



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE CIENCIAS

**MANUAL DE APOYO PARA EL CURSO DE
INGENIERÍA DE SOFTWARE EN LA FACULTAD DE
CIENCIAS: GENERANDO UNA APLICACIÓN WEB Y
SU DOCUMENTACIÓN**

**REPORTE DE ACTIVIDAD
DE APOYO A LA DOCENCIA**

**QUE PARA OBTENER EL TÍTULO DE:
LICENCIADO EN CIENCIAS DE LA
COMPUTACIÓN**

P R E S E N T A :

GIOVANNI ALBERTO GARCÍA



**DIRECTORA DE TESIS:
M. EN C. MARÍA GUADALUPE ELENA
IBARGÜENGOITIA GONZÁLEZ**

2015



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

1.-Datos del alumno:

Alberto
García
Giovanni
56571786
Universidad Nacional Autónoma de México
Facultad de Ciencias
Ciencias de la Computación
30610227-3

2.-Datos de la tutora:

M. en C.
Ibargüengoitia
González
María Guadalupe Elena

3.-Datos de sinodal 1:

Dra.
Oktaba
Hanna Jadwiga

4.-Datos de sinodal 2:

Dr.
De la Cruz
Martínez
Gustavo

5.-Datos de sinodal 3:

M. en I.
Avilés
Rosas
Gerardo

6.-Datos de sinodal 4:

M. en D.
Martínez
Ramírez
Selene Marisol

7.-Datos del trabajo escrito:

Manual de apoyo para el curso de Ingeniería de Software en la Facultad de Ciencias:
Generando una aplicación web y su documentación
166 páginas
2015

| | |
|---|-----------|
| Prefacio | 6 |
| Introducción | 7 |
| Capítulo 1 Ingeniería de Software | 10 |
| 1.1 Concepto..... | 10 |
| 1.2 Contexto..... | 11 |
| 1.3 Método Inicial de Desarrollo de Software (MIDS) | 11 |
| Capítulo 2 Apoyos a los docentes del curso de Ingeniería de Software | 13 |
| 2.1 Ayuda para el planteamiento del <i>Enunciado del Problema</i> | 13 |
| 2.1.1 Instrucciones a los docentes para crear el <i>Enunciado del Problema</i> | 13 |
| 2.1.2 Plantilla para el <i>Enunciado del Problema</i> | 14 |
| 2.1.3 Recomendaciones generales para la creación del <i>Enunciado del Problema</i> | 16 |
| 2.1.4 Ejemplo a seguir para la creación del <i>Enunciado del Problema</i> | 16 |
| 2.2 <i>Guión para entrega de documentos</i> | 19 |
| Capítulo 3 Prácticas sociales | 23 |
| 3.1 Práctica PS1. Conformación del equipo | 23 |
| 3.1.1 Objetivo | 23 |
| 3.1.2 Entradas | 23 |
| 3.1.3 Resultados..... | 23 |
| 3.1.4 Actividades | 23 |
| 3.2 Práctica PS2. Definición de la comunicación en el equipo | 24 |
| 3.2.1 Objetivo | 24 |
| 3.2.2 Entradas | 24 |
| 3.2.3 Resultados..... | 24 |
| 3.2.4 Actividades | 25 |
| 3.3 Práctica PS3. Creación del repositorio común de documentos | 25 |
| 3.3.1 Objetivo | 25 |
| 3.3.2 Entradas | 25 |
| 3.3.3 Resultados..... | 25 |
| 3.3.4 Actividades | 25 |
| 3.4 Práctica PS4. Retrospectiva de la iteración | 26 |
| 3.4.1 Objetivo | 26 |
| 3.4.2 Entradas | 26 |
| 3.4.3 Resultados..... | 26 |
| 3.4.4 Actividades | 26 |
| 3.4.5 Plantilla..... | 27 |
| Capítulo 4 Prácticas administrativas | 30 |
| 4.1 Definir el proyecto | 30 |
| 4.1.1 Objetivo | 30 |
| 4.1.2 Entradas | 30 |

| | |
|---|-----------|
| 4.1.3 Resultados..... | 30 |
| 4.1.4 Actividades | 30 |
| 4.1.5 Plantilla..... | 30 |
| 4.2 Práctica PA1. Planificar el proyecto..... | 33 |
| 4.2.1 Objetivo | 33 |
| 4.2.2 Entradas | 33 |
| 4.2.3 Resultados..... | 33 |
| 4.2.4 Actividades | 33 |
| 4.2.5 Plantilla..... | 33 |
| 4.3 Práctica PA2. Planificar la iteración | 38 |
| 4.3.1 Objetivo | 38 |
| 4.3.2 Entradas | 38 |
| 4.3.3 Resultados..... | 38 |
| 4.3.4 Actividades | 38 |
| 4.4 Práctica PA3. Ejecutar el <i>Plan del Proyecto</i> de la iteración..... | 39 |
| 4.4.1 Objetivo | 39 |
| 4.4.2 Entradas | 39 |
| 4.4.3 Resultados..... | 39 |
| 4.4.4 Actividades | 39 |
| 4.5 Práctica PA4. Evaluar y controlar la iteración | 39 |
| 4.5.1 Objetivo | 39 |
| 4.5.2 Entradas | 40 |
| 4.5.3 Resultados..... | 40 |
| 4.5.4 Actividades | 40 |
| 4.6 Práctica PA5. Cerrar la iteración..... | 40 |
| 4.6.1 Objetivo | 40 |
| 4.6.2 Entradas | 40 |
| 4.6.3 Resultados..... | 40 |
| 4.6.4 Actividades | 40 |
| 4.7 Práctica PA6. Cerrar el proyecto..... | 41 |
| 4.7.1 Objetivo | 41 |
| 4.7.2 Entradas | 41 |
| 4.7.3 Resultados..... | 41 |
| 4.7.4 Actividades | 41 |
| 4.7.5 Plantilla..... | 41 |
| Capítulo 5 Prácticas para el desarrollo | 43 |
| 5.1 Práctica PD1. Requerimientos | 43 |
| 5.1.1 Objetivo | 43 |
| 5.1.2 Entradas | 43 |
| 5.1.3 Resultados..... | 43 |
| 5.1.4 Actividades | 43 |
| 5.1.5 Plantilla..... | 44 |
| 5.2 Práctica PD2. Diseño | 51 |

| | |
|--|--------------------------------------|
| 5.2.1 Objetivo | 51 |
| 5.2.2 Entradas | 51 |
| 5.2.3 Resultados..... | 51 |
| 5.2.4 Actividades | 51 |
| 5.2.5 Plantilla | 51 |
| 5.3 Práctica PD3. Construcción | 58 |
| 5.3.1 Objetivo | 58 |
| 5.3.2 Entradas | 58 |
| 5.3.3 Resultados..... | 58 |
| 5.3.4 Actividades | 58 |
| 5.3.5 Plantilla | 59 |
| 5.4 Práctica PD4. <i>Integración y pruebas</i> | 62 |
| 5.4.1 Objetivo | 62 |
| 5.4.2 Entradas | 62 |
| 5.4.3 Resultados..... | 62 |
| 5.4.4 Actividades | 62 |
| 5.4.5 Plantilla | 62 |
| Conclusiones | 67 |
| Anexos | 68 |
| A0 Carátula | 68 |
| Ejemplos | ¡Error! Marcador no definido. |
| A1 Retrospectiva de la iteración..... | 69 |
| A2 Definir el proyecto | 73 |
| A3 Planificar el proyecto..... | 75 |
| A5 Cerrar el proyecto | 80 |
| A5 Requerimientos | 81 |
| A6 Diseño | 114 |
| A7 Construcción | 125 |
| A8 Integración y pruebas..... | 131 |
| Bibliografía | 166 |

Prefacio

Antes de comenzar con la lectura de este trabajo, es preciso hacer unas aclaraciones:

- Este trabajo está dirigido a los alumnos que desean ahondar en la asignatura de Ingeniería de Software, por motivos escolares, dicho acercamiento se tomó desde la perspectiva de un curso escolarizado de un plan de estudios semestral.
- El capítulo dos, tal como se indica en su título, está dedicado a los docentes para ayudarles a crear la meta que deberán satisfacer los alumnos de su curso al término del mismo, y para proporcionarles un esquema temporal sobre las entregas y desarrollo del curso, siendo éste totalmente flexible y adaptable a las necesidades de cada curso.
- Los capítulos restantes se dividen conforme a las prácticas que integran al Método Inicial de Desarrollo de Software, conceptos que se retomarán más adelante. Siendo muy importante no confundir las prácticas del Método Inicial de Desarrollo de Software, con prácticas de laboratorio, mismas que son habituales en temas de computación.
- A lo largo de este trabajo están contenidas las plantillas para los productos que los alumnos deben generar durante el curso de Ingeniería de Software, aplicables a las etapas de desarrollo de software.
- Todas las plantillas son personalizables en cuanto a estilos y formato, mas no en cuanto a contenido, pues se contemplan todos los elementos básicos que deben conocer los alumnos de Ingeniería de Software.
- Las plantillas tienen contenido en color azul y subrayado (en caso de ser texto), el cual es texto o imágenes que se deben reemplazar con los datos, imágenes y diagramas concretos de cada equipo y su sistema. Las plantillas no presentan carátula para evitar repeticiones, sin embargo en el anexo A0 se proporciona una plantilla de la misma.
- Los ejemplos debidamente llenados de las plantillas se separaron del texto para brindar legibilidad al mismo y facilitar la consulta de este trabajo; los cuales se encuentran en los anexos.
- Es de suma importancia no considerar el orden de las plantillas, durante el desarrollo de un producto de software, presentado en este trabajo, ya que dicho orden alude a la clasificación de conforme al tipo de práctica mas no al desarrollo temporal del curso.

Introducción

El presente trabajo está encaminado a apoyar al curso de Ingeniería de Software de la carrera de Ciencias de la Computación de la Facultad de Ciencias, a través de material de apoyo tanto para los estudiantes como para los docentes en la aplicación del *Método Inicial de Desarrollo de Software*, que es el método que se sigue en el curso y se compone por tres tipos de prácticas (sociales, administrativas y de desarrollo).

El objetivo es proporcionar:

- Para los estudiantes, ayuda para que la construcción de la documentación esté estructurada y completa, y que a su vez cumpla con lo solicitado en este curso. Este apoyo se logra a través de *plantillas* de los documentos para seguir los estándares solicitados en el curso;
- Ejemplos concisos y completos de los productos solicitados en el curso;
- Para los docentes, un esquema sencillo y bien definido que los ayude a generar el *Enunciado del Problema*, que es la base del problema que se plantea a los estudiantes para construir soluciones de software; así como la repartición del material de enseñanza en el *Guión para entrega de documentos*.

Este trabajo surgió de la necesidad de proporcionar una guía completa y editable para la generación de documentación de las actividades de los estudiantes conforme van aprendiendo las prácticas del MIDS, teniendo como resultado bases de ayuda en la creación de su producto de software, lo que se reflejará en aspectos relevantes como lo son el tiempo de programación, la disminución de problemas en la creación e integración de las partes que integrarán los productos a entregar en cada práctica, ayuda en el reconocimiento de las habilidades y capacidades de todos y cada uno de los integrantes del equipo al avanzar en el desarrollo del software.

La estructura del trabajo se basó en los tres tipos de prácticas del MIDS. Para cada capítulo se presenta una breve introducción, descripción y contenido relevante a cada práctica; la lista de productos finales que se pretende obtener en la práctica, detallando cada uno de ellos y algunos consejos para poder trabajar de forma eficaz y adecuada a partir de *plantillas* que se presentan para cada producto.

También se presentan ejemplos de los productos de cada práctica, siguiendo las *plantillas* propuestas pero debidamente llenadas, siguiendo el desarrollo de un ejemplo real de un proyecto web.

En el curso se trabaja en *iteraciones*, es decir, se divide el proyecto en ciclos de trabajo, al término de los cuales, se obtiene un producto de software funcional aunque no cumpla con toda la funcionalidad requerida. Dicha manera de trabajar se llama *iterativa e incremental*, pues en las iteraciones se van agregando nuevas funcionalidades al producto de software.

Las partes que integran este trabajo son:

- Capítulo 1. Ingeniería de Software. Describe el concepto y el objetivo de la asignatura; así como la evolución de la misma a lo largo del tiempo; un contexto de la misma en la Facultad de Ciencias.
- Capítulo 2. Método Inicial de Desarrollo de Software. Describe el método, objetivo y fundamentos que motivaron su desarrollo; el procedimiento que deben seguir los docentes para plantear el *Enunciado del Problema* que llevará al proyecto de software.
- Capítulo 3. Prácticas sociales. Describe las prácticas que serán la base de una convivencia armónica y una buena comunicación para realizar los otros tipos de prácticas (administrativas y de desarrollo) sin impedimento alguno.
- Capítulo 4. Prácticas administrativas. Describe las prácticas para administrar y organizar el desarrollo del producto de software.
- Capítulo 5. Prácticas para el desarrollo. Describe las prácticas para llevar a cabo la creación del proyecto, desde entender el problema hasta la corrección de errores. Se compone de las siguientes practicas:
 - Requerimientos. Describe la identificación de requerimientos funcionales y no funcionales que conformarán el producto de software a partir del *Enunciado del Problema*.
 - Diseño. Describe la elección de la arquitectura del producto de software; la creación de paquetes de clases conforme a la arquitectura; la generación de los diagramas de secuencia que describen de una manera más técnica cada caso de uso; la creación del diagrama de navegación y el esquema de la base de datos.
 - Construcción. Describe la codificación de cada componente definido en la etapa de diseño basados en un ambiente de desarrollo (especificación de herramientas técnicas); así como la obtención de una métrica de calidad de software, con base en los errores de cada componente.
 - *Integración y pruebas de Software*. Describe la forma de integrar cada uno de los componentes codificados en la etapa de construcción; la aplicación de los casos de prueba, definidos en la etapa de requerimientos.
- Conclusiones. Describe brevemente las ventajas del presente trabajo con base a las experiencias y resultados obtenidos de la aplicación del Método Inicial de

Desarrollo de Software en cursos de Ingeniería de Software de la Facultad de Ciencias.

- Anexos. Contiene los ejemplos de todas y cada una de las plantillas que se enuncian a lo largo del trabajo.

Capítulo 1 Ingeniería de Software

1.1 Concepto

La Ingeniería de Software surge de la necesidad de poder ayudar a crear soluciones de software estables y robustas que cumplan con las necesidades de un cliente, en un tiempo razonable.

Dado que actualmente disponemos de definiciones variadas para la Ingeniería de Software, nos basaremos en una que abarque todas las actividades y aspectos relacionados con el desarrollo de software, definiéndola como sigue: "Es una disciplina de la ingeniería que se ocupa de todos los aspectos de la producción del software, desde sus pasos iniciales de la especificación del sistema, hasta el mantenimiento cuando está en uso" (Sommerville, Software Engineering, 2011).

Dado que existen numerosas definiciones para software, se puede malinterpretar lo que es en realidad, tal es el caso de la siguiente definición: "El software es el código, que resuelve un problema, generado por programas escritos en algún lenguaje de programación" (Sommerville, Software Engineering, 2011). Pero que sucede si sólo contemplamos el código como el software en sí, debido a la ilegibilidad del mismo, no podríamos tener un panorama claro de cuál es el problema que resuelve, ni cómo se pensó en su solución, tampoco tendríamos claros los módulos que conforman al software, no podríamos brindar mantenimiento ni adicionar módulos con facilidad; tendríamos que inspeccionar detalladamente el código para conocer y resolver las incógnitas y problemas. Razón por la cual, el software no es sólo el código, sino también: las estructuras de datos que permiten al código manipular la información; como los documentos asociados que proporcionan información descriptiva que está recopilada en documentos virtuales y físicos, mismos que describen la operación y uso del código (Pressman, 2010), lo cual será de gran utilidad para darle mantenimiento y en su debido caso extenderlo; siendo así más integral esta definición que la que sólo contempla al software como el código en sí.

Actualmente han surgido tendencias para generar productos de software que mezclan los enfoques disciplinados de los procesos (elaboración de documentación) con prácticas y métodos ágiles (iteraciones cortas, hacer partícipe al cliente en el desarrollo del producto), resultados de la experiencia de cada organización de software; adaptando así las metodologías al funcionamiento y provecho de cada empresa en particular (Ibargüengoitia, G., Oktaba, H).

1.2 Contexto

Ahora que tenemos una idea de qué es la Ingeniería de Software, brindaremos un panorama breve de la misma en la Facultad de Ciencias, institución que imparte dicha asignatura como parte de una formación básica para los futuros profesionistas de Ciencias de la Computación, principalmente.

La asignatura se imparte en el 6º semestre, tiene una duración de 16 semanas mismas que se recomienda dividir en dos iteraciones, la primera de 12 semanas (incluye dos semanas de preámbulo e introducción) y la segunda de 4 semanas. Es preciso aclarar que los tiempos no son equitativos, ya que es durante la primera iteración donde los alumnos aprenderán a trabajar en equipo, adquirirán el conocimiento de las tecnologías (Ambiente de desarrollo integrado, framework) a emplear y reconocerán sus capacidades y habilidades, además de generar la documentación de su producto de software. Durante la segunda iteración los equipos están organizados, conocen su forma de trabajo, las tecnologías que ellos habrán escogido, de modo que pueden realizar el trabajo de una manera más rápida y eficiente.

