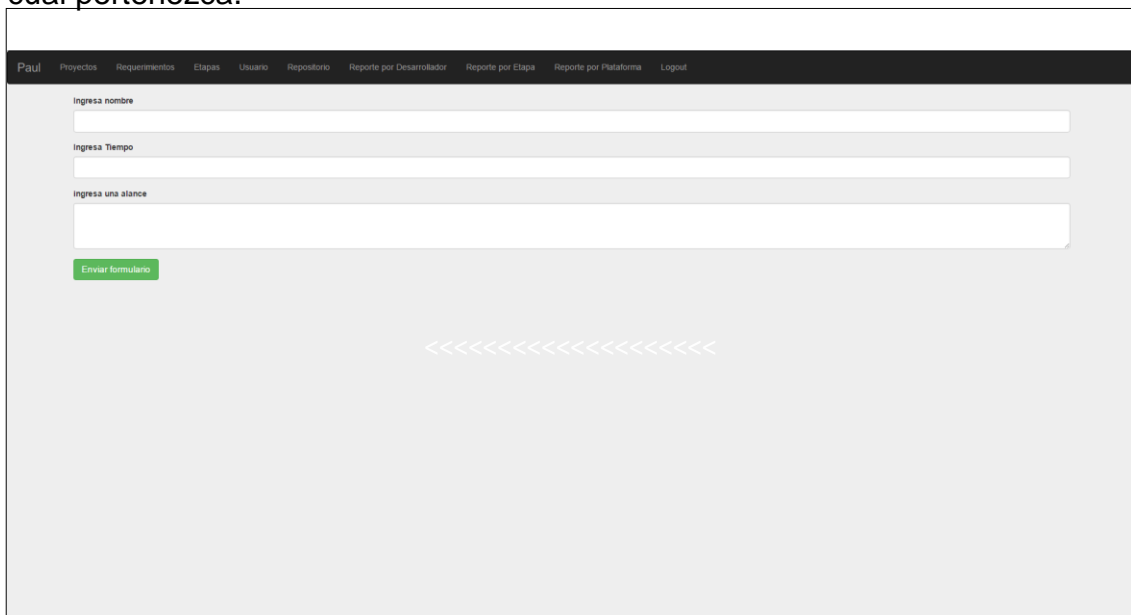
Al ingresar al sistema como administrador dispondrá de las siguientes opciones:

- Creación de proyectos.
- Edición de proyectos.
- Eliminación de proyectos.
- Creación de requerimiento.
- Edición de requerimiento.
- Eliminación de requerimiento.
- Creación de etapa.
- Edición de etapa.
- Eliminación de etapa.
- Cambio de contraseñas de usuarios.

- Descargar código fuente de los proyectos por su número de versión y plataforma.
- Visualización de estadísticas por: Desarrollador, Etapa o Plataforma.

4.1 Creación de proyectos.

En la pantalla de creación de proyecto se pedirá que se llene tres campos los cuales son nombre, tiempo y alcance el nombre de proyecto no se puede repetir ya que cada proyecto se le asigna una carpeta con su nombre para poder subir el código fuente de cada versión dependiendo de la plataforma a la cual pertenezca.



The screenshot shows a web application interface for creating a project. At the top, there is a navigation bar with the name 'Paul' and several menu items: 'Proyectos', 'Requerimientos', 'Etapas', 'Usuario', 'Repositorio', 'Reporte por Desarrollador', 'Reporte por Etapa', 'Reporte por Plataforma', and 'Logout'. Below the navigation bar, the main content area contains a form with three input fields: 'Ingresar nombre', 'Ingresar Tiempo', and 'Ingresar una alcance'. Each field is followed by a horizontal line indicating the input area. Below the fields is a green button labeled 'Enviar formulario'. At the bottom of the form area, there is a decorative horizontal line consisting of a series of white chevrons pointing to the left.

Figura 2: pantalla de creación de proyectos.

4.2 Edición de proyecto.

Para modificar un proyecto se tiene que seleccionar primero el que se desea en la lista de proyectos. Una vez seleccionado se desplegara un formulario con los siguientes campos: nombre, tiempo y alcance que estarán llenados con los datos guardados se puede modificar un campo o todos.

Paul Proyectos Requerimientos Etapas Usuario Repositorio Reporte por Desarrollador Reporte por Etapa Reporte por Plataforma Logout

Ingresa nombre
Flexnet ecuador

Ingresa Tiempo
1 mes

Alcance
Pagina web de la empresa y su gestor

Enviar formulario

Figura 3: pantalla de edición de proyectos.

4.3 Eliminación de proyectos.

En la página principal de proyecto se listara los proyectos donde cada uno tendrá un botón donde eliminarlo de color rojo. Una vez pulsado se eliminara el proyecto al igual que los requerimientos que este tenga y sus etapas.

Paul Proyectos Requerimientos Etapas Usuario Repositorio Reporte por Desarrollador Reporte por Etapa Reporte por Plataforma Logout

Proyectos
Lista de los proyectos de la empresa Nahia Solutions

Crear Proyecto

Nombre: Flexnet ecuador	Eliminar	Modificar
Nombre: Rocoto	Eliminar	Modificar
Nombre: Gabriel Rivera	Eliminar	Modificar

Figura 4: Pagina para eliminación de proyectos.