La herramienta principal de la asignatura, es el uso del Método Inicial de Desarrollo de Software, el cual se describe con más detalle en el capítulo siguiente.

1.3 Método Inicial de Desarrollo de Software (MIDS)

Un proyecto de software es “un esfuerzo temporal que lleva a cabo un equipo de trabajo usando un método para desarrollar, mantener o integrar un producto de software, que responda a las necesidades de un interesado específico bajo condiciones particulares” (Morales M., 2012).

Ahora que entendemos que es un proyecto de software nos enfocaremos en cómo realizarlo; y precisamente la herramienta que guía al curso para el desarrollo del mismo es el MIDS y como su nombre lo dice es un método inicial y completo para poder desarrollar software.

Un método es un marco de trabajo; un conjunto estandarizado, coherente, completo y consistente de prácticas (Morales M., 2012) y para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar; usado para estructurar, planificar y controlar el desarrollo de software.

Una práctica es una guía de trabajo a realizar, tiene un objetivo y recibe entradas para generar salidas (resultados), es una serie de pasos bien definidos que involucra actividades, restricciones y recursos para obtener un objetivo.

Las prácticas que componen al MIDS son:

- Prácticas Sociales: Incluyen actividades para conformar el equipo, establecer la comunicación, crear el repositorio común de documentos del equipo y hacer la retrospectiva de cada iteración sobre el desempeño del equipo.
- Prácticas Administrativas: Dentro de estas prácticas se incluyen planificar el proyecto y cada iteración, ejecutar el plan, evaluar la iteración, cerrar la iteración y cerrar el proyecto, elaborar y presentar el documento de aceptación de software.
- Prácticas para el Desarrollo: A través estas prácticas se ejecutarán un conjunto de actividades para generar el código del producto de software en sus distintas versiones. Las prácticas son:
 - *Especificación de Requerimientos de Software*: que consiste en comprender lo que el cliente o usuario del software espera que haga.
 - *Diseño de Software*: en donde se define la estructura del software que responderá a los requerimientos.
 - *Construcción de Software*: es la codificación de los componentes del software siguiendo el diseño y se llevan a cabo las pruebas de unidad.
 - *Integración y pruebas de Software*: consiste en integrar los componentes y probarlos en un sistema completo.
 - Entrega de software al cliente: consiste en la liberación al cliente del producto de software funcional, instalado, si lo requiere, y debidamente documentado.

Basados en MIDS, se propone generar una serie de *plantillas* que proporcionen una guía clara para el desempeño de las actividades en cada práctica del desarrollo del producto de software.

Dichas *plantillas* de las prácticas ayudarán a llenar debidamente los documentos que conformarán los resultados y proporcionarán al equipo de alumnos apoyo para terminar en tiempo y forma, con un producto de software de alta calidad.

Este trabajo, se enfocará en la documentación para una sola iteración, la otra es semejante.

Capítulo 2 Apoyos a los docentes del curso de Ingeniería de Software

2.1 Ayuda para el planteamiento del *Enunciado del Problema*

La asignatura de Ingeniería de Software tiene como parte importante el planteamiento del *Enunciado del Problema* que debe ser: algún problema cotidiano, que debido a la duración del curso (14 semanas en el caso de la Facultad de Ciencias) permita el desarrollo de un producto de software en el que trabaje un equipo de 4 personas; que ofrezca una solución funcional al final del desarrollo, que sea agradable e intuitivo para los usuarios, pues constituye en general su primer proyecto de desarrollo de software mediante un método.

“La especificación de dicho problema y su solución deseada se integran explícitamente en el *Enunciado del Problema*” (Swebok 3.0, 2014).

Cabe mencionar que a lo largo de los cursos de Ingeniería de Software que se han impartido, se ha detectado un patrón en lo que respecta a los distintos tipos de problemas aplicables al curso. Dicho patrón, a grandes rasgos, está compuesto por dos entidades básicas que interactúan entre sí, generalmente una ofrece servicios y la otra los solicita, “identificando causa y fuente” (Swebok 3.0, 2014).

En esta sección se desea apoyar a los docentes para que puedan escribir el *Enunciado del Problema*.

2.1.1 Instrucciones a los docentes para crear el *Enunciado del Problema*

A continuación se proponen las siguientes instrucciones, como base para la creación del *Enunciado del Problema*:

1. Elegir algún problema cotidiano donde estén involucradas, en el mejor de los casos, dos entidades (productor, consumidor), dicho problema puede ser por ejemplo: una escuela (profesores, alumnos), una taquilla (artistas, seguidores), un cine (películas, cinéfilos), un parque acuático (entradas, bañistas), un restaurante (mesas, comensales), etc. Otra posibilidad será que una sola entidad proporcionará los servicios y los consumirá, tal es el caso de una red social, en donde los usuarios interactúan entre ellos. Sin embargo, en este caso es preciso considerar la conveniencia de añadir una entidad autoritaria, tal como un administrador, que realice ciertas funcionalidades sobre la entidad.
2. Conforme se elige el problema, la definición de los usuarios son los que solicitan o proveen las funcionalidades. Sin embargo, en este paso nos daremos a la tarea de redactar la descripción del problema elegido, intentando restringir el número de

funcionalidades deseadas a un número aproximado a siete más menos dos, para que alcance el tiempo del semestre en su desarrollo.

3. Como resultado del paso anterior, tenemos una idea de las funcionalidades que se deberán proporcionar y qué usuarios. Sin embargo, es preciso ver si basta con los anteriores para cubrir plenamente la lógica de negocios del problema inicial. De esta manera se procederá a hacer un listado de todos y cada uno de las funcionalidades que pueden ser solicitados por los usuarios involucrados.
4. El paso siguiente será acotar el número de parámetros para cada funcionalidad, y reducir al mínimo el número de variantes de cada punto, por ejemplo, dando características específicas para cada uno de los datos o servicios solicitados. Esto se debe a que el tiempo de desarrollo consta de 14 semanas.
5. Releer completamente todo lo que se produjo como resultado de los puntos 1- 4 y ver si es apropiada la redacción del problema, las restricciones y el número de las funcionalidades que se proporcionarán. Cabe mencionar que hay algunas funcionalidades que no deben contarse como independientes, ya que conforman parte de alguna función principal, por ejemplo subir una fotografía de perfil para algún registro.
6. Adicionar al enunciado del problema la restricciones: “Requerimientos no funcionales que tienen que ver con las cualidades o características, que el cliente solicita para el funcionamiento del software, las cuales no tienen que ver directamente con las funcionalidades requeridas.” (Ibargüengoitia, G., Oktaba, H). Entre ellas se encuentran las siguientes:
 - a. interfaz con el usuario,
 - b. interfaz externa,
 - c. confiabilidad,
 - d. eficiencia,
 - e. seguridad,
 - f. compatibilidad,
 - g. mantenimiento,
 - h. portabilidad,
 - i. restricciones de diseño y construcción,
 - j. legales y reglamentarios

2.1.2 Plantilla para el *Enunciado del Problema*

Enseguida se muestra una plantilla para los docentes, cuyo contenido deberá ser el resultado de aplicar cada una de las instrucciones anteriores (1- 6), el formato de las plantillas para los alumnos no se aplicará para evitar confusiones:

Título del *Enunciado del Problema*

Aquí se mostrará la breve descripción del *Enunciado del Problema*.

Entidades

- Entidad 1: Breve descripción.
- Entidad 2: Breve descripción.
- Administrador: Breve descripción.

Funcionalidades

- 1.-Descripción breve y restringida de la funcionalidad 1.
- 2.-Descripción breve y restringida de la funcionalidad 2.
- ...
- 7.-Descripción breve y restringida de la funcionalidad 7.

Restricciones

Se pueden referir entre otros rubros, a los siguientes:

- Interfaz con el usuario: descripción de las características que permitan al software apoyar al usuario en sus tareas.
- Interfaz externa: relación o conexión con otros sistemas.
- Confiabilidad: disponibilidad, tolerancia a fallas y su recuperación.
- Eficiencia: límites de tiempo de respuesta o uso racional de espacios de almacenamiento.
- Seguridad: confidencialidad, integridad, sin rechazo, autenticación, información y datos protegidos contra el acceso no autorizado y disponible para los autorizados.
- Compatibilidad: coexistencia o interoperabilidad con otros sistemas.
- Mantenimiento: facilidad de comprensión y realización de modificaciones futuras.
- Portabilidad: facilidad de transferencia de un ambiente de ejecución a otro.
- Restricciones de diseño y construcción: las que imponga el cliente.
- Legales y reglamentarios: necesidades impuestas por leyes o reglamentos de otros.

2.1.3 Recomendaciones generales para la creación del *Enunciado del Problema*

Para la creación de nuestro *Enunciado del Problema* podemos hacer uso de lo siguiente:

- Para cada una de las entidades considerar los siguientes atributos como indispensables:
 - Nombre de usuario.
 - Correo electrónico.
 - Contraseña.
- Para las funcionalidades considerar las siguientes acciones para cada una de las entidades como indispensables:
 - Registrar cuenta.
 - Modificar datos de su cuenta.
 - Eliminar cuenta.
 - Envío de contraseña olvidada vía correo electrónico.
 - Confirmación de registro vía correo electrónico.
 - Iniciar/cerrar sesión.
Para el administrador, en su caso.
 - Bloquear/eliminar usuarios.
 - Enviar mensajes/correos a los usuarios.

2.1.4 Ejemplo a seguir para la creación del *Enunciado del Problema*

En seguida se muestra un ejemplo realizado con las instrucciones anteriores, es preciso mencionar que no se desglosa el resultado individual de cada instrucción aplicada, sino el resultado final, producto de todas las instrucciones:

Portal para personas que buscan pareja

Un grupo de amigos que están sin pareja ha solicitado la creación de un portal web, para dispositivos móviles, a sus compañeros de Ingeniería de Software que los apoye a buscar pareja. A dicho portal accederán los amigos solicitantes.

Se brindará el servicio de chat para comunicarse con otros usuarios. Adicionalmente al perfil, se dispondrá de una galería fotográfica de cada usuario y una página de inicio donde se anunciarán eventos organizados por los amigos solicitantes (presenciales) y se colocarán anuncios de perfiles que sean más propicios para cada usuario, conforme a sus gustos y actividades.

Entidades

- Usuario
Una persona que interesado por el servicio, que efectuó su registro al portal para probar el concepto, o bien pagar su suscripción.
- Administrador
Persona que tendrá privilegios elevados en el sistema y podrá consultar estadísticas referentes a los usuarios.

Funcionalidades

1. Permitir el registro (crear perfil) de las personas al portal con su nombre, apellido paterno, fecha de nacimiento, tipo de pareja que prefiere (sexo y preferencia sexual), correo electrónico y contraseña, lo anterior como datos esenciales, haciendo hincapié en que la completitud del perfil conlleva a un mejor proceso de encontrar pareja.
2. Adicionalmente se pedirán datos opcionales como foto de perfil, descripción personal, lugares favoritos, actividades favoritas, música favorita, intereses, fantasías, seriedad de la relación buscada (escala del 1-5) y nivel de perversión (escala del 1-5).
3. Permitir a los usuarios los cambios en sus datos de registro y la desactivación de la cuenta.
4. Los usuarios podrán actualizar su estado (sólo texto e hipervínculos); y publicar o eliminar fotografías personales en una galería única (con un límite de 5 fotos).
5. Los usuarios podrán chatear con otros usuarios.
6. Los usuarios podrán bloquear perfiles específicos.
7. El administrador podrá consultar estadísticas referentes a los usuarios (número de usuarios por sexo, edad, gustos, preferencia, entre otros).

Ejemplo del *Enunciado del problema 1/2* (Alberto, G. 2015)

Restricciones:

- Interfaz con el usuario: Debe ser limpia, clara y ordenada, accesible mediante un menú lateral
- Interfaz externa: No se pide compatibilidad para otros sistemas en esta versión.
- Confiabilidad: Debe estar disponible para los usuarios en cualquier momento, no debe desplegar errores extraños que sólo entienden los programadores.
- Eficiencia: La respuesta a todas y cada una de las acciones de los usuarios debe ser casi instantánea, pues los usuarios deben estar satisfechos con nuestro producto y deben sentir que su tiempo es valioso.
- Compatibilidad: No se espera interacción con otros sistemas para esta versión.
- Mantenimiento: Se espera que la documentación y el código fuente sean hechos de tal manera que sean entendibles y accesibles para cualquier programador.
- Portabilidad: El sistema será una página web para que sea ejecutado en diversos sistemas operativos.
- Restricciones de diseño y construcción: Debe ser hecho en lenguaje de programación Java, basados en los prototipos que se proporcionan.
- Legales y reglamentarios: Se debe dar a conocer el aviso de privacidad para la Protección de datos personales, mismo que se detalla en la ley de acceso a la información y protección de datos personales del Distrito Federal.

Ejemplo del *Enunciado del problema 2/2* (Alberto, G. 2015)

2.2 Guión para entrega de documentos

Dada la precisión de los tiempos en que se debe llevar a cabo todo el trabajo del curso, en especial un proyecto de software, se deben planificar todas y cada una de las tareas a realizar y la entrega de documentos que se deben hacer.

En esta sección se proporciona un guión que planifica las entregas de documentos que deben hacer los alumnos durante el curso de Ingeniería de Software. Este guión puede ajustarse conforme a las necesidades del grupo y los docentes y la duración del curso.

En este guión se especifica lo que deberá tratarse en cada semana por parte del profesor titular, el ayudante (si lo hay) y la entrega a realizar. El guión que se propone toma en cuenta un semestre de 16 semanas.

A continuación el *Guión para la entrega de documentos*:

| Semana | Temas (teoría) | Temas (ayudantía) | Entregas |
|--------|--|--|---|
| 1 | <ul style="list-style-type: none"> • Presentación del curso. • Entregar el temario de curso. • Introducción a la Ingeniería de Software. | <ul style="list-style-type: none"> • Reafirmar concepto de Ingeniería de Software y software. | |
| 2 | <ul style="list-style-type: none"> • Método del curso. • Prácticas sociales: Conformación del equipo. • Presentación del Enunciado de problema. | <ul style="list-style-type: none"> • Refirma concepto de prácticas y métodos. • Explicación de los roles. • Ayudar a la elaboración y organización del repositorio de documentos y de código. | |
| 3 | <ul style="list-style-type: none"> • Conceptos de planificación del proyecto. • Definición de casos de uso generales. | <ul style="list-style-type: none"> • Definición y explicación del producto de software. • Explicación del <i>Enunciado del Problema</i>. • Explicación y ejemplos de los diagramas de casos de uso. | <ul style="list-style-type: none"> • Plantilla del primer proyecto llena con datos del equipo. • <i>Estándar de Documentos</i>. |

| Semana | Temas (teoría) | Temas (ayudantía) | Entregas |
|--------|--|--|--|
| 4 | <ul style="list-style-type: none"> • Diagrama de casos de uso de la iteración. • Planificación de la primera iteración. • Casos de uso de la iteración: Detalle de casos de uso. • Casos de prueba. • Prototipo de la interfaz. • Requerimientos no funcionales. | <ul style="list-style-type: none"> • Ayudar a la elaboración del diagrama general de casos de uso. • Ayudar a la elaboración y organización del <i>Tablero de Seguimiento</i>. • Explicación de herramienta para generar prototipos de interfaces gráficas. | <ul style="list-style-type: none"> • <i>Plan del Proyecto</i> • Diagrama general de casos de uso. • Creación del <i>Tablero de Seguimiento</i> primera iteración. |
| 5 | <ul style="list-style-type: none"> • Refinamiento de lo anterior. | <ul style="list-style-type: none"> • Revisión y correcciones de la documentación generada (diagramas, prototipos). | <ul style="list-style-type: none"> • <i>Tablero de Seguimiento</i>. • Diagrama de casos de uso para la iteración. • Detalle de casos de uso. • Casos de prueba para los casos de uso. • Prototipo de interfaz • Requerimientos no funcionales. |
| 6 | <ul style="list-style-type: none"> • Diagramas de paquetes. • Definir ambiente de implementación. • Diagrama de distribución. | <ul style="list-style-type: none"> • Apoyar en la elaboración de los diagramas. • Apoyar en la selección de herramientas adecuadas para la implementación. • Revisar documentación de diseño. | <ul style="list-style-type: none"> • Requerimientos de la iteración refinados (lo anterior). |
| 7 | <ul style="list-style-type: none"> • Refinamiento de la interfaz. • Diagramas de clases. • Diagramas de secuencia. • Diagrama de navegación. • Diseño de la base de datos. | <ul style="list-style-type: none"> • Apoyar en la elaboración de los diagramas. • Revisar documentación de diseño. | <ul style="list-style-type: none"> • Diagrama de paquetes. • Ambiente de implementación. • Diagramas de distribución. |

| Semana | Temas (teoría) | Temas (ayudantía) | Entregas |
|--------|--|---|---|
| 8 | <ul style="list-style-type: none"> Refinamiento del diseño. | <ul style="list-style-type: none"> Apoyar en la elaboración de los diagramas. | <ul style="list-style-type: none"> Diagrama de clases. Diagrama de navegación. Diagrama de secuencia. Diseño de la base de datos. |
| 9 | <ul style="list-style-type: none"> Construcción. | <ul style="list-style-type: none"> Apoyar en dudas de código, presentar ejemplos útiles tales como mandar correo, iniciar sesión, etc. | <ul style="list-style-type: none"> Refinamiento de la documentación de diseño. |
| 10 | <ul style="list-style-type: none"> Construcción. | <ul style="list-style-type: none"> Apoyar en dudas de código. Introducir el concepto de pruebas unitarias para probar el código desarrollado. Revisar documentación de construcción. | <ul style="list-style-type: none"> Código de clases. |
| 11 | <ul style="list-style-type: none"> Integración de todo el producto de software. Prueba del software. | <ul style="list-style-type: none"> Ayudar a la integración del código correspondiente a cada caso de uso. Revisar documentación de <i>Integración y pruebas de Software</i>. | <ul style="list-style-type: none"> Documentación de construcción. Código de clases probado. Casos de uso integrados. |
| 12 | <ul style="list-style-type: none"> Entrega. Retrospectiva de la iteración. | <ul style="list-style-type: none"> Verificar el software entregado y su congruencia con la documentación entregada. Apoyar en la elaboración del documento de aprobación del producto de software. Apoyar en la elaboración de la retrospectiva de la iteración. | <ul style="list-style-type: none"> Documentación de <i>Integración y pruebas de Software</i>. Integración y prueba del producto de software. Entrega del producto de software. |

| Semana | Temas (teoría) | Temas (ayudantía) | Entregas |
|--------|--|---|--|
| 13 | <ul style="list-style-type: none"> Planificación de la 2° iteración con caminos alternos y excepciones. Requerimientos y diseño de la segunda iteración. | <ul style="list-style-type: none"> Ayudar y resolver dudas. Revisar la documentación de planeación y requerimientos de la 2° iteración. | <ul style="list-style-type: none"> Examen. Retrospectiva de la iteración. |
| 14 | <ul style="list-style-type: none"> Construcción | <ul style="list-style-type: none"> Revisar la documentación de construcción de la 2° iteración. | <ul style="list-style-type: none"> Plan de la 2° iteración. Requerimientos y diseño de la 2° iteración. |
| 15 | <ul style="list-style-type: none"> Integración. Prueba. Código de ética. | <ul style="list-style-type: none"> Ayudar y resolver dudas. Revisar documentación de <i>Integración y pruebas de Software</i> de la 2° iteración. | <ul style="list-style-type: none"> Código de la 2ª iteración. Documentación de construcción. |
| 16 | <ul style="list-style-type: none"> Retrospectiva 2° iteración. | <ul style="list-style-type: none"> Apoyar y resolver dudas. Ayudar a la elaboración de la retrospectiva de la 2° iteración | <ul style="list-style-type: none"> Integración y prueba de la 2ª iteración Documentación de <i>Integración y pruebas de Software</i> de la 2° iteración. Retrospectiva de la 2ª iteración |
| 17 | <ul style="list-style-type: none"> Entrega | <ul style="list-style-type: none"> Verificar el producto de software y la congruencia con la documentación de ambas iteraciones. | <ul style="list-style-type: none"> Entrega de la 2° iteración. |

Guión para la entrega de documentos (Ibargüengoitia, G. 2015)

Capítulo 3 Prácticas sociales

“Las prácticas sociales incluirán actividades para conformar el equipo, establecer la comunicación, crear el repositorio común de documentos del equipo y hacer la retrospectiva de cada iteración sobre el desempeño del equipo” (Ibargüengoitia, G., Oktaba, H).

3.1 Práctica PS1. Conformación del equipo

“*Un equipo de trabajo* es un grupo de practicantes que trabajan juntos de manera colaborativa durante un periodo de tiempo, para construir un producto de software” (Morales M., 2012), dicho propósito debe ser cumplido gracias al *espíritu de equipo* (Sommerville, Software Engineering, 2011), el cual es motivado por una identidad única que se genera en los miembros del equipo al crear el logo, y nombre de su equipo (primeras actividades realizadas como equipo de trabajo), al determinar que deberán cumplir una meta con éxito. Dicho espíritu motiva a los miembros a apoyarse unos a otros, a compartir sus conocimientos y habilidades.

3.1.1 Objetivo

- Poner nombre y logo que identifique al equipo.
- Conformar el equipo y repartir los roles.
- Entender el producto de software a desarrollar.

3.1.2 Entradas

- 4 (+-1) alumnos del curso de Ingeniería de Software que conformarán el equipo de trabajo.
- Tener un texto con el *Enunciado del Problema*.
- Conocer de forma general el método del curso (MIDS), mediante una introducción.

3.1.3 Resultados

- El equipo se conformó y tiene una identidad.
- Todos entienden el producto de software a desarrollar.

3.1.4 Actividades

- Reunir a los integrantes del equipo.
- Exponer, explicar, describir y asignar los roles: desarrollador (todos), responsable de equipo, responsable técnico, responsable de calidad, responsable de colaboración.

- Asignar nombre y logo al equipo:
 1. Cada miembro del equipo deberá aportar como mínimo dos propuestas de logo y nombre para el equipo, estos pueden ser independientes o compartir una vertiente ideológica.
 2. Posteriormente se reúnen los miembros y presentan sus propuestas de logo y nombre de equipo.
 3. Se realiza una votación por cada una de las propuestas y se toma la ganadora, en caso de que no sea distinguible una propuesta como ganadora, se votará nuevamente y se descartarán las menos queridas, y así hasta obtener un logo y un nombre ganador.
 4. Independientemente de las propuestas ganadoras, se pueden aportar ideas de mejora siempre y cuando todos estén conformes.

- Leer y entender el *Enunciado del Problema*.

3.2 Práctica PS2. Definición de la comunicación en el equipo

Según la Real Academia Española comunicación es el efecto de comunicar, entre cuyas definiciones destacan:

- Consultar con otros un asunto, tomando su parecer.
- Descubrir, manifestar o hacer saber a alguien algo.

De modo que en el desarrollo de un sistema es de vital importancia para la organización y colaboración estar involucrando a los demás miembros del equipo en nuestras actividades, ya sea para simplemente hacérselas saber (informar) o para tomar su parecer (opinión, consejo, crítica o ayuda).

3.2.1 Objetivo

- Establecer las formas de comunicación continua entre el equipo.

3.2.2 Entradas

- El equipo está conformado.

3.2.3 Resultados

- Se definen reuniones síncronas con las que el equipo dará seguimiento diario al proyecto, en dichas reuniones se establecerá comunicación directa entre los asistentes.

3.2.4 Actividades

- El equipo se reúne.
- Se define la comunicación síncrona.
- Se establece el horario de una reunión síncrona de 15 minutos, con todos los miembros, para cada día; en ellas se revisará:
 - El avance del proyecto.
 - La toma de decisiones específicas.

3.3 Práctica PS3. Creación del repositorio común de documentos

Toda la documentación generada a lo largo del desarrollo del sistema debe ser contenida en un repositorio común de documentos, el cual debe presentar una organización específica y clara, así como una estructura simple y asequible. “El conjunto de dichos repositorios genera una *infraestructura de la organización*, la cual servirá para futuras tomas de decisiones” (Ibargüengoitia, G., Oktaba, H) y el establecimiento o mejora de procesos.

3.3.1 Objetivo

- Crear el repositorio común de documentos para su resguardo, mismo que facilitará la comunicación continua entre el equipo.

3.3.2 Entradas

- El equipo requiere un repositorio común de documentos.
- *Plantillas* para los documentos.

3.3.3 Resultados

- Repositorio común de documentos para facilitar el acceso a los mismos.
- *Estándar de Documentos*.

3.3.4 Actividades

- El equipo se pone de acuerdo sobre la edición del formato de las *plantillas* que se les brindaron, para conformar su *Estándar de Documentos*, el cual les brindará originalidad e individualidad, así como presentación a sus documentos.
- El equipo elige una herramienta para el repositorio compartido y diseña una estructura bien organizada de fácil acceso para su repositorio de documentos. Se puede tomar el modelo siguiente:

Iteración n (1°, 2°, etc...)

- Entregables (versiones finales de los documentos de cada etapa).
- Versiones anteriores (versiones que en algún momento fungieron como finales).
- Planeación.
 - Recursos (diagramas, imágenes, otros).
- Requerimientos.
 - Recursos (diagramas, imágenes, otros).
- Diseño.
 - Recursos (diagramas, imágenes, otros).
- Construcción.
 - Recursos (diagramas, imágenes, otros).
- Integración y pruebas.
 - Recursos (diagramas, imágenes, otros).
- Cierre.
 - Recursos (diagramas, imágenes, otros).

3.4 Práctica PS4. Retrospectiva de la iteración

“Al terminar una iteración, los métodos ágiles proponen que se haga una retrospectiva de lo que sucedió con el desempeño del equipo en esa iteración. Hacer una retrospectiva significa que cada miembro del equipo reflexiona sobre su desempeño y el del equipo a fin de mejorar en la siguiente iteración” (Ibargüengoitia, G., Oktaba, H).

La palabra retrospectiva, según la Real Academia Española, nos dice: “Que se considera en su desarrollo anterior”, es decir que toma todo lo que ya ha sido generado para verificar qué es aprovechable y qué no.

3.4.1 Objetivo

- Revisar el desempeño del equipo en una iteración y proponer mejoras para la siguiente.

3.4.2 Entradas

- Se ha realizado una iteración en el desarrollo del software y se tienen actividades exitosas y negativas.

3.4.3 Resultados

- Retrospectiva de la iteración.

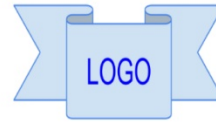
3.4.4 Actividades

- Identificar las actividades que el equipo hizo bien.
- Identificar las actividades que tuvieron un efecto negativo en la iteración.
- Proponer mejoras sobre el funcionamiento del equipo.
- Registrar el estado del progreso de la iteración en el *Tablero de Seguimiento*.
- Elaborar las métricas de calidad del producto de software.

3.4.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A1):

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Retrospectiva

1.-Registro de pros y contras de la iteración

| Número de votos | Actividades que se hicieron bien |
|-----------------|--|
| # | Descripción de la actividad. |
| ... | ... |

| Número de votos | Necesidades de mejora |
|-----------------|--|
| # | Descripción de la necesidad. |
| ... | ... |

Conclusiones de la retrospectiva:

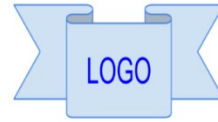
| Número de votos | Mejoras a aplicar en la siguiente iteración |
|-----------------|---|
| # | Descripción de la mejora. |
| ... | ... |

1

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Plantilla de la Retrospectiva de la iteración 1/2 (Alberto, G. 2015).

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



2.-Registro del estado del progreso de la iteración

[Hipervínculo al tablero de control en la iteración N \(hipervínculo público, en su defecto pdf\).](#)

Revisión del tablero de control

| | Trabajo pendiente | Incluir en la siguiente iteración |
|-----------------------|---------------------------------|-----------------------------------|
| Tareas no terminadas | (Lista de 1-N). | SÍ/NO |
| Defectos sin corregir | (Lista de 1-N). | SÍ/NO |
| Cambios pendientes | (Lista de 1-N). | SÍ/NO |

3.-Métricas de la iteración

| | Número | Porcentaje |
|-----------------------|-------------------|--------------------|
| Tareas terminadas | # | #% |
| Tareas no terminadas | # | #% |
| Defectos corregidos | # | #% |
| Defectos sin corregir | # | #% |

2

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Plantilla de la Retrospectiva de la iteración 2/2 (Alberto, G. 2015).

Capítulo 4 Prácticas administrativas

“Dentro de las prácticas administrativas se incluyen el planificar el proyecto y cada iteración, ejecutar el plan, evaluar la iteración, cerrar la iteración y cerrar el proyecto.” (Ibargüengoitia, G., Oktaba, H).

4.1 Definir el proyecto

Especificar de forma detallada el proyecto a desarrollar durante el curso, haciendo hincapié en que el objetivo no es sólo el sistema, sino la aplicación del MIDS.

4.1.1 Objetivo

- Delimitar los parámetros del proyecto a desarrollar, tiempo, fechas, equipo y sus roles.
- Determinar las actividades de cada iteración y sus fechas.

4.1.2 Entradas

- Equipo conformado
- Material para el curso por parte de los docentes.
- Conocimientos básicos para el desarrollo del sistema.
- Conocimiento del *Enunciado del Problema*.

4.1.3 Resultados

- La definición del proyecto de software cumple las especificaciones del *Enunciado del Problema*.

4.1.4 Actividades

- Llenar la plantilla que se ofrece para esta etapa proponiendo las fechas que se acordaron con anterioridad según el *Guión para entrega de documentos* del curso.

4.1.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A2):

Definir el proyecto de software

Primer Proyecto de Software.

Involucrados

Nombre del equipo: [Nombre de equipo.](#)

Logo del equipo:



Maestro(a): [Nombre completo de tu maestro\(a\).](#)

Ayudante: [Nombre completo de tu ayudante.](#)

Laboratorista: [Nombre completo de tu laboratorista.](#)

Periodo

Fecha de inicio: [Fecha de inicio del proyecto según el *Guión para entrega de documentos del curso.*](#)

Fecha de fin: [Fecha de fin del proyecto según el *Guión para entrega de documentos del curso.*](#)

Entradas

- [Título del proyecto con hipervínculo oculto al mismo.](#)
 - Los alumnos tienen los conocimientos requeridos para cursar la materia de Ingeniería de Software.
 - El (La) Maestro(a), ayudante y laboratorista tienen material de apoyo para el curso.
-

Resultados

- Los alumnos aprendieron y practicaron un método con prácticas sociales, administrativas y de desarrollo.

- [Características del producto de software con base al *Enunciado del Problema* \(1-n\).](#)
-

Método

Método Inicial de Desarrollo de Software.

Equipo de trabajo

- [Nombre del alumno 1, rol que desempeñará.](#)
- [Nombre del alumno 2, rol que desempeñará.](#)
- [...](#)

4.2 Práctica PA1. Planificar el proyecto

Un proyecto es un esfuerzo que se lleva a cabo durante un periodo de tiempo delimitado, con un equipo de trabajo asignado y con recursos disponibles para generarlo. En el caso del desarrollo de software, el resultado del proyecto es un producto de software.

4.2.1 Objetivo

- Establecer un conjunto de actividades para que “el proyecto logre obtener el resultado que se propone según el alcance especificado; en el tiempo acordado y al costo establecido” (Ibargüengoitia, G., Oktaba, H).

4.2.2 Entradas

- *Enunciado del Problema.*
- *Guión para entrega de documentos.*

4.2.3 Resultados

- *Plan del Proyecto*

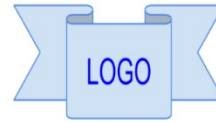
4.2.4 Actividades

- Identificar las necesidades del cliente y las funcionalidades requeridas.
- Entender la descripción del producto de software.
- Especificar el alcance del producto (lo que estará incluido y lo que no).
- Especificar los entregables para el cliente.
- Especificar las instrucciones de entrega.
- Especificar las actividades, recursos y calendarización de los mismos para el desarrollo óptimo y correcto.
- Identificación del costo y esfuerzo del proyecto.
- Identificación de los riesgos del proyecto.
- Especificar el control de versiones para el código producido.