4.4 Información de proyecto.

Para poder observar la información del proyecto se tendrá que dar click sobre el nombre del proyecto en la pantalla donde se lista los proyectos.

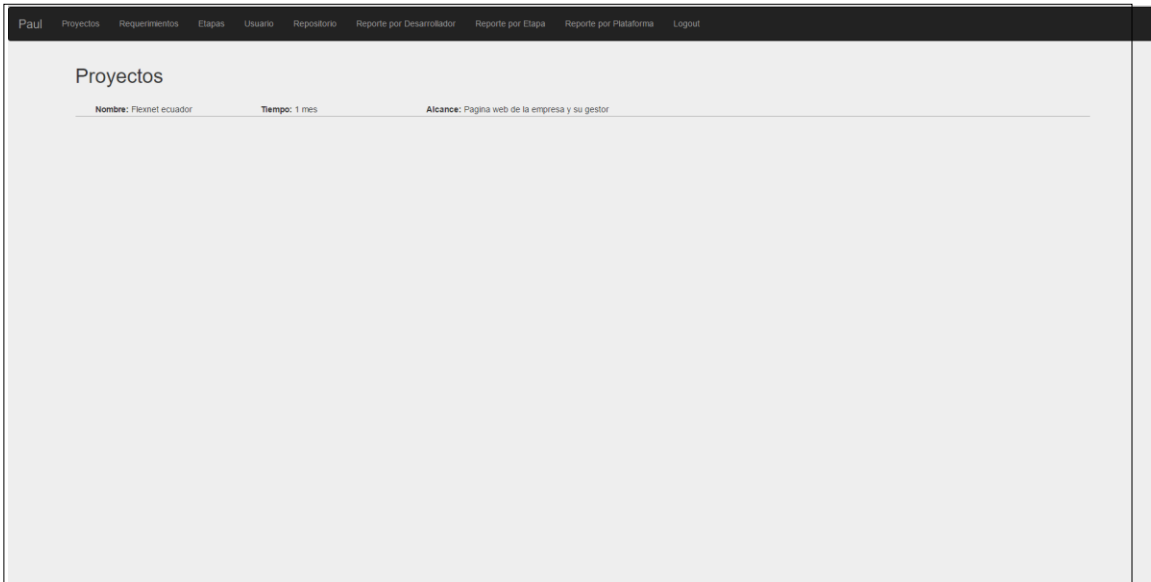


Figura 5: Pagina de información de proyecto

4.5 Creación de requerimiento.

En la pantalla de creación de requerimiento se pedirá que se llenen los siguientes campos los cuales son nombre, descripción, tipo y al proyecto que pertenece.

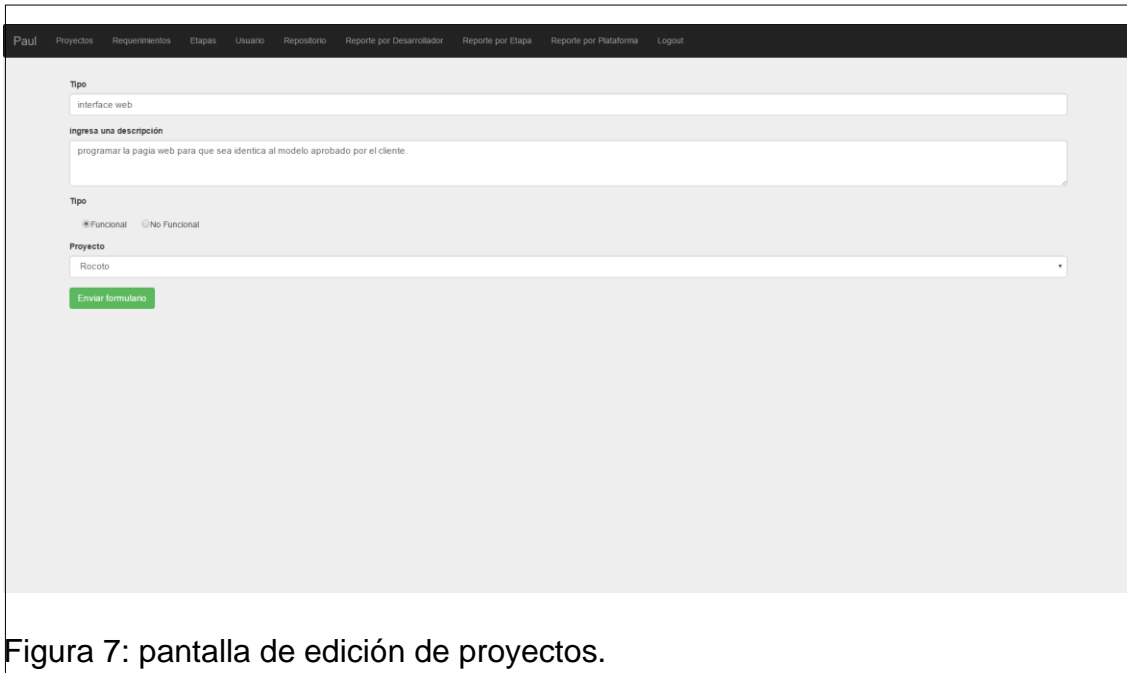
The screenshot shows a web application interface with a dark header bar containing the name 'Paul' and several navigation links: 'Proyectos', 'Requerimientos', 'Etapas', 'Usuario', 'Repositorio', 'Reporte por Desarrollador', 'Reporte por Etapa', 'Reporte por Plataforma', and 'Logout'. Below the header, the main content area is a form for creating a project. It contains the following fields and controls: a text input field labeled 'Nombre'; a text input field labeled 'Ingresar una descripción'; a section for 'Tipo' with two radio buttons: 'Funcional' (selected) and 'No Funcional'; a dropdown menu labeled 'Proyecto' with 'Flexnet ecuador' selected; and a green button labeled 'Enviar formulario'.