4.2.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A3):

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Plan del Primer Proyecto de Software

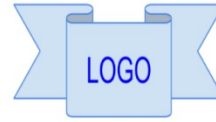
Contenido

Enunciado del Problema
Entregables
Fechas
Diagrama general de casos de uso
Definición de las iteraciones
Estrategia de control de versiones de código
Fecha y forma de entrega del producto

1

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Enunciado del Problema

[Objetivos principales del sistema y alcance en curso.](#)

Entregables

[Todo aquello que conforma el producto de software y será entregado al cliente \(revisar el *Guión para entrega de documentos*\).](#)

Fechas

Fecha de inicio: [dd/mm/aaaa \(revisar el *Guión para entrega de documentos*\).](#)
Fecha de fin: [dd/mm/aaaa \(revisar el *Guión para entrega de documentos*\).](#)

2

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)

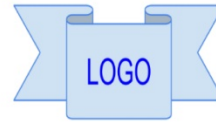


Diagrama general de casos de uso

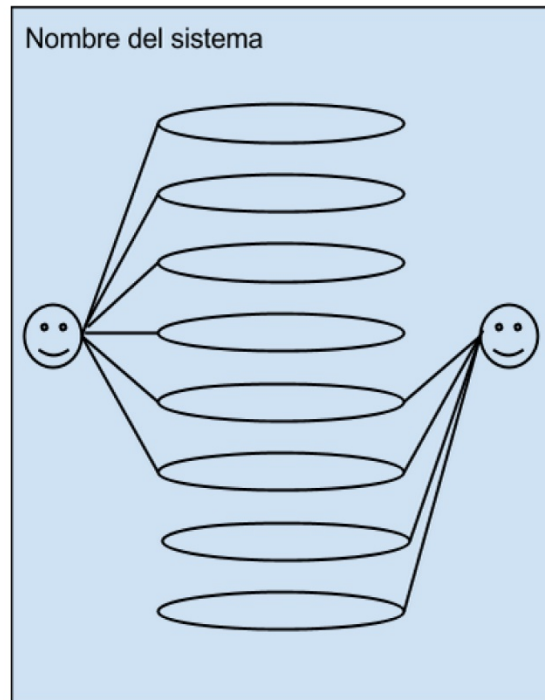


Diagrama general de casos de uso ([Nombre de la persona que elaboró, año](#))

Definición de las iteraciones

Primera iteración:

Inicio: [dd/mm/aaaa \(revisar el Guión para entrega de documentos\).](#)

Fin: [dd/mm/aaaa \(revisar el Guión para entrega de documentos\).](#)

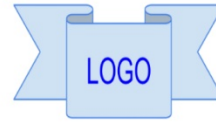
3

Elaboró: [Nombre de la persona que elaboró](#)

Fecha de elaboración: [dd/mm/aaaa](#)

Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



1. [Funcionalidad-caso de uso 1.](#)
2. [Funcionalidad-caso de uso 2.](#)
3. [...](#)

Segunda iteración:

Inicio: [dd/mm/aaaa \(revisar el Guión para entrega de documentos\).](#)

Fin: [dd/mm/aaaa \(revisar el Guión para entrega de documentos\).](#)

1. [Funcionalidad-caso de uso 1.](#)
2. [Funcionalidad-caso de uso 2.](#)
3. [...](#)

Estrategia de control de versiones de código

[Elección de la herramienta adecuada, ventajas y desventajas.](#)

Fecha y forma de entrega del producto

[Establecimiento de la fecha y forma específica de la entrega de software. Cómo y cuándo \(revisar el Guión para entrega de documentos\).](#)

4

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

4.3 Práctica PA2. Planificar la iteración

“Los métodos ágiles de desarrollo de software proponen que el desarrollo sea iterativo e incremental. Esto es, que el desarrollo se divida en varias iteraciones de tiempos cortos de aproximadamente 2 a 4 semanas. En cada iteración se realizan las actividades que permiten entregar un producto funcional, aunque no necesariamente útil para el cliente. El producto va incrementando sus funcionalidades con las entregas de cada iteración hasta satisfacer completamente las necesidades del cliente” (Ibargüengoitia, G., Oktaba, H).

4.3.1 Objetivo

- Delimitar el alcance de la iteración en curso.

4.3.2 Entradas

- *Plan del Proyecto.*

4.3.3 Resultados

- Definición del alcance de la iteración en curso.
- *Tablero de Seguimiento* de la iteración.

4.3.4 Actividades

- Acotar el *Enunciado del Problema*, si es posible, para que sólo enuncie los casos de uso de la iteración, de lo contrario, argumentar de forma breve los mismos.
- Definir el *Tablero de Seguimiento* para controlar y visualizar las actividades de la iteración en curso.
- Elaborar *Tarjetas de Trabajo* para cada actividad a realizar con los siguientes datos:
 - Nombre y rol del responsable.
 - Fecha de entrega real y esperada.
 - Título y descripción de la actividad.
- Conocer los servicios ofrecidos por herramientas de apoyo para la construcción de *Tableros de Seguimiento* como Trello (<https://www.trello.com>) y Leankit (<https://www.leankit.com>), el cual se empleó en el desarrollo del producto de software planteado en este trabajo.
- Elaborar el tablero con un esquema como el siguiente:

| Entradas | Requerimientos | | Diseño | | Construcción | | Integración y pruebas | | Entregas |
|----------|----------------|---|--------|---|--------------|---|-----------------------|---|----------|
| | P | T | P | T | P | T | P | T | |
| | | | | | | | | | |

Poner las tarjetas inicialmente en la columna P (planeadas) y conforme se van ejecutando las actividades se pasan a la columna de T (terminadas).

4.4 Práctica PA3. Ejecutar el *Plan del Proyecto* de la iteración

4.4.1 Objetivo

- Asegurarse que el *Plan del Proyecto* de la iteración se lleve a cabo.

4.4.2 Entradas

- *Plan del Proyecto*.

4.4.3 Resultados

- Evidencia de las reuniones del equipo de trabajo.
- Registro del estado del progreso actualizado (*Tablero de Seguimiento*).
- Repositorio del proyecto (código y documentos) actualizado y respaldado.
- Solicitudes de cambios que se analizan y evalúan revisando los impactos a los costos, calendario y requerimientos técnicos que se necesitan.
- Documentación de los cambios aprobados.

4.4.4 Actividades

- Registrar las evidencias de las reuniones que se llevaron a cabo.
- Desplazar las *Tarjetas de Trabajo* conforme a su estatus actual.
- Almacenar los nuevos documentos en el repositorio de documentos.
- Actualizar el código en el repositorio mediante el software de control de versiones.
- Añadir la documentación de los cambios que fueron aprobados.

4.5 Práctica PA4. Evaluar y controlar la iteración

“Se identifica si alguna desviación es significativa, si hay algún problema con respecto al costo, calendario y ejecución técnica. Se revisan los riesgos e identifican nuevos. Se documentan cambios y acciones correctivas que surjan de la evaluación y control” (Ibargüengoitia, G., Oktaba, H).

4.5.1 Objetivo

- Asegurarse que el *Plan del Proyecto* de la iteración se lleve a cabo a pesar de los contratiempos.

4.5.2 Entradas

- *Tablero de Seguimiento* (registro del estado de progreso del proyecto).

4.5.3 Resultados

- Registro del estado del proyecto.
- Registro de correcciones.
- Solicitudes de cambios.

4.5.4 Actividades

- Registrar en la retrospectiva las correcciones hechas.

4.6 Práctica PA5. Cerrar la iteración

Concluir las actividades planeadas para la iteración actual, resumiendo las tareas terminadas, las tareas pendientes, las buenas y malas actitudes y las aptitudes que ayudaron o afectaron el desempeño del equipo.

4.6.1 Objetivo

- Determinar el nivel de cumplimiento de los compromisos establecidos en el *Plan del Proyecto* de la iteración en curso.

4.6.2 Entradas

- *Tablero de Seguimiento*.
- Producto parcial y funcional
- Respectivo código fuente.
- Documentación respectiva en el repositorio común de documentos.

4.6.3 Resultados

- Registro del estado del proyecto.
- Registro de actividades pendientes.
- Métrica de calidad de software.

4.6.4 Actividades

- Registrar en la retrospectiva las actividades inconclusas.

- Calcular la métrica de calidad de software.
- Entregar el producto parcial junto con su documentación y código fuente.

4.7 Práctica PA6. Cerrar el proyecto

“La actividad de cierre de proyecto se realiza al final del proyecto cuando se proporciona la documentación completa y el producto solicitado al cliente, según los requerimientos del contrato” (Ibargüengoitia, G., Oktaba, H).

4.7.1 Objetivo

- Determinar el nivel de cumplimiento de los compromisos establecidos en el *Plan del Proyecto*.
- Entregar el producto de software, acompañado de la documentación pertinente.
- Brindar una instalación y/o capacitación conforme a lo acordado en la manera de entregar.

4.7.2 Entradas

- *Plan del proyecto*.
- Producto de software con su documentación y código fuente correspondientes.

4.7.3 Resultados

- Producto de software entregado.
- Registro de aceptación.
- Repositorio completo y resguardado.
- Código fuente correspondiente.

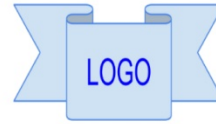
4.7.4 Actividades

- Elaborar y presentar para firma del cliente el registro de aceptación.
- Instalar el producto de software y/o capacitar en el caso debido.
- Entregar documentación y código fuente del proyecto.

4.7.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A4):

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Aceptación

Por medio de este conducto, yo [nombre del cliente](#), que en lo subsecuente se denominará "el cliente", me permito expresar mi completa y total satisfacción referente al desarrollo del producto de software [nombre del proyecto](#) por parte de [nombre del equipo](#), que en lo subsecuente se denominará "el equipo", mismo que me proporciona lo siguiente:

- Código fuente del producto de software.
- Instalación del producto de software conforme a lo establecido en la documentación del mismo.
- Capacitación para el uso del producto de software.
- Documentación generada a lo largo del desarrollo del producto de software.

Asimismo, deslindo de toda responsabilidad al equipo, pues quedan cubiertos todos y cada uno de los requerimientos que le fueron solicitados para el desarrollo del producto de software.

Firmo de conformidad:

Cliente (nombre y firma)

Firmo de conformidad:

Representante del equipo (nombre y firma)

[Lugar y fecha.](#)

1

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Capítulo 5 Prácticas para el desarrollo

“A través estas prácticas se ejecutarán un conjunto de actividades para generar el producto de software en sus distintas versiones”. (Ibargüengoitia, G., Oktaba, H).

5.1 Práctica PD1. Requerimientos

5.1.1 Objetivo

- Identificar los requerimientos del producto de software.
- Comprender lo que el cliente o usuario del sistema espera que haga.

5.1.2 Entradas

- Equipo conformado.
- Alcance de la iteración en curso.
- *Enunciado del Problema*.
- Diagrama general de casos de uso.
- *Tablero de Seguimiento* de la iteración.

5.1.3 Resultados

- Documento de *Especificación de Requerimientos de Software*, que incluye:
 - *Enunciado del Problema* acotado al alcance de la iteración,
 - Glosario de términos,
 - Diagrama de casos de uso de la iteración,
 - Detalle de los casos de uso de la iteración,
 - Prototipo de interfaz de usuario
 - Casos de prueba de la iteración,
 - Especificación de los requerimientos no funcionales.

5.1.4 Actividades

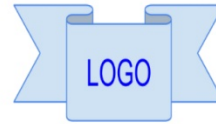
- Entender a detalle el *Enunciado del Problema* dentro del alcance de la iteración, tanto la parte de funcionalidades como los requerimientos no funcionales.
- Especificar los requerimientos no funcionales.
- Especificar los requerimientos funcionales:
 - Hacer el diagrama de casos de uso de la iteración conforme al alcance de la iteración en curso.
 - Para cada caso de uso de la iteración, documentar lo siguiente:
 - Flujo normal,
 - Flujo alternativo,
 - Flujo excepcional,

- Casos de prueba para cada flujo,
 - Prototipo de interfaz de usuario.
- Integrar la *Especificación de Requerimientos de Software* según la plantilla que sigue.

5.1.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A5):

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Especificación de Requerimientos de Software

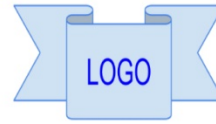
Contenido

Enunciado del Problema
Diagrama de casos de uso de la iteración
Glosario de términos
Detalle de casos de uso de la iteración
Prototipo de interfaz de usuario de la iteración
Casos de prueba de la iteración
Requerimientos no funcionales

1

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Enunciado del Problema

[Objetivos principales del sistema y alcance de la iteración en curso \(objetivos a cubrir en la iteración en curso\).](#)

Diagrama de casos de uso de la iteración

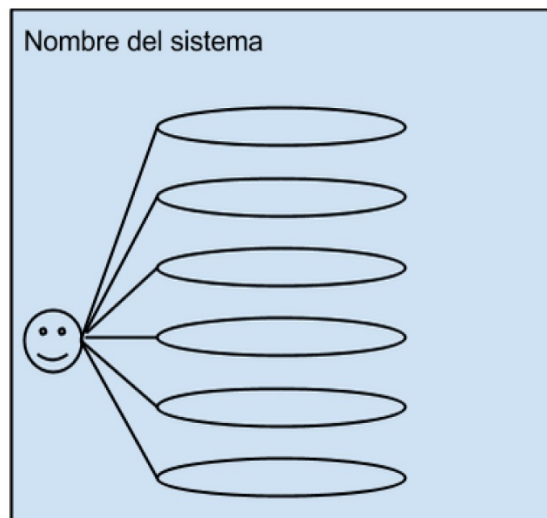
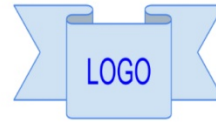


Diagrama general de casos de uso de la iteración ([Nombre de la persona que elaboró, año](#))

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Glosario de términos

| Término | Definición |
|---------------------|-----------------------------|
| A | Definición. |
| ... | ... |
| Z | Definición. |

Casos de uso detallados, prototipos de interfaz y casos de prueba

[Caso de uso 1: Nombre.](#)

1.-Detalle del caso de uso de la iteración:

[Actor\(es\)](#)

[Diagrama de caso de uso:](#)



Descripción: [Breve descripción del caso de uso.](#)

Precondiciones [Qué se necesita para empezar a ejecutar las acciones del flujo normal.](#)

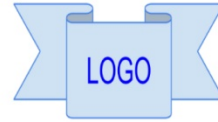
Flujo normal de eventos

| Actor(es) | | Sistema | |
|-------------------|-----------------------------------|---------------------|--|
| Paso | Acción | Paso | Acción |
| 1 | Acción del actor. | 2 | Respuesta del sistema. |
| 3 | Acción del actor. | ... | ... |

3

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Flujo alternativo de eventos

| ID | Nombre | Acción |
|--------------------|--|--|
| A1 | Acción alternativa del actor que permite que el sistema muestre un mensaje y el usuario vuelva a intentar la acción. | Respuesta alternativa del sistema. |

Flujo excepcional de eventos

| ID | Nombre | Acción |
|--------------------|---|--|
| E1 | Acción errónea o excepcional del actor que muestra mensaje y no se ejecuta la acción. | Respuesta con aviso o alerta de falla del sistema. |

Poscondiciones: [Cómo queda el estado del sistema y el nivel de satisfacción de la tarea.](#)

2.-Prototipo de interfaz de usuario de la iteración

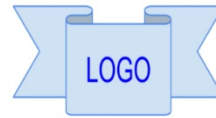


[Nombre del prototipo de interfaz \(Nombre de la persona que elaboró, año\)](#)

4

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



3.-Casos de prueba de los casos de uso de la iteración
Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|---|--|
| Descripción de entradas correspondientes a un flujo normal del caso de uso. | Descripción de salidas correspondientes a un flujo normal del caso de uso. |

...

Caso de prueba n (flujo alternativo)

| Entradas | Resultado esperado |
|--|---|
| Descripción de entradas correspondientes a un flujo alternativo del caso de uso. | Descripción de salidas correspondientes a un flujo alternativo del caso de uso. |

...

Caso de prueba m (flujo excepcional)

| Entradas | Resultado esperado |
|--|---|
| Descripción de entradas correspondientes a un flujo excepcional del caso de uso. | Descripción de salidas correspondientes a un flujo excepcional del caso de uso. |

...

[Caso de uso 2: Nombre.](#)

...

Requerimientos no funcionales

Interfaz con el usuario:

[Descripción de las características que permitan al software apoyar al usuario en sus tareas.](#)

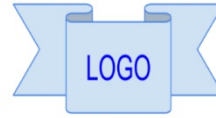
Interfaz externa:

[Relación o conexión con otros sistemas.](#)

5

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Confiabilidad

[Disponibilidad, tolerancia a fallas y su recuperación.](#)

Eficiencia

[Límites de tiempo de respuesta o uso racional de espacios de almacenamiento.](#)

Seguridad

[Confidencialidad, integridad, sin rechazo, autenticación, información y datos protegidos contra el acceso no autorizado y disponible para los autorizados.](#)

Compatibilidad

[Co-existencia o interoperabilidad con otros sistemas.](#)

Mantenimiento

[Facilidad de comprensión y realización de modificaciones futuras.](#)

Portabilidad

[Facilidad de transferencia de un ambiente de ejecución a otro.](#)

Restricciones de diseño y construcción

[Necesidades impuestas por el cliente.](#)

Legales y reglamentarios

[Necesidades impuestas por leyes o reglamentos de otros.](#)

6

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

5.2 Práctica PD2. Diseño

“En esta etapa se define la estructura del software que responderá a los requerimientos” (Ibargüengoitia, G., Oktaba, H).

5.2.1 Objetivo

- Definir la arquitectura del software.
- Diseñar los componentes principales del software.

5.2.2 Entradas

- Documento de *Especificación de Requerimientos de Software*.

5.2.3 Resultados

- Documento de *Diseño de Software*.

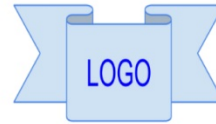
5.2.4 Actividades

- Establecer la arquitectura general del software mediante:
 - Elaboración del diagrama de paquetes o componentes.
 - Elaboración del diagrama de distribución.
 - Definir el ambiente de implementación.
- Diseñar los componentes principales del software:
 - Elaborar diagramas de clases.
 - Elaborar los diagramas de secuencia.
 - Elaborar el diagrama de navegación.
 - Diseñar la base de datos.
- Integrar el documento de *Diseño de Software* según la plantilla para esta etapa

5.2.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A6):

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Diseño de Software

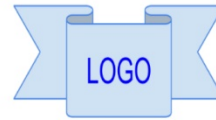
Contenido

Referencia al documento de *Especificación de Requerimientos de Software*
Diagrama de paquetes
Diagrama de distribución
Definición del ambiente de implementación
Diagramas de clases
Diagrama de secuencia
Diagramas de navegación
Diagrama entidad-relación de la base de datos

1

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Referencia al Documento de *Especificación de Requerimientos de Software*.

[Nombre y versión al documento respectivo, hipervínculo.](#)

Diagrama de paquetes

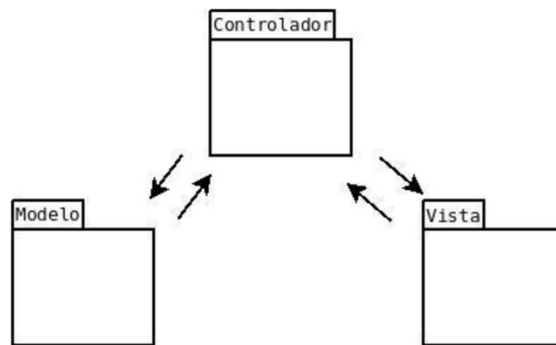
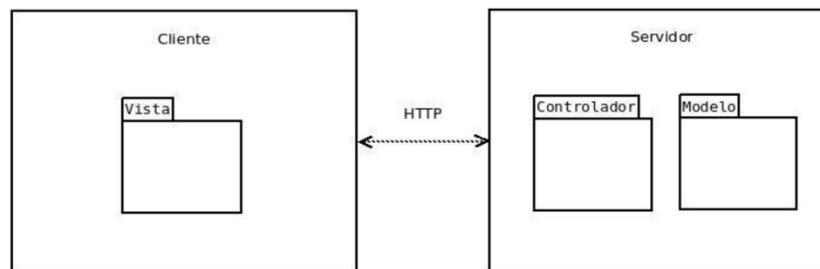


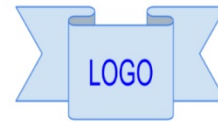
Diagrama de distribución



2

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
 Sistema: [Nombre del sistema](#)
 Iteración: [Número de la iteración](#)



Definición del ambiente de implementación

| Concepto | Herramienta | Versión |
|-----------------------------|-------------------------|--------------------------|
| Framework | Nombre. | Versión. |
| Diagramador UML | Nombre. | Versión. |
| Manejador de bases de datos | Nombre. | Versión. |
| Servidor de aplicaciones | Nombre. | Versión. |
| IDE | Nombre. | Versión. |
| Control de versiones | Nombre. | Versión. |

Diagramas de clases

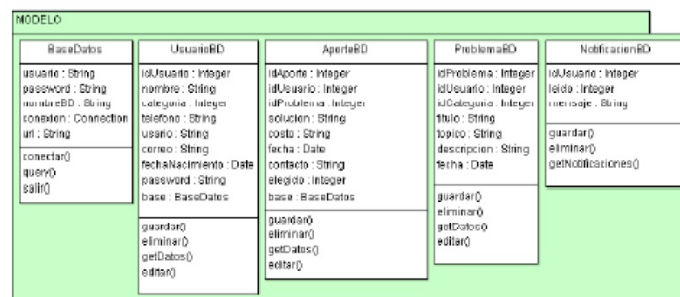


Diagrama de clases (Modelo) [\(Nombre de la persona que elaboró, año\)](#)

3

Elaboró: [Nombre de la persona que elaboró](#)
 Fecha de elaboración: [dd/mm/aaaa](#)
 Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
 Sistema: [Nombre del sistema](#)
 Iteración: [Número de la iteración](#)

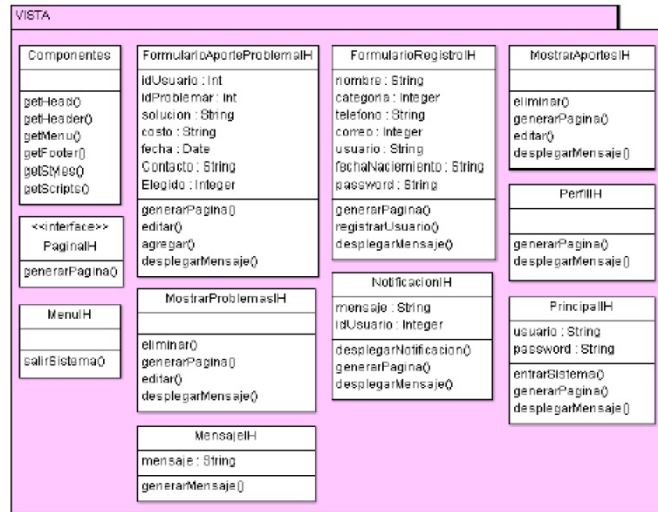
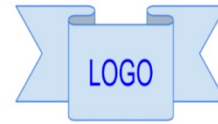


Diagrama de clases (Vista) [\(Nombre de la persona que elaboró, año\)](#)

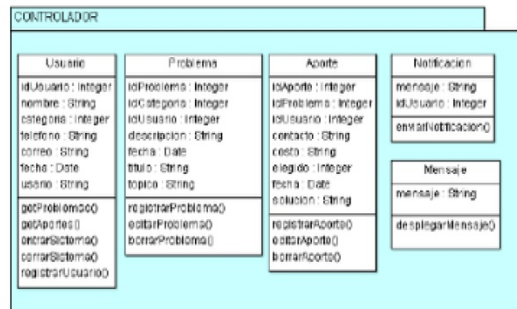


Diagrama de clases (Controlador) [\(Nombre de la persona que elaboró, año\)](#)

4

Elaboró: [Nombre de la persona que elaboró](#)
 Fecha de elaboración: [dd/mm/aaaa](#)
 Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)

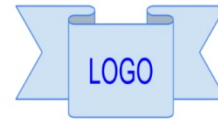


Diagrama entidad-relación de la base de datos

[De los casos de uso de la iteración actual, más los anteriores](#)

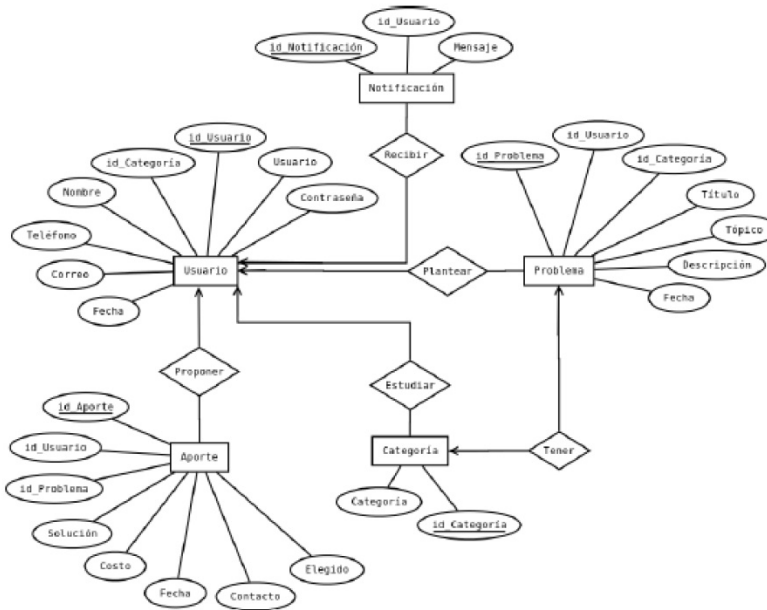


Diagrama entidad-relación de la base de datos ([Nombre de la persona que elaboró, año](#))

6

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

5.3 Práctica PD3. Construcción

“En esta etapa la actividad principal es la codificación de los componentes del software siguiendo el diseño y sus pruebas de unidad.” (Ibargüengoitia, G., Oktaba, H).

5.3.1 Objetivo

- Generación de código simple, adaptable y confiable para cada uno de los componentes definidos en el diseño.
- Probar el código generado.

5.3.2 Entradas

- Documento de *Especificación de Requerimientos de Software*.
- Documento de *Diseño de Software*.

5.3.3 Resultados


- Código de los componentes de software.
- Pruebas individuales de los componentes.
- Código de cada caso de uso integrado.
- Prueba de cada caso de uso.
- Documento de *Construcción de Software*.

5.3.4 Actividades

- Preparar el ambiente de implementación:
 - Instalar herramientas.
 - Definir estrategia de construcción con base a la arquitectura y el ambiente de implementación.
 - Establecer estándar para documentar el código.
 - Implementar las interfaces gráficas homologadas.
 - Construir la base de datos.
- Construir y probar el código de cada caso de uso:
 - Generar el código de cada componente.
 - Probar el código de cada componente.
 - Integrar el código de cada caso de uso.
 - Probar el código de cada caso de uso.
 - Integrar el documento de *Construcción de Software según la plantilla que sigue*.

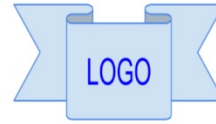
5.3.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A7):

| | | |
|--|--|---|
| Equipo: | Nombre de equipo |  |
| Sistema: | Nombre del sistema | |
| Iteración: | Número de la iteración | |
| <i>Construcción de Software</i> | | |
| Contenido | | |
| Referencia al documento de <i>Especificación de Requerimientos de Software</i> | | |
| Referencia al documento de <i>Diseño de Software</i> | | |
| Especificación del ambiente de desarrollo | | |
| Estándar de documentación y codificación en java | | |
| Lista de casos de uso construidos | | |
| Registro de defectos encontrados | | |
| Repositorio de código | | |
| 1 | | |
| Elaboró: | Nombre de la persona que elaboró | |
| Fecha de elaboración: | dd/mm/aaaa | |
| Versión: | Número de versión | |

Plantilla de *Construcción de Software* 1/3 (Alberto, G. 2015).

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Referencia al Documento de *Especificación de Requerimientos de Software*.

[Nombre y versión al documento respectivo, hipervínculo.](#)

Referencia al documento de *Diseño de Software*

[Nombre y versión al documento respectivo, hipervínculo.](#)

Especificación del ambiente de desarrollo

| Concepto | Herramienta | Versión |
|-----------------------------|-------------------------|--------------------------|
| Framework | Nombre. | Versión. |
| Diagramador UML | Nombre. | Versión. |
| Manejador de bases de datos | Nombre. | Versión. |
| Servidor de aplicaciones | Nombre. | Versión. |
| IDE | Nombre. | Versión. |
| Control de versiones | Nombre. | Versión. |

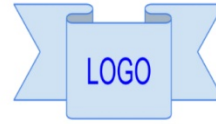
Estándar de documentación y codificación en java

[Nombre e hipervínculo al archivo.](#)

2

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Lista de casos de uso construidos

[Caso de uso 1](#)
[NombreClaseModelo](#) [Versión.](#)
[NombreClaseVista](#) [Versión.](#)
[NombreClaseControlador](#) [Versión.](#)
...

Registro de defectos encontrados

| Caso de uso | Número de casos de prueba | Defectos encontrados | Defectos corregidos |
|-------------------------------|---------------------------|----------------------|---------------------|
| Caso de uso 1 | # | # | # |
| ... | ... | ... | ... |
| Total | # | # | # |

Calidad de software = [Número de defectos corregidos /Número de casos de uso.](#)

Repositorio de código

[Código fuente, en su defecto hipervínculo público al repositorio de código de la versión final entregable.](#)

3

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

5.4 Práctica PD4. *Integración y pruebas*

“Esta etapa consiste en integrar los componentes y probarlos en un sistema completo” (Ibargüengoitia, G., Oktaba, H).

5.4.1 Objetivo

- Comprobar que los componentes construidos en la iteración funcionan bien, una vez que se integraron y también cuando se incorporan al software de iteraciones anteriores.

5.4.2 Entradas

- Documento de *Especificación de Requerimientos de Software*.
- Documento de *Diseño de Software*.
- Código de los componentes de software.
- Pruebas individuales de los componentes.
- Código de cada caso de uso integrado.
- Prueba de cada caso de uso.
- Documento de *Construcción de Software*.

5.4.3 Resultados

- Producto de software integrado y probado en la configuración del software.
- Documento de *Integración y pruebas de Software*.

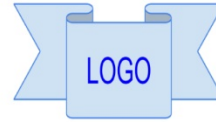
5.4.4 Actividades

- Integrar el producto de software:
 - Establecer una estrategia de integración.
 - Integrar el software.
 - Probar la integración.
- Probar el sistema integrado:
 - Preparar las pruebas.
 - Ejecutar las pruebas.
 - Corregir los defectos.
- Integrar el documento de *Integración y pruebas de Software* según la plantilla siguiente.

5.4.5 Plantilla

A continuación presentaremos la plantilla del documento de esta etapa (consulte el ejemplo en el anexo A8):

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Integración y pruebas de Software

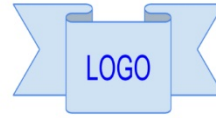
Contenido

Documento de *Especificación de Requerimientos de Software*
Documento de *Diseño de Software*
Referencia al documento de *Construcción de Software*
Especificación de la estrategia de integración
Configuración del software
Registro de defectos encontrados
Tabla de casos de prueba aplicados

1

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Referencia al documento de *Especificación de Requerimientos de Software*

[Nombre y versión al documento respectivo, hipervínculo.](#)

Referencia al documento de Diseño de Software

[Nombre y versión al documento respectivo, hipervínculo.](#)

Referencia al documento de Construcción de Software

[Nombre y versión al documento respectivo, hipervínculo.](#)

Especificación de la estrategia de integración

[Descripción de la estrategia de integración del software.](#)

Configuración del software

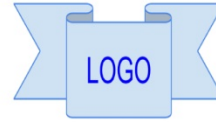
[Caso de uso 1: Nombre](#)
[NombreClaseModelo](#) [Versión.](#)
[NombreClaseVista](#) [Versión.](#)
[NombreClaseControlador](#) [Versión.](#)
...

2

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Plantilla de *Integración y pruebas de Software 2/4* (Alberto, G. 2015).

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



Registro de defectos encontrados

| Prueba del caso de uso | Responsable de la prueba | Identificador del caso de prueba | Defecto identificado |
|---|--|---------------------------------------|--|
| Nombre del caso de uso. | Nombre del integrante. | ID (N-1,A-1,E-1,...). | Descripción del defecto. |
| ... | ... | ... | ... |
| Total | # | | |

Tabla de casos de prueba aplicados

[Por caso de uso aplicar las pruebas](#)

[Caso de uso 1](#)

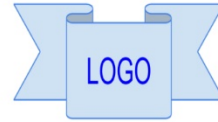
Caso de prueba 1

| Entradas | Resultado esperado |
|---|--|
| Descripción de entradas correspondientes a un flujo normal del caso de uso. | Descripción de salidas correspondientes a un flujo normal del caso de uso. |

3

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Equipo: [Nombre de equipo](#)
Sistema: [Nombre del sistema](#)
Iteración: [Número de la iteración](#)



[Nombre del prototipo de interfaz \(Nombre de la persona que elaboró, año\)](#)

⋮
⋮

4

Elaboró: [Nombre de la persona que elaboró](#)
Fecha de elaboración: [dd/mm/aaaa](#)
Versión: [Número de versión](#)

Conclusiones

El objetivo de este trabajo era apoyar al curso de Ingeniería de Software de la carrera de Ciencias de la Computación de la Facultad de Ciencias, a través de material de apoyo tanto para los estudiantes como para los docentes en la aplicación del *Método Inicial de Desarrollo de Software* (MIDS) que es el método que se sigue en el curso.

Como resultado del trabajo presentado es posible concluir lo siguiente:

- Los estudiantes tienen un apoyo completo y estructurado para generar su documentación basándose en *plantillas*, documentos prefabricados con partes modulares reemplazables por información propia. Dichos documentos contemplan todos los datos relevantes a cada etapa y el orden preferente para la realización de las actividades necesarias para el desarrollo de software en el curso de Ingeniería de Software.
- De igual manera los estudiantes cuentan con ejemplos reales de la documentación de un proyecto de software descrito en este trabajo, mismos que se encargarán de satisfacer y aclarar las dudas que pudieran surgir a lo largo del curso, pues ellos se han ido obteniendo de llenar las *plantillas* previamente mencionadas con la información obtenida durante la realización de un producto de software.
- Los docentes tienen una base sólida, bien definida y determinada que les permitirá, a partir de una serie de instrucciones, idear y consolidar un *Enunciado del Problema* basados en ciertos requerimientos; cuya formulación es clara y delimitada.