Figura 6: pantalla de creación de proyectos.

4.6 Edición de requerimiento.

Para modificar un requerimiento se tiene que seleccionar primero el que se desea en la lista de requerimientos. Una vez seleccionado se desplegara un

formulario con los siguientes campos: nombre, descripción, tipo y al proyecto que pertenece que estarán llenados con los datos guardados se puede modificar un campo o todos.

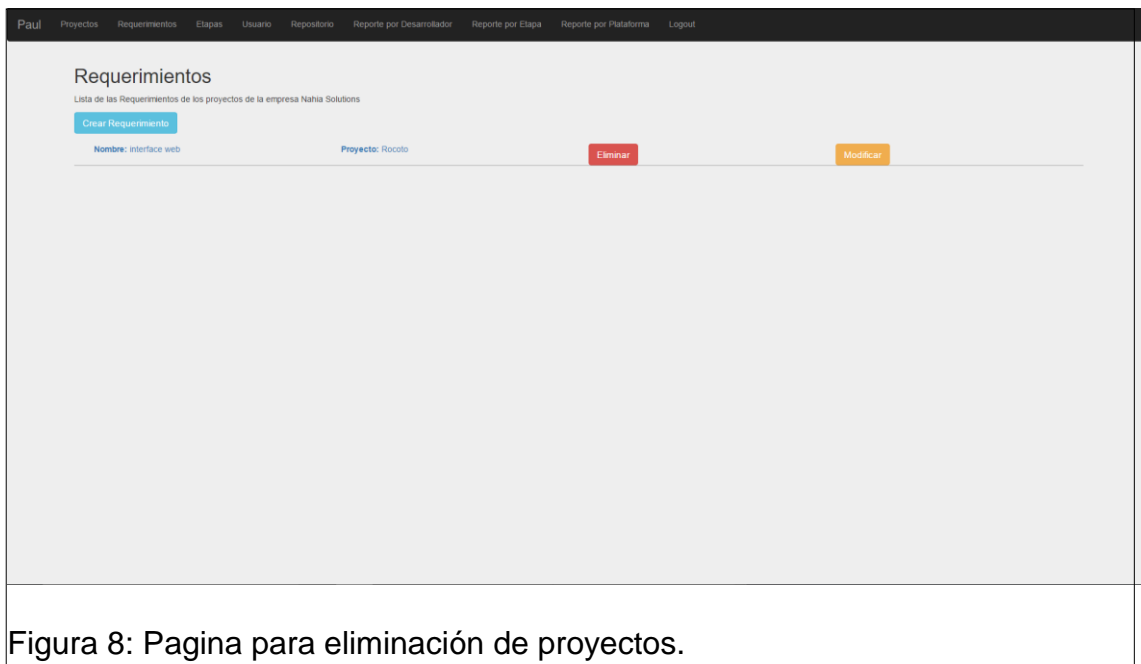


The screenshot shows a web application interface for editing a project. At the top, there is a navigation bar with the name 'Paul' and several menu items: 'Proyectos', 'Requerimientos', 'Etapas', 'Usuario', 'Repositorio', 'Reporte por Desarrollador', 'Reporte por Etapa', 'Reporte por Plataforma', and 'Logout'. The main content area is a form with the following fields: a 'Tipo' dropdown menu with 'interface web' selected; a text area for 'Ingresar una descripción' containing the text 'programar la pagina web para que sea idetica al modelo aprobado por el cliente.'; radio buttons for 'Funcional' (selected) and 'No Funcional'; and a 'Proyecto' dropdown menu with 'Rocoto' selected. A green button labeled 'Enviar formulario' is located at the bottom of the form.

Figura 7: pantalla de edición de proyectos.

4.7 Eliminación de proyectos.

En la página principal de requerimiento se listara los proyectos donde cada uno tendrá un botón donde eliminarlo de color rojo. Una vez pulsado se eliminara el requerimiento.



The screenshot shows the 'Requerimientos' page. The navigation bar is the same as in Figure 7. The main content area has the title 'Requerimientos' and a subtitle 'Lista de las Requerimientos de los proyectos de la empresa Nahia Solutions'. There is a blue button labeled 'Crear Requerimiento'. Below this, there is a table with one row of data. The table has columns for 'Nombre', 'Proyecto', and actions. The row contains: 'Nombre: interface web', 'Proyecto: Rocoto', a red 'Eliminar' button, and an orange 'Modificar' button.