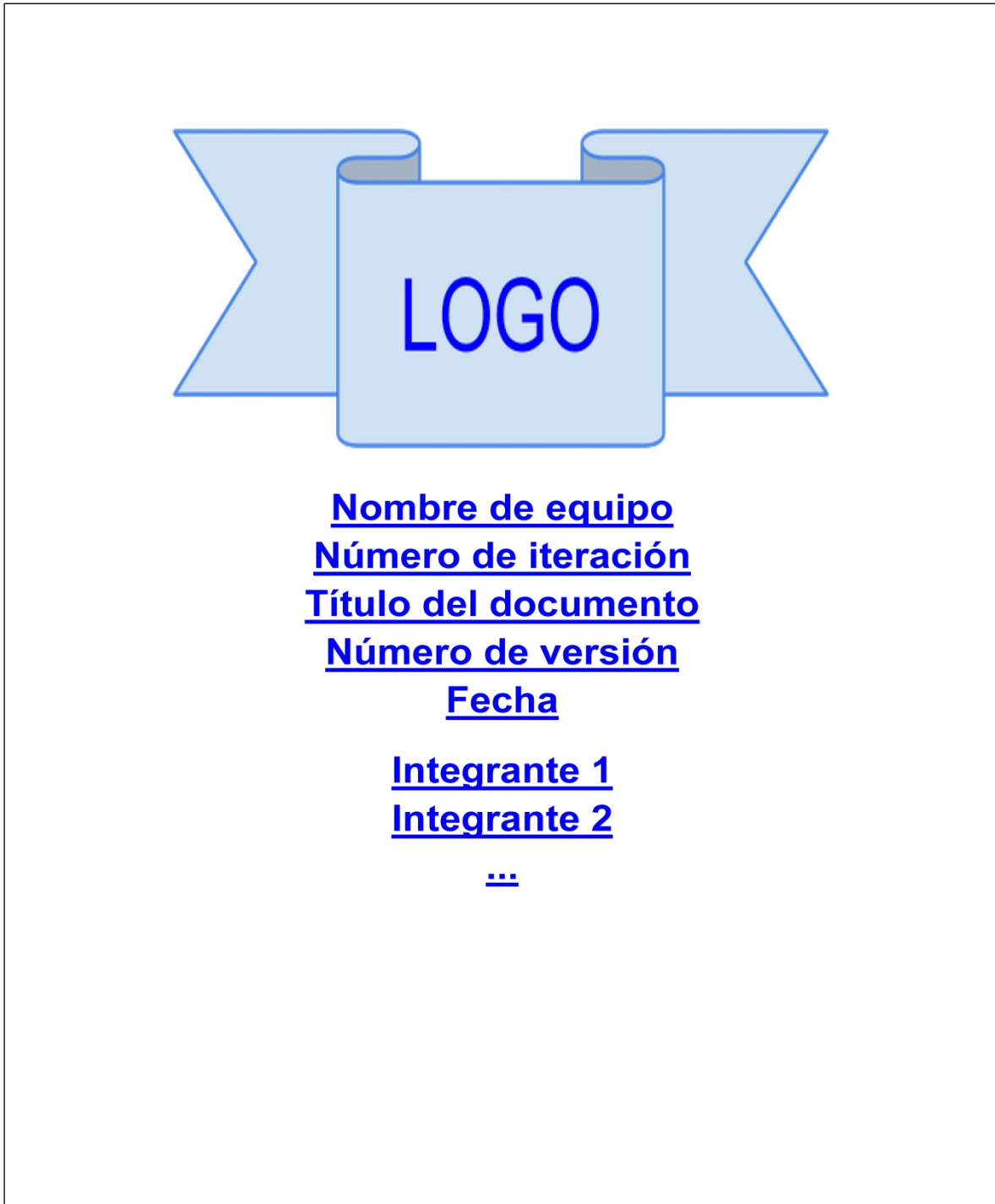
Con las conclusiones anteriores, es claro ver que se aporta una ventaja estratégica y de contenido al curso de Ingeniería de Software, pues los docentes tendrán las instrucciones para crear un *Enunciado del Problema* ideal (sencillo, bien determinado y de extensión media) para un curso de Ingeniería de Software inicial.

Los estudiantes tienen posibilidad de prestar mayor atención al contenido de la documentación que a la estructura y modularización, características que ya se brindan en las plantillas proporcionadas; obteniendo de esta ventaja, tiempo aprovechable para actividades de comunicación, aplicación de pruebas y desarrollo de código entre otras.

La dinámica de las clases para poder mostrar y revisar el contenido de los documentos se puede mejorar gracias a la consolidación de ejemplos completos de los mismos, pues estos pueden ser mostrados previamente como modelo a seguir para las primeras iteraciones y en iteraciones posteriores como punto de debate y discusión en cuanto a la relevancia de su contenido para cada equipo, es decir discutir que elementos de la documentación son cruciales y cuáles no.

Anexos

A0 Carátula



Plantilla de la carátula para los documentos (Alberto, G. 2015).

Plantillas llenas con un ejemplo del sistema Media Naranja
A1 Retrospectiva de la iteración



**Primera iteración
Retrospectiva
1.0
23/03/2015
Giovanni Alberto García**

Ejemplo de la Retrospectiva de la iteración 1/4 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Retrospectiva

1.-Registro de pros y contras de la iteración

| Número de votos | Actividades que se hicieron bien |
|-----------------|----------------------------------|
| 4 | Asignación de roles. |
| 3 | Juntas diarias. |
| 4 | Comunicación en tiempo real. |
| 3 | Colaboración. |
| 3 | Trabajo oportuno. |

| Número de votos | Necesidades de mejora |
|-----------------|---|
| 4 | Disminuir diferencias entre miembros. |
| 2 | Generar manuales de herramientas. |
| 2 | No perder de vista el tablero de control. |
| 4 | Mostrar avances al cliente. |

Conclusiones de la retrospectiva:

| Número de votos | Mejoras a aplicar en la siguiente iteración |
|-----------------|---|
| 3 | Generar manuales de herramientas. |
| 2 | Mostrar avance al cliente. |
| 4 | No perder de vista el tablero de control. |
| 4 | Comentar discrepancias oportunamente. |

1

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/03/2015
Versión: 1.0

Ejemplo de la Retrospectiva de la iteración 2/4 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



2.-Registro del estado del progreso de la iteración



Tablero de control (Alberto, G. 2015).

Revisión del tablero de control

| | Trabajo pendiente | Incluir en la siguiente iteración |
|------------------------------|-------------------|-----------------------------------|
| Tareas no terminadas | 0 | NO |
| Defectos sin corregir | 0 | NO |
| Cambios pendientes | 0 | NO |

2

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/03/2015
 Versión: 1.0

Ejemplo de la Retrospectiva de la iteración 3/4 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



3.-Métricas de la iteración

| | Número | Porcentaje |
|-----------------------|--------|------------|
| Tareas terminadas | 61 | 100% |
| Tareas no terminadas | 0 | 0% |
| Defectos corregidos | 32 | 100% |
| Defectos sin corregir | 0 | 0% |

3

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/03/2015
Versión: 1.0

Ejemplo de la Retrospectiva de la iteración 4/4 (Alberto, G. 2015).

A2 Definir el proyecto

Práctica 1 Definir el proyecto de software

Primer Proyecto de Software.

Involucrados

Nombre del equipo: Diablo Systems.

Logo del equipo:



Maestro(a): [Nombre completo de tu maestro\(a\).](#)

Ayudante: [Nombre completo de tu ayudante.](#)

Laboratorista: [Nombre completo de tu laboratorista.](#)

Periodo

Fecha de inicio: 09/02/2015

Fecha de fin: 15/05/2015

Entradas

- “Portal para personas que buscan pareja”.
 - Los alumnos tienen los conocimientos requeridos para cursar la materia de Ingeniería de Software.
 - El (La) Maestro(a), ayudante y laboratorista tienen material de apoyo para el curso.
-

Resultados

- Los alumnos aprendieron y practicaron un método con prácticas sociales, administrativas y de desarrollo.
 - Características del producto de software con base al *Enunciado del Problema*:
 - Registrar usuarios en el portal de búsqueda de pareja con datos básicos, como nombre, correo, contraseña, etc.
 - Permitir a los usuarios iniciar y cerrar sesión.
 - Permitir a los usuarios cambios en sus datos de registro.
 - Permitir a los usuarios publicar estados de texto e hipervínculos.
 - Permitir a los usuarios subir hasta 5 fotos en una galería única.
 - Permitir a los usuarios chatear con otros usuarios.
 - Permitir al administrador iniciar y cerrar sesión.
 - Consultar estadísticas referentes a los usuarios.
-

Método

Método Inicial de Desarrollo de Software.

Equipo de trabajo

- Giovanni Alberto García, responsable de equipo y desarrollador.
- Giovanni Alberto García, responsable técnico y desarrollador.
- Giovanni Alberto García, responsable de colaboración y desarrollador.
- Giovanni Alberto García, responsable de calidad y desarrollador.



Plan del Primer Proyecto de Software

1.0

16/02/2015

Giovanni Alberto García

Equipo: Diablo Systems
Sistema: Media Naranja



Contenido

Enunciado del Problema
Entregables
Fechas
Diagrama general de casos de uso
Definición de las iteraciones
Estrategia de control de versiones de código
Fecha y forma de entrega del producto

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/02/2015
Versión: 1.0

1

Ejemplo del Plan del Primer Proyecto de Software 2/5 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja



Plan del Primer Proyecto de Software

Enunciado del Problema

Un grupo de amigos que están sin pareja ha solicitado la creación de un portal web, para dispositivos móviles, a sus compañeros de Ingeniería de Software que los apoye a buscar pareja. A dicho portal accederán los amigos solicitantes.

Se brindará el servicio de chat para comunicarse con otros usuarios. Adicionalmente al perfil, se dispondrá de una galería fotográfica de cada usuario y una página de inicio donde se anunciarán eventos organizados por los amigos solicitantes (presenciales) y se colocarán anuncios de perfiles que sean más propicios para cada usuario, conforme a sus gustos y actividades.

Entregables

- El código fuente del sistema.
 - Los documentos generados durante las etapas de desarrollo del sistema:
 - Plan del Primer Proyecto de Software.
 - Documento de *Especificación de Requerimientos de Software*.
 - Documento de *Diseño de Software*.
 - Documento de *Construcción de Software*.
 - Documento de *Integración y pruebas de Software*.
-

Fechas

Fecha de inicio: 09/02/2015
Fecha de fin: 15/05/2015

2

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/02/2015
Versión: 1.0

Ejemplo del Plan del Primer Proyecto de Software 3/5 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja



Diagrama general de casos de uso

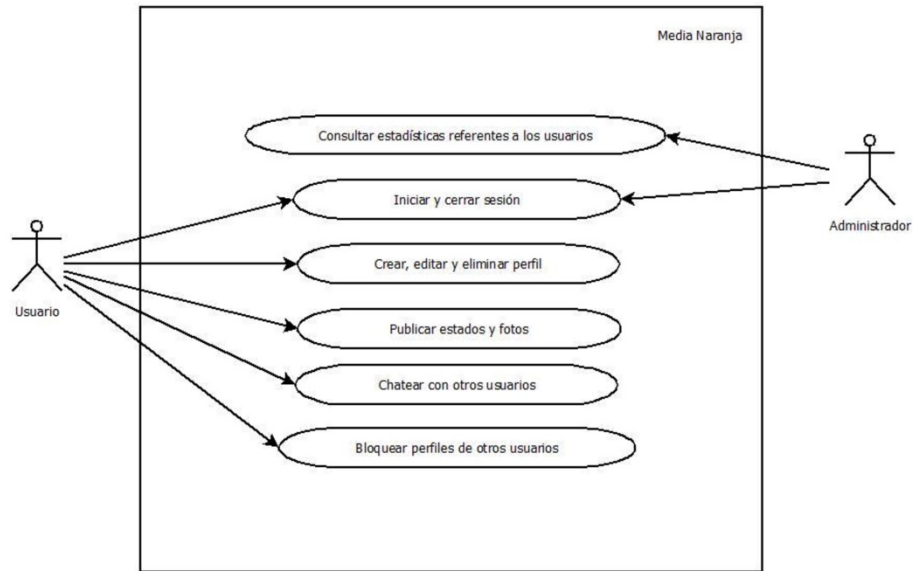


Diagrama general de casos de uso (Alberto, G. 2015)

Definición de las iteraciones

Primera iteración:

Inicio: 09/02/2015

Fin: 17/04/2015

1. Crear, editar y eliminar perfil de usuario.
2. Iniciar y cerrar sesión (usuario).
3. Publicar estados y subir fotos.
4. Chatear con otros usuarios.

3

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/02/2015
Versión: 1.0

Ejemplo del Plan del Primer Proyecto de Software 4/5 (Alberto, G. 2015).

Equipo: Diablos Systems
Sistema: Media Naranja



5. Bloquear perfiles de otros usuarios.

Segunda iteración:

Inicio: 18/04/2015
Fin: 15/05/2015

6. Iniciar y cerrar sesión (administrador).
7. Consultar estadísticas de los usuarios.

Estrategia de control de versiones de código

La herramienta de control de versiones que emplearemos para este producto de software es git, ya que últimamente ha tomado un gran auge debido a que es descentralizado, existen plataformas en línea que trabajan con él, tal es el caso de Github y Google Code. Existen algunos otros que posibilitan la creación de repositorios privados. El manejo de dicha herramienta es casi universal para los desarrolladores debido a la facilidad de los comandos y el aprendizaje.

Fecha y forma de entrega del producto

La entrega del producto de software se realizará el día 29 de mayo de 2015, mismo que será puesto en producción de manera local en una computadora usada como servidor, en donde podrán visualizar y emplear el sitio libremente, accediendo desde la misma red LAN. Asimismo, se proveerá una demostración breve sobre el uso y las secciones principales del mismo, que tendrá como objetivo una rápida introducción al conocimiento del producto.

4

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/02/2015
Versión: 1.0

Ejemplo del Plan del Primer Proyecto de Software 5/5 (Alberto, G. 2015).

A5 Cerrar el proyecto

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Aceptación

Por medio de este conducto, yo Juan Pérez, que en lo subsecuente se denominará “el cliente”, me permito expresar mi completa y total satisfacción referente al desarrollo del producto de software Media Naranja por parte del equipo Diablo Systems, que en lo subsecuente se denominará “el equipo”, mismo que me proporciona lo siguiente:

- Código fuente del producto de software.
- Instalación del producto de software conforme a lo establecido en la documentación del mismo.
- Capacitación para el uso del producto de software.
- Documentación generada a lo largo del desarrollo del producto de software.

Asimismo, deslindo de toda responsabilidad al equipo, pues quedan cubiertos todos y cada uno de los requerimientos que le fueron solicitados para el desarrollo del producto de software.

Firmo de conformidad:

[Firma]
Juan Pérez

Cliente (nombre y firma)

Firmo de conformidad:

[Firma]
Giovanni Alberto García

Representante del equipo (nombre y firma)

México DF a 20 de abril de 2015.

1

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/03/2015
Versión: 1.0

Ejemplo del Registro de aceptación 1/1 (Alberto, G. 2015).



Primera iteración
Especificación de Requerimientos de
Software
1.0
23/02/2015
Giovanni Alberto García

Ejemplo de *Especificación de Requerimientos de Software* 1/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Contenido

Enunciado del Problema
Diagrama de casos de uso de la iteración
Glosario de términos
Detalle de casos de uso de la iteración
Prototipo de interfaz de usuario de la iteración
Casos de prueba de la iteración
Requerimientos no funcionales

Elaboró: ¹ Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Especificación de Requerimientos de Software

Enunciado del Problema

Un grupo de amigos que están sin pareja ha solicitado la creación de un portal web, para dispositivos móviles, a sus compañeros de Ingeniería de Software que los apoye a buscar pareja. A dicho portal accederán los amigos solicitantes.

Se brindará el servicio de chat para comunicarse con otros usuarios. Adicionalmente al perfil, se dispondrá de una galería fotográfica de cada usuario y una página de inicio donde se anunciarán eventos organizados por los amigos solicitantes (presenciales) y se colocarán anuncios de perfiles que sean más propicios para cada usuario, conforme a sus gustos y actividades.

Para esta parte nos enfocaremos sólo a la parte de usuarios sin privilegios, es decir, los usuarios finales, personas que pretenden encontrar pareja.

Diagrama de casos de uso de la iteración uno

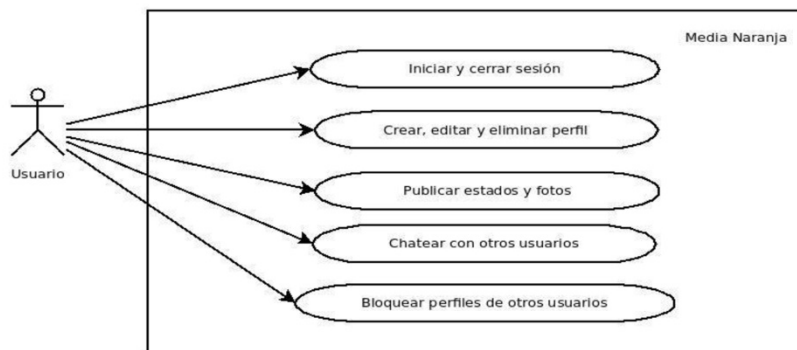


Diagrama general de casos de uso (Alberto, G. 2015).

2

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 3/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Glosario de términos

| Término | Definición |
|---------------|---|
| Administrador | Persona que tendrá privilegios ampliados en el sistema y podrá consultar estadísticas referentes a los usuarios. |
| Chatear | Acción de entablar una conversación escrita, será uno a uno. |
| Navegador | Herramienta para consultar sistemas de internet mediante protocolo HTTP y HTTPS, entre otros. |
| Nodo | Dispositivo, parte atómica que conforma una red. |
| Perfil | Página que concentra toda la información del usuario, sus mensajes y fotos. |
| Raíz | Página principal del sistema. |
| Usuario | Una persona que interesado por el servicio, que efectuó su registro al portal para probar el concepto, o bien pagar su suscripción. |
| Web | Dícese de todo lo relativo a internet, la red de redes. |

Casos de uso detallados, prototipos de interfaz y casos de prueba

Crear, editar y eliminar perfil

Actor: Usuario



3

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 4/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Crear perfil

1.-Detalle del caso de uso:

Descripción:

Este caso de uso se encargará de dar de alta un usuario en el sistema para que forme parte del mismo.