Figura 8: Pagina para eliminación de proyectos.

4.8 Información de requerimiento.

Para poder observar la información del requerimiento se tendrá que dar click sobre el nombre del requerimiento en la pantalla donde se lista los requerimientos.

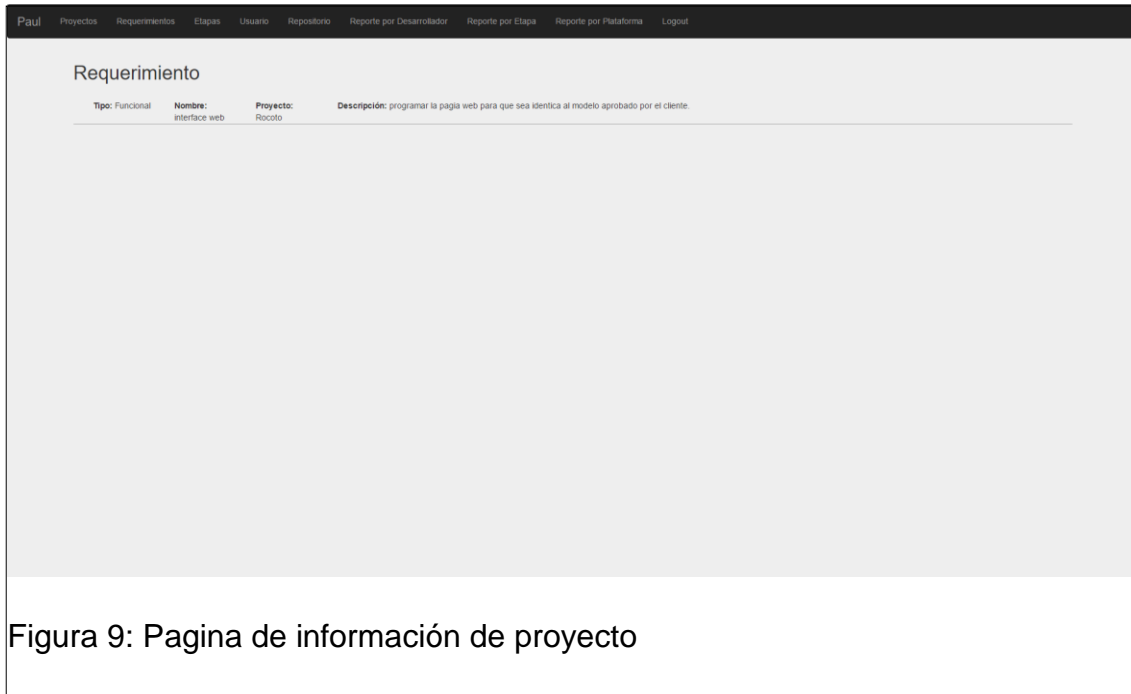


Figura 9: Pagina de información de proyecto

4.9 Creación de etapa.

En la pantalla de creación de etapa se pedirá que se llenen los siguientes campos los cuales son nombre, tiempo, alcance, estado, las plataformas que incluye, los roles de los desarrolladores, observaciones y al proyecto que

pertenece.

The screenshot shows a web application interface for creating a stage. The navigation bar at the top includes 'Paul', 'Proyectos', 'Requerimientos', 'Etapas', 'Usuario', 'Repositorio', 'Reporte por Desarrollador', 'Reporte por Etapa', 'Reporte por Plataforma', and 'Logout'. The form contains the following fields:

- Nombre:** A dropdown menu with 'Primera' selected.
- Tiempo:** A text input field.
- ingresa una alcance:** A text input field.
- Estado:** Radio buttons for 'Curso', 'No Iniciado', and 'Acabada'.
- iOS:** Radio buttons for 'No' and 'Si'.
- Android:** Radio buttons for 'No' and 'Si'.
- Web:** Radio buttons for 'No' and 'Si'.
- Rafael:** A dropdown menu with 'No participa' selected.
- Dario:** A dropdown menu with 'No participa' selected.
- Proyecto:** A dropdown menu with 'Flexnet ecuador' selected.
- Observacion:** A text input field.

Figura 11: pantalla de creación de etapa.

4.10 Edición de etapa.

Para modificar una etapa se tiene que seleccionar primero el que se desea en la lista de etapas. Una vez seleccionada se desplegara un formulario con los siguientes campos: nombre, tiempo, alcance, estado, las plataformas que incluye, los roles de los desarrolladores, observaciones y al proyecto que pertenece, que estarán llenados con los datos guardados se puede modificar un campo o todos.

Paul | Proyectos | Requerimientos | Etapas | Usuario | Repositorio | Reporte por Desarrollador | Reporte por Etapa | Reporte por Plataforma | Logout

Nombre
Primera

Tiempo
2 semanas

Alcance
crear lista de noticia, comunicados y tips.

Estado
 Curso
 No Iniciado
 Acabada

IOS
 No
 Si

Android
 No
 Si

Web
 No
 Si

Rafael
Web/Movil

Dario
Android

Proyecto
Rocoto

Organización

Figura 11: pantalla de edición de etapa.