Precondiciones:

Desde la página principal del sistema acceder al hipervínculo de registro.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|--|---------|--|
| Paso | Acción | Paso | Acción |
| 1 | Llenar el formulario presentado con sus datos: correo electrónico, contraseña, confirmación de la misma y fecha de nacimiento. | | |
| 2 | Presionar el botón de registrar. | 3 | Realizar el registro preliminar del usuario para que éste tenga acceso al sistema. Llevar al usuario a la página principal y notificar resultado de la acción. |

Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|--|--|
| A1 | Ingresar un correo no válido (no incluir arroba) o una contraseña no válida (longitud menor a 8 caracteres). | Indicar al usuario cual es el campo que no se llenó debidamente para que lo corrija. |

4

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 5/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Flujo excepcional de eventos

| ID | Nombre | Acción |
|----|---|---|
| E1 | Pérdida de conexión con la base de datos. | Respuesta con aviso o alerta de falla del sistema, para que se reintente la acción deseada. |

Poscondiciones:

El usuario quedará registrado pero con un perfil parcial, mismo que deberá ser llenado más tarde de forma total.

2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|---|------------------------------|
| Introducir el correo, contraseña, confirmación y fecha de nacimiento siguientes: delirable@gmail.com password password 02/12/1990 | Esperar un registro exitoso. |

Caso de prueba 2 (flujo alternativo)

| Entradas | Resultado esperado |
|---|---|
| Introducir el correo, contraseña y confirmación siguientes: delirable@gmail.com pas pa 02/12/1990 | Esperar una advertencia sobre el correo no válido y una contraseña muy débil y corta. |

5

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 6/33 (Alberto, G. 2015).

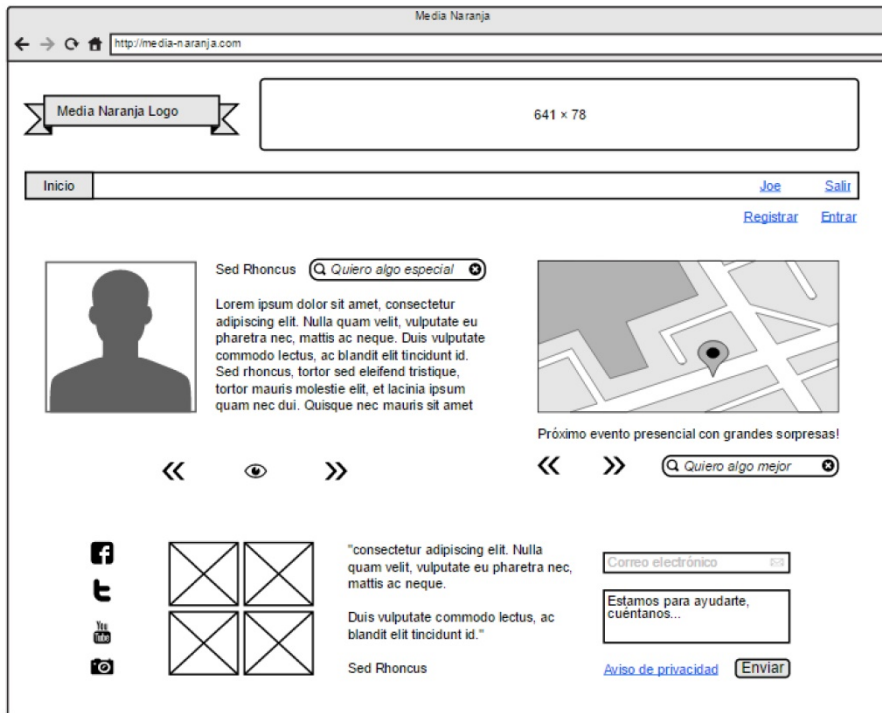
Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Caso de prueba 3 (flujo excepcional)

| Entradas | Resultado esperado |
|---|---|
| Se desactivará intencionalmente la base de datos. | El sistema regresará a la página principal indicando que la acción deseada no pudo completarse. |

3.-Prototipo



Prototipo de interfaz página principal (Alberto, G. 2015)

6

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software 7/33* (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Prototipo de interfaz crear perfil (Alberto, G. 2015).

Editar perfil

1.-Detalle:

Descripción:

Este caso de uso servirá para actualizar los datos del usuario para una mejor experiencia en el uso de los servicios del mismo, tal como encontrar perfiles afines.

7

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 8/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Precondiciones:

Acceder desde la página principal al hipervínculo del correo electrónico en la parte superior derecha de la página, el cual indica que hemos iniciado sesión.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|--|---------|--|
| Paso | Acción | Paso | Acción |
| 1 | Llenar el formulario presentado con sus datos nuevos: correo electrónico, contraseña vieja, contraseña nueva, nombre de usuario, fecha de nacimiento, foto de perfil, celular, descripción personal, lugares favoritos actividades usuales, música favorita, intereses, fantasías y anhelos estado civil, nivel de seriedad en la relación, nivel de perversión. | | |
| 2 | Presionar el botón de guardar. | 3 | Realizar el registro del usuario para que éste tenga completo su perfil y abrir la página de detalles de perfil notificando el resultado de la acción. |

Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|-----------------------------|---|
| A1 | Ingreso de datos inválidos. | El sistema indica los datos inválidos para que el usuario los corrija de inmediato. |

8

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 9/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Flujo excepcional de eventos

| ID | Nombre | Acción |
|----|---|---|
| E1 | Pérdida de conexión con la base de datos. | Respuesta con aviso o alerta de falla del sistema, para que se reintente la acción deseada. |

Poscondiciones:

El usuario habrá podido completar el registro/edición de su perfil con éxito, de modo que será redirigido a la página de detalles, donde puede apreciar sus datos recién modificados.

2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|--|--|
| Ingresar números en todos los campos solicitados a excepción de la contraseña y la confirmación que deben ser: password. | El sistema editará exitosamente los datos del registro y llevará al usuario a la página de detalles. |

Caso de prueba 2 (flujo alternativo)

| Entradas | Resultado esperado |
|---|---|
| El usuario ingresará datos erróneos como contraseñas sin coincidir o valores mal contruidos o no correspondientes al tipo que se pide (letras donde se requieran números) . | El sistema indicará dónde están los errores para que el usuario los corrija de inmediato. |

Caso de prueba 3 (flujo excepcional)

| Entradas | Resultado esperado |
|---|---|
| Se desactivará intencionalmente la base de datos. | El sistema regresará a la página principal indicando que la acción deseada no pudo completarse. |

9

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 10/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



3.-Prototipo

Prototipo de interfaz editar perfil (Alberto, G. 2015)

Eliminar perfil

1.-Detalle:

Descripción:

Este caso de uso servirá para desactivar un perfil de usuario.

10

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 11/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Precondiciones:

Estar en la página de edición de perfil.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|--|---------|---|
| Paso | Acción | Paso | Acción |
| 1 | Introducir la contraseña actual y hacer clic en el botón de desactivar perfil. | 2 | El sistema desactivará el perfil correspondiente. |

Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|--------|--------|
| NA | NA | NA |

Flujo excepcional de eventos

| ID | Nombre | Acción |
|----|---|---|
| E1 | Pérdida de conexión con la base de datos. | Respuesta con aviso o alerta de falla del sistema, para que se reintente la acción deseada. |

Poscondiciones:

El usuario habrá desactivado con éxito su perfil, de modo que éste no puede ser visualizado por nadie más en el sistema, hasta que ocurra un inicio de sesión.

2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|---|--|
| Marcar la casilla desactivar perfil y presionar el botón guardar. | Esperar que el sistema abra la página principal indicando que la acción deseada fue exitosa. |

11

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 12/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Caso de prueba 2 (flujo excepcional)

| Entradas | Resultado esperado |
|---|---|
| Se desactivará intencionalmente la base de datos. | El sistema regresará a la página principal indicando que la acción deseada no pudo completarse. |

3.-Prototipo

The screenshot shows a web browser window titled 'Media Naranja' with the URL 'http://www.media-naranja.com'. The page contains a form for profile management. At the top left is the 'Media Naranja Logo' and a placeholder for a 641 x 78 image. Below this is a navigation bar with 'Inicio', 'Perfil', 'Detalles', and 'Salir'. The main form area includes several input fields: 'Correo electrónico', 'Contraseña nueva', 'Confirmación', 'Nombre de usuario', 'Imagen de perfil', 'Celular', 'Lugar de residencia', 'Preferencia sexual', 'Seriedad en la relación', and 'Nivel de perversión'. There are also radio buttons for 'Hombre' and 'Mujer'. Text areas for 'Intereses en música, libros, lugares, actividades, escuela, trabajo, deportes, artistas...' and 'Fantasías y anhelos, deseos y aspiraciones para con la persona buscada...' are present. A 'Desactivar perfil' checkbox is also visible. At the bottom, there are social media icons (Facebook, Twitter, YouTube, Instagram), a 'Limpiar' button, a 'Guardar' button, and a section with placeholder text and an 'Enviar' button.

Prototipo de interfaz eliminar perfil (Alberto, G. 2015)

12

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

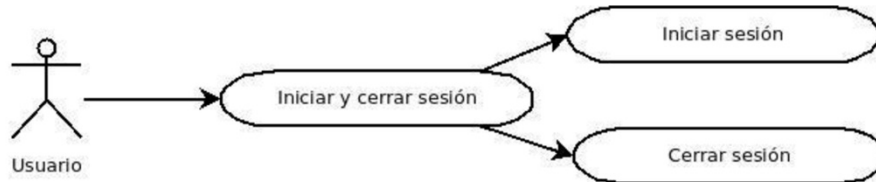
Ejemplo de *Especificación de Requerimientos de Software* 13/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Iniciar y cerrar sesión

Actor: Usuario



Iniciar sesión

1.-Detalle:

Descripción:

Este caso de uso se encargará de permitir la autenticación de un usuario en el sistema para que pueda acceder a los servicios del mismo.

Precondiciones:

Una vez debidamente registrado, desde la página principal deberá hacer clic en la esquina superior derecha al hipervínculo descrito como "Entrar".

Flujo normal de eventos

| Actor | | Sistema | |
|-------|---|---------|--|
| Paso | Acción | Paso | Acción |
| 1 | Llenar el formulario presentado con sus datos: correo electrónico y contraseña. | 2 | Realizar la autenticación del usuario y abrir la página principal. |

13

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|--|---|
| A1 | Ingresar un correo no válido (mal construido) o una contraseña no válida (distinta a la correcta). | Indicar al usuario que el correo y/o la contraseña son inválidos. |

Flujo excepcional de eventos

| ID | Nombre | Acción |
|----|---|---|
| E1 | Pérdida de conexión con la base de datos. | Respuesta con aviso o alerta de falla del sistema, para que se reintente la acción deseada. |

Poscondiciones:

Será otorgado el acceso al sistema para el usuario que realizó el inicio de sesión

2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|---|----------------------------|
| Introducir el correo y contraseña siguientes: delirable@gmail.com/password | Esperar un acceso exitoso. |

Caso de prueba 2 (flujo alternativo)

| Entradas | Resultado esperado |
|---|---|
| Introducir el correo y contraseña siguientes: delirable@gmail.com/pass | Esperar una advertencia de correo y/o contraseña inválidos. |

Caso de prueba 3 (flujo excepcional)

14

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

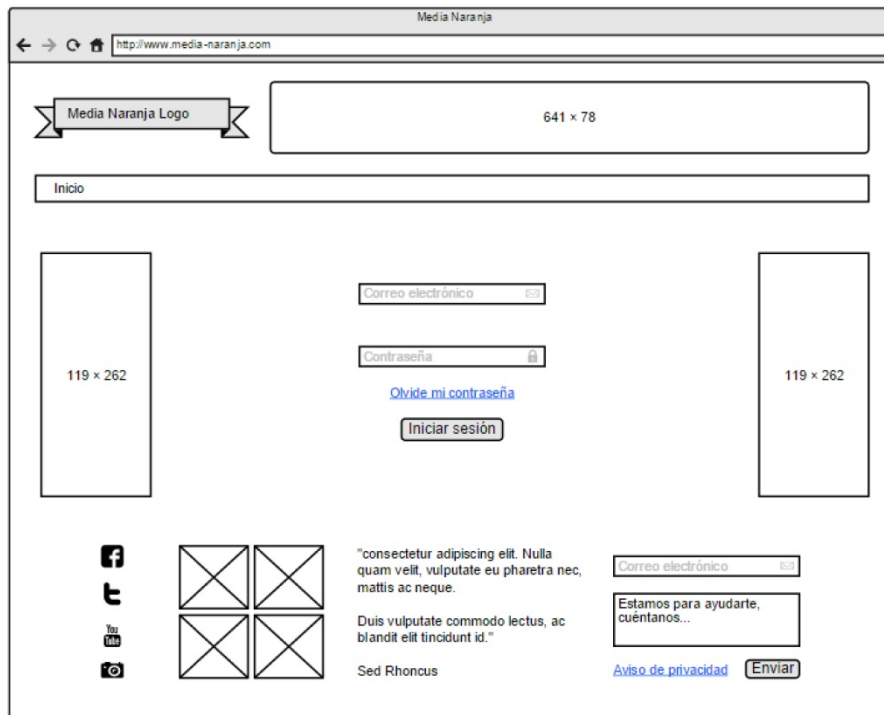
Ejemplo de *Especificación de Requerimientos de Software* 15/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



| Entradas | Resultado esperado |
|---|---|
| Se desactivará intencionalmente la base de datos. | El sistema regresará a la página principal indicando que la acción deseada no pudo completarse. |

3.-Prototipo



Prototipo de interfaz iniciar sesión (Alberto, G. 2015)

15

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 16/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Cerrar sesión

1.-Detalle:

Descripción:

Este caso de uso se encargará de no autorizar el uso de los servicios del sistema a menos que se inicie sesión nuevamente.

Precondiciones:

Estar en la página principal del sistema.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|---|---------|--|
| Paso | Acción | Paso | Acción |
| 1 | Hacer clic en el hipervínculo de la esquina superior derecha indicado como "Salir". | 2 | Cerrar la sesión del usuario y llevarlo a la página principal. |

Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|--------|--------|
| NA | NA | NA |

Flujo excepcional de eventos

| ID | Nombre | Acción |
|----|---|--|
| E1 | Pérdida de conexión con la base de datos. | Cerrar la sesión del usuario y llevarlo a la página principal. |

Poscondiciones:

El usuario ya no estará autenticado en el sistema, por lo que ya no podrá realizar acción alguna que requiera privilegios en el mismo.

16

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 17/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|--|---|
| Hacer clic en el hipervínculo "Salir". | Ser redirigido a la página principal y no tener privilegios de usuario. |

Caso de prueba 2 (flujo excepcional)

| Entradas | Resultado esperado |
|---|--|
| Se desactivará intencionalmente la base de datos. | Cerrar la sesión del usuario y llevarlo a la página principal. |

17

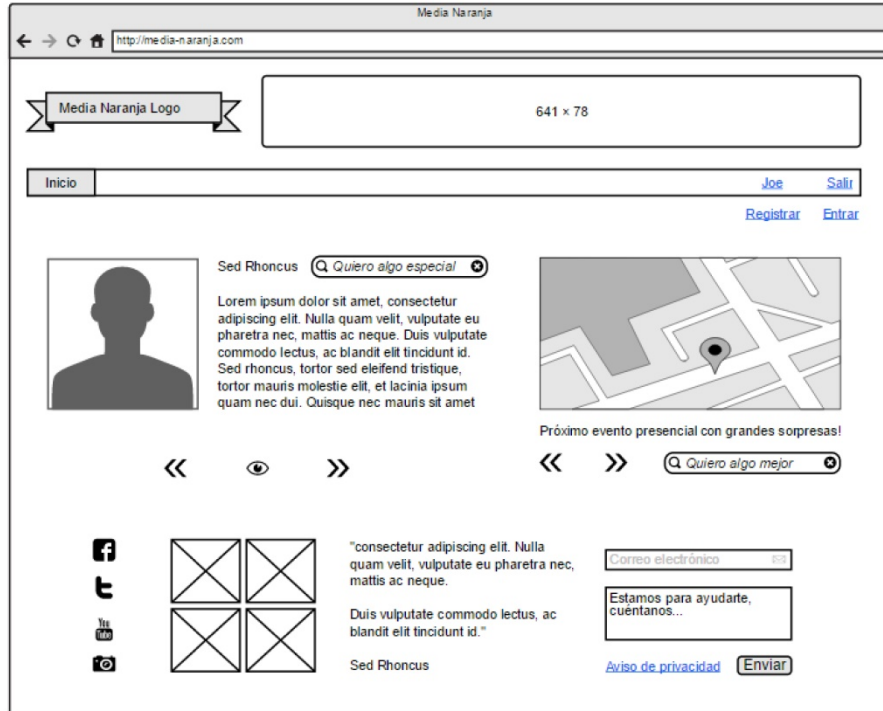
Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 18/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



3.-Prototipo



Prototipo de interfaz cerrar sesión (Alberto, G. 2015)

18

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

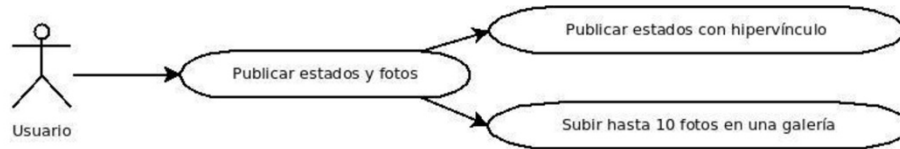
Ejemplo de *Especificación de Requerimientos de Software* 19/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Publicar estados y subir fotos

Actor: Usuario



Publicar estados con hipervínculo

1.-Detalle:

Descripción:

Este caso de uso se encargará de poder publicar estados de texto en un perfil, acompañados de un hipervínculo que será externo al sistema, es decir no será incrustado contenido alguno.