4.11 Eliminación de etapas.

En la página principal de requerimiento se listara las etapas donde cada uno tendrá un botón donde eliminarlo de color rojo. Una vez pulsado se eliminara la etapa.

Paul | Proyectos | Requerimientos | Etapas | Usuario | Repositorio | Reporte por Desarrollador | Reporte por Etapa | Reporte por Plataforma | Logout

Etapa
Lista de las etapas de proyectos de la empresa Nahia Solutions

Crear Proyecto

Nombre	Proyecto	Eliminar	Modificar	Ver
Nombre: 1	Proyecto: Rocoto	Eliminar	Modificar	Ver

Figura 12: Página para eliminación de proyectos.

4.12 Información de etapa.

Para poder observar la información de la etapa se tendrá que dar clic sobre el nombre de la etapa en la pantalla donde se lista las etapas.

The screenshot shows a web application interface with a dark navigation bar at the top containing the name 'Paul' and several menu items: 'Proyectos', 'Requerimientos', 'Etapas', 'Usuario', 'Repositorio', 'Reporte por Desarrollador', 'Reporte por Etapa', 'Reporte por Plataforma', and 'Logout'. The main content area is light gray and displays the following project details:

- Proyecto: Rocoto
- Nombre: 1
- Tiempo: 2 semanas
- Estado: Curso
- iOS: Si
- Android: Si
- Web: Si
- Rafael: Web/Movil
- Dario: Android
- Observaciones: utilizar base de datos de prueba
- Alcance: crear lista de noticia, comunicados y tips.

Figura 13: Pagina de información de proyecto

4.13 Cambio de contraseña usuario.

Para poder modificar la contraseña de los usuarios se tiene que dar clic en el botón color amarillo en la lista de usuarios una vez dentro, se pedirá que ingrese la nueva contraseña del usuario.

The screenshot shows a web application interface with a dark navigation bar at the top containing the name 'Paul' and several menu items: 'Proyectos', 'Requerimientos', 'Etapas', 'Usuario', 'Repositorio', 'Reporte por Desarrollador', 'Reporte por Etapa', 'Reporte por Plataforma', and 'Logout'. The main content area is light gray and displays a form for changing a user's password:

- Nombre de Usuario: Dario
- Ingresar Clave
- A text input field for the password.
- A green button labeled 'Enviar formulario'.

Figura 14: pantalla de modificación de contraseña de usuario.

4.14 Repositorio.

En la pantalla de repositorio se listaran los nombres de los proyectos creados, donde se podrá cada uno tendrá carpetas para las plataformas Android, iOS y

Web. Cada una de estas carpetas tendrá las diferentes versiones de código fuente del proyecto al que corresponde el cual podrá ser descargado.

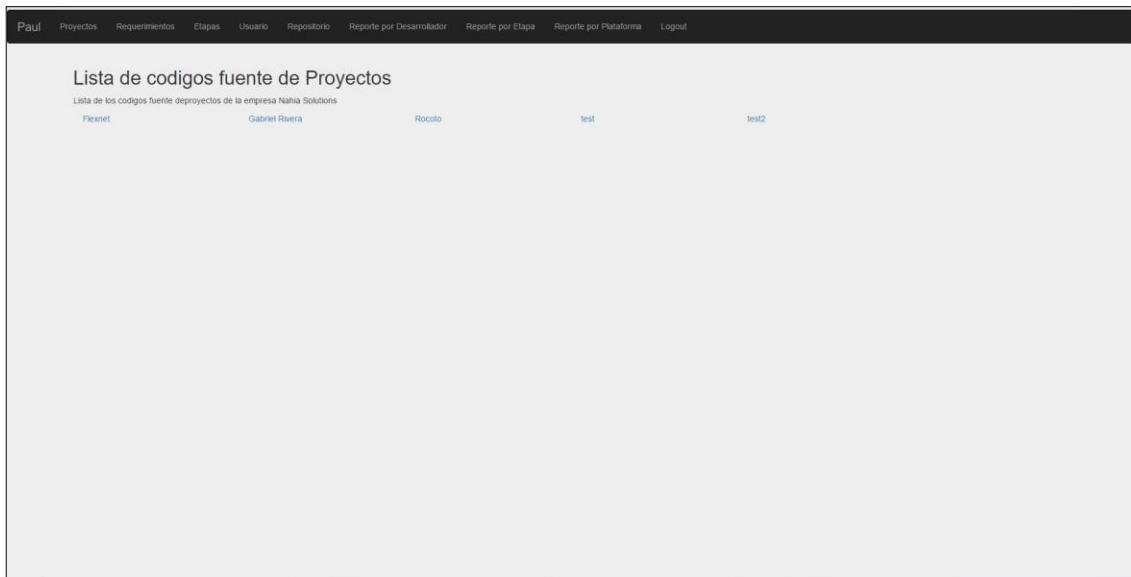


Figura 15: pantalla de repositorio.

4.15 Estadísticas.

Existen tres pantallas para las estadísticas las cuales son para saber el número de proyectos por desarrollador, número de proyectos por etapa y su estado, y número de proyectos por plataforma. Para acceder a dichas pantallas se tiene dar click en Reporte por Desarrollador, Reporte por Etapa o Reporte por Plataforma en la barra de navegación del sistema.

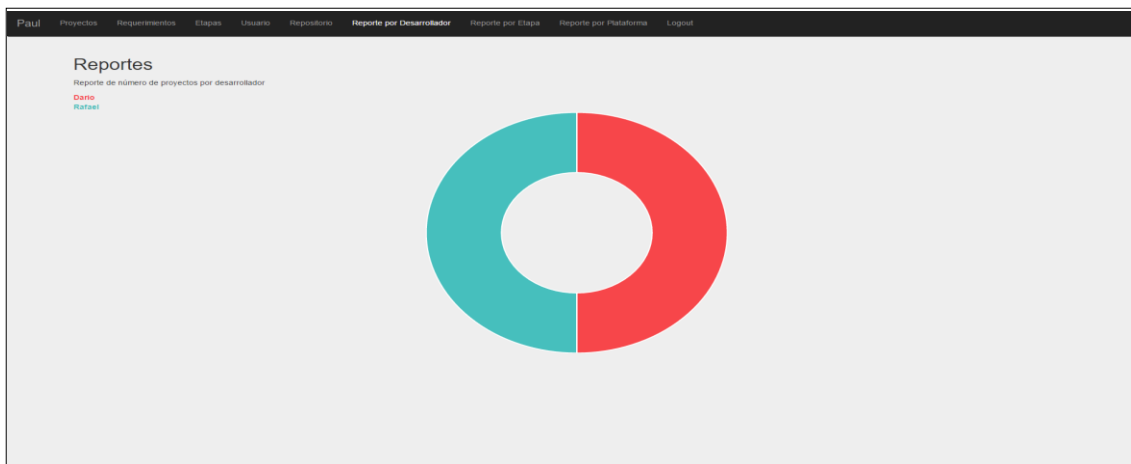


Figura 16: Pantalla de reporte por usuario

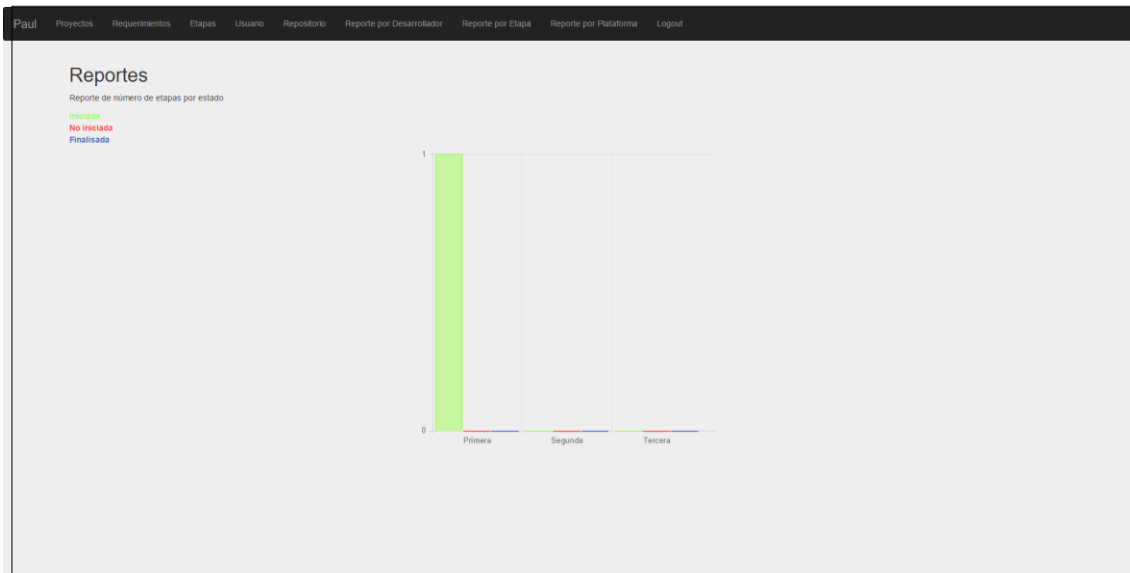


Figura 17: pantalla de reporte por etapa

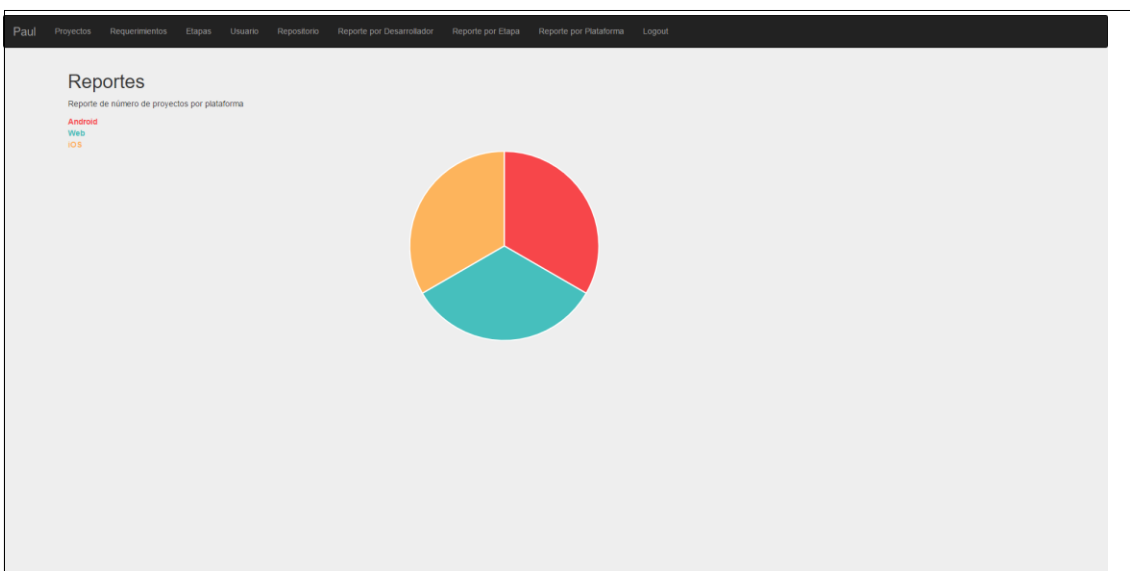


Figura 18: pantalla de reporte por plataforma.

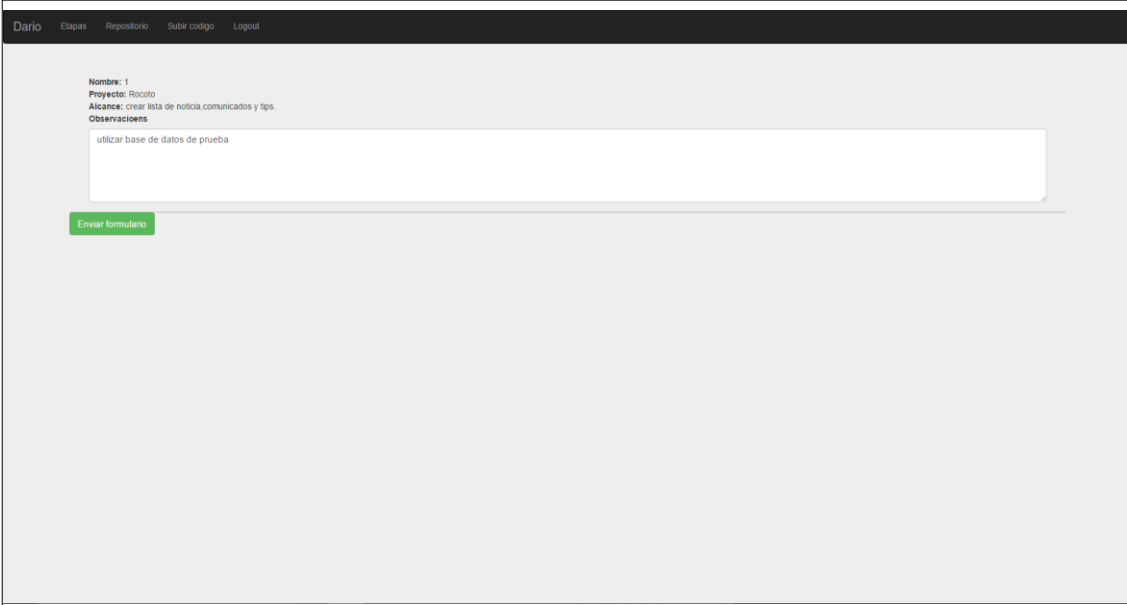
5 Funcionalidades del usuario desarrollador.

Al ingresar al sistema como desarrollador dispondrá de las siguientes opciones:

- Ingresar observaciones a las etapas asignadas.
- Subir código fuente de los proyectos.
- Descargar código fuente de los proyectos por su número de versión y plataforma.

5.1 Ingreso de observaciones.

En la pantalla principal se listarán las etapas asignadas al desarrollador donde se encuentra un botón color amarillo con el texto modificar para cada uno. Una vez seleccionado se desplegará la información de la etapa, donde solo se podrá modificar el campo de las observaciones.



The screenshot shows a web application interface for entering observations. At the top, there is a navigation bar with the following items: "Dario", "Etapas", "Repositorio", "Subir código", and "Logout". Below the navigation bar, the form displays the following information:

- Nombre: 1
- Proyecto: Rocoto
- Alcance: crear lista de noticia, comunicados y tps.
- Observaciones

Below this information is a large text input field containing the text "utilizar base de datos de prueba". At the bottom of the form is a green button labeled "Enviar formulario".

Figura 19: pantalla de ingreso de observaciones para el desarrollador.

5.2 Subir código fuente.

Para subir el código fuente el desarrollador tendrá que elegir el proyecto, la plataforma al que desea subir. Para acceder esta pantalla tiene la opción dentro de la barra de navegación.

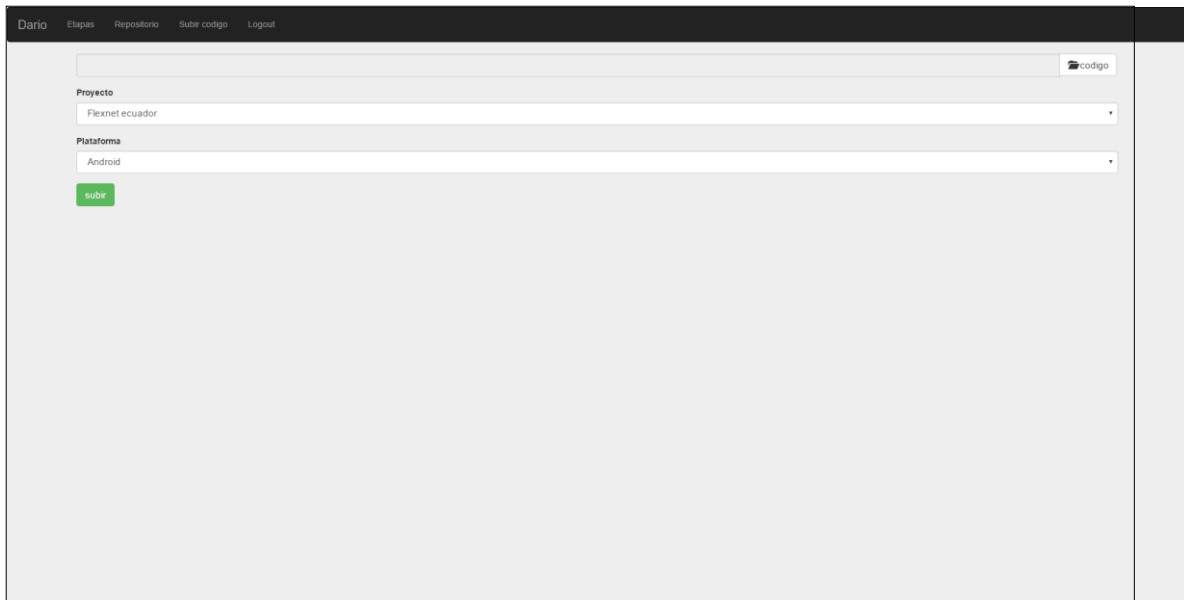


Figura20: Pantalla para subir código fuente.

5.3 Repositorio.

En la pantalla de repositorio se listarán los nombres de los proyectos creados, donde se podrá cada uno tendrá carpetas para las plataformas Android, iOS y Web. Cada una de estas carpetas tendrá las diferentes versiones de código fuente del proyecto al que corresponde el cual podrá ser descargado.

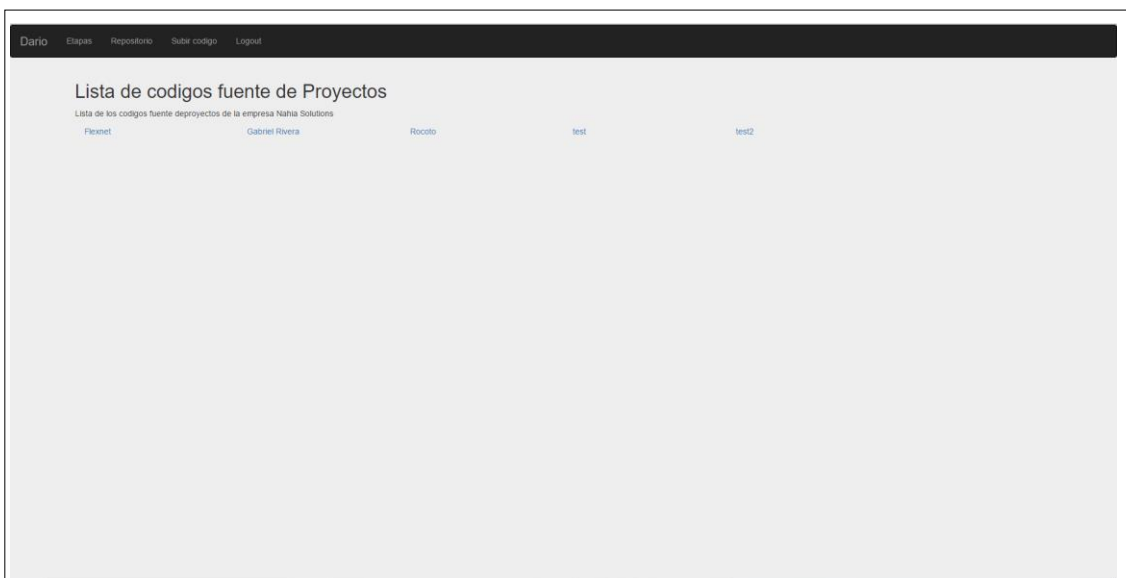


Figura 15: pantalla de repositorio.

Anexo 3 Carta de conformidad

Quito, 2 de Febrero de 2016

Señores UDLA

Presente.-

De nuestra consideración:

Por la presente, yo Ing. Natasha Valencia Narváz, en mi calidad de Gerente de Producto la empresa NAHIA SOLUCIONES S.A., quiero indicar que el software implementado por el Sr Rafael Falconí que desarrolló en su trabajo de titulación ha solucionado los inconvenientes a la hora de controlar los proyectos de nuestra empresa.

Atentamente



NAHIA SOLUCIONES S.A.
1792383161001

Ing. Natasha Valencia
Gerente de Producto
NAHIA SOLUCIONES S.A.