Precondiciones:

Estar en la página principal, y hacer clic en editar perfil y de ahí clic en perfil, para llegar al perfil propio o desde la principal hacer clic en la foto de perfil, nombre o descripción para llegar al perfil deseado.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|---|---------|--|
| Paso | Acción | Paso | Acción |
| 1 | Llenar el formulario presentado con un breve texto y un hipervínculo válido (opcional). | 2 | Realizar la publicación de lo antes escrito en el perfil que se visualiza. |

19

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|---|--|
| A1 | Dejar en blanco el texto o poner algo que no es hipervínculo. | Indicar al usuario que el texto no puede dejarse en blanco o que el hipervínculo no es válido. |

Flujo excepcional de eventos

| ID | Nombre | Acción |
|----|---|---|
| E1 | Pérdida de conexión con la base de datos. | Respuesta con aviso o alerta de falla del sistema, para que se reintente la acción deseada. |

Poscondiciones:

Será publicado el contenido previamente escrito en el perfil correspondiente

2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|--|---|
| Introducir el texto: "Enlace a google" y en el hipervínculo: https://google.com.mx . | Esperar una publicación de un enlace a google en el perfil correspondiente. |

Caso de prueba 2 (flujo alternativo)

| Entradas | Resultado esperado |
|--|--|
| Ningún texto e hipervínculo: "htt://google". | Esperar una advertencia de texto dejado en blanco e hipervínculo inválido. |

20

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 21/33 (Alberto, G. 2015).

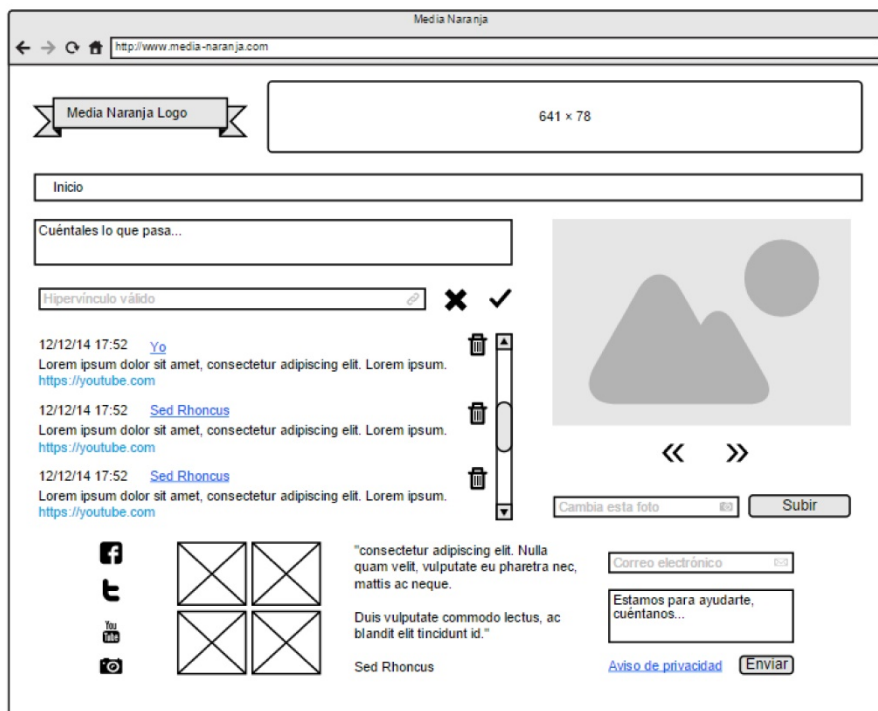
Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Caso de prueba 3 (flujo excepcional)

| Entradas | Resultado esperado |
|---|---|
| Se desactivará intencionalmente la base de datos. | El sistema regresará a la página principal indicando que la acción deseada no pudo completarse. |

3.-Prototipo



Prototipo de interfaz publicar estados con hipervinculos (Alberto, G. 2015).

21

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 22/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Subir hasta 5 fotos en una galería

1.-Detalle:

Descripción:

Este caso de uso se encargará de permitir al usuario compartir hasta 5 fotos en su perfil.

Precondiciones:

Estar en la página principal, y hacer clic en editar perfil y de ahí clic en perfil, para llegar al perfil propio.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|---|---------|---|
| Paso | Acción | Paso | Acción |
| 1 | Seleccionar un archivo para subir y presionar el botón de cargar. | 2 | Esperar a que se cargue la foto y se inserte en la galería. |

Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|--|--|
| A1 | Subir un archivo con formato inválido. | El sistema avisará al usuario que el formato no es válido, para esperar a que suba uno con formato válido. |

Flujo excepcional de eventos

| ID | Nombre | Acción |
|----|---|---|
| E1 | Pérdida de conexión con la base de datos. | Respuesta con aviso o alerta de falla del sistema, para que se reintente la acción deseada. |

Poscondiciones:

El usuario ya no estará autenticado en el sistema, por lo que ya no podrá realizar acción alguna que requiera privilegios en el mismo.

22

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 23/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|---|--|
| Seleccionar un archivo con formato jpg. | Cargar la foto e insertarla en la galería. |

Caso de prueba 2 (flujo alternativo)

| Entradas | Resultado esperado |
|---|---------------------------------------|
| Seleccionar un archivo con formato pdf. | Esperar un aviso de formato inválido. |

Caso de prueba 3 (flujo excepcional)

| Entradas | Resultado esperado |
|---|---|
| Se desactivará intencionalmente la base de datos. | El sistema regresará a la página principal indicando que la acción deseada no pudo completarse. |

23

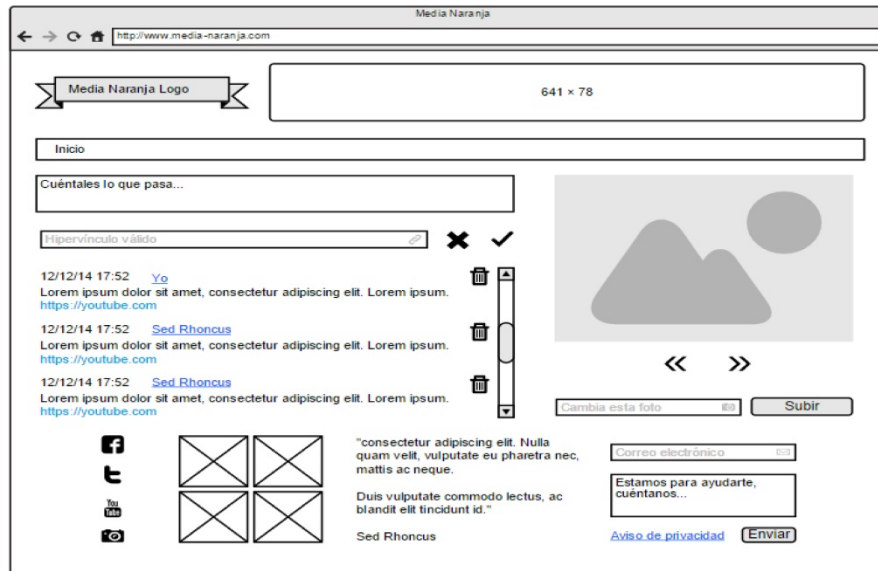
Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 24/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



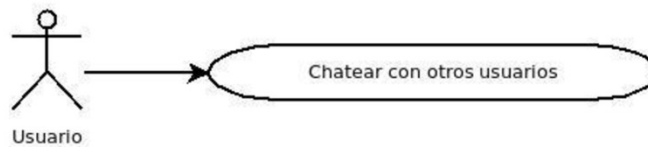
3.-Prototipo



Prototipo de interfaz subir hasta cinco fotos (Alberto, G. 2015)

Chatear con otros usuarios

Actor: Usuario



24

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 25/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



1.-Detalle:

Descripción:

Este caso de uso se encargará brindar comunicación escrita con otros usuarios en tiempo real.

Precondiciones:

Acceder a la pantalla de chat desde el menú principal.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|---|---------|------------------------------|
| Paso | Acción | Paso | Acción |
| 1 | Introducir el texto deseado en el campo inferior y pulsar enviar. | 2 | El sistema enviará el texto. |

| Actor | | Sistema | |
|-------|---|---------|--|
| Paso | Acción | Paso | Acción |
| 1 | Introducir el texto deseado en el campo inferior y pulsar borrar. | 2 | El sistema borrará el mensaje previamente escrito. |

Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|------------------------------|--|
| A1 | Enviar un mensaje en blanco. | Esperar un aviso del mensaje no puede ser vacío. |

Poscondiciones:

Será publicado el contenido previamente escrito en el cuadro de texto superior.

25

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 26/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|---|---|
| Introducir el texto: "Hola" y presionar enviar. | Esperar a que el texto se muestre en el cuadro de conversación. |

Caso de prueba 2 (flujo alternativo)

| Entradas | Resultado esperado |
|--------------------------|---|
| Enviar un mensaje vacío. | Enviar una advertencia de mensaje inválido. |

26

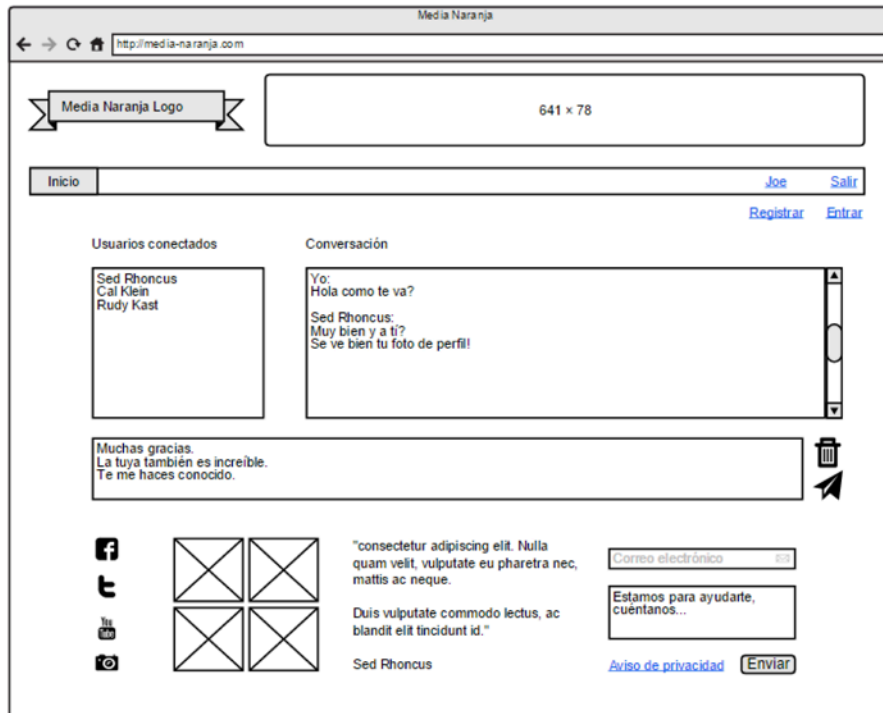
Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 27/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



3.-Prototipo



Prototipo de interfaz chatear con otros usuarios (Alberto, G. 2015)

27

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 28/33 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Bloquear perfiles

Actor: Usuario



1.-Detalle:

Descripción:

Este caso de uso permitirá a los usuarios bloquear personas que los hayan agredido u ofendido, en cuyo caso, no se podrá establecer contacto nuevamente entre ellos.

Precondiciones:

Estar en la página de detalles de alguna persona, para visualizar el botón de bloqueo.

Flujo normal de eventos

| Actor | | Sistema | |
|-------|--------------------------------|---------|--|
| Paso | Acción | Paso | Acción |
| 1 | Presionar el botón de bloqueo. | 2 | Bloquear a la persona para no desplegar resultado alguno de ella a la persona que hizo la denuncia. Abrir la página principal. |

Flujo alternativo de eventos

| ID | Nombre | Acción |
|----|--------|--------|
| NA | NA | NA |

Flujo excepcional de eventos

28

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 23/02/2015
 Versión: 1.0

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



| ID | Nombre | Acción |
|----|---|---|
| E1 | Pérdida de conexión con la base de datos. | Respuesta con aviso o alerta de falla del sistema, para que se reintente la acción deseada. |

Poscondiciones:

Será publicado el contenido previamente escrito en el perfil correspondiente.

2.-Casos de prueba

Caso de prueba 1 (flujo normal)

| Entradas | Resultado esperado |
|--|---|
| Seleccionar algún perfil, guardar la URL, bloquearlo e ingresar la URL previa. | Esperar a que se abra la página principal.. |

Caso de prueba 2 (flujo excepcional)

| Entradas | Resultado esperado |
|---|--|
| Se desactivará intencionalmente la base de datos. | El sistema regresará a la página principal indicando que la acción deseada no pudo completarse |

29

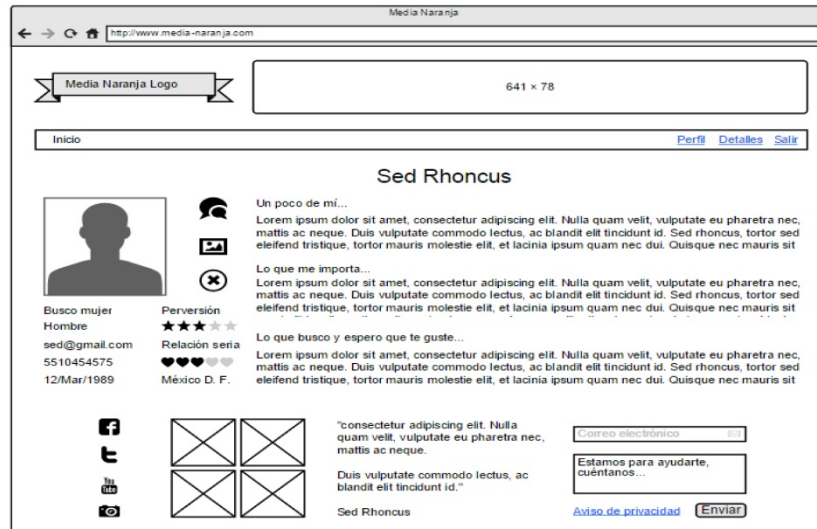
Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 30/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



3.-Prototipo



Prototipo de interfaz bloquear perfiles (Alberto, G. 2015)

Requerimientos no funcionales

Interfaz con el usuario:

La interfaz del sistema será limpia y ordenada, teniendo en cuenta la fácil accesibilidad a todos y cada uno de los elementos del mismo, permitiendo un esquema semejante a las actuales redes sociales, mostrando una línea gráfica armónica y agradable. Con un número máximo de 3 clics desde la raíz. Haciendo énfasis en los dispositivos móviles.

Interfaz externa:

Se planea a futuro hacer una conexión con Facebook, de modo que el registro sea mucho más fácil y rápido; de igual manera se hará una aplicación que emplee los amigos de Facebook para invitarlos a utilizar el sistema y pueda ayudarlos a darse cuenta si quieren entablar una relación entre ellos que ya son personas conocidas.

30

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 31/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Confiabilidad

Se pretende que el sistema sea puesto en producción de forma redundante y distribuida con un balanceador de carga para tener una disponibilidad 24/7.

Eficiencia

Se pretende un tiempo de respuesta de a lo más tres segundos en cada petición y se ha restringido el almacenamiento de archivos para no tener problemas de falta de espacio en el servidor.

Seguridad

El sistema tendrá contraseñas cifradas con el método de semilla para evitar exponer información en caso de que la base de datos se comprometa, pretende, en el futuro, el uso de un certificado SSL para el cifrado de datos entre el sistema y los usuarios, el acceso de los administradores estará disponible sólo por hipervínculo directo y mediante autenticación con contraseña.

Compatibilidad

Actualmente el sistema es independiente y no necesita de alguno otro para poder brindar sus servicios de forma óptima y eficiente, puede hacer uso de aplicaciones nativas del sistema operativo en donde se acceda a él, tal como el portapapeles, el selector de archivos y el navegador web.

Mantenimiento

El sistema es modular para poder facilitar la adición y mantenimiento del mismo, será comentado de acuerdo a los estándares para poder brindar soporte a los desarrolladores en caso de mantenimiento, de igual manera se tomarán en cuenta las buenas prácticas de programación para el lenguaje empleado, y en el debido caso el uso de algún framework que facilite la estructuración, diseño e implementación del mismo.

Portabilidad

El sistema será accesible desde un navegador web soporte para HTML5, CSS3 y JS de modo que podría decirse que el sistema es universal.

Restricciones de diseño y construcción

Se tiene contemplado el uso de Heroku (<https://www.heroku.com>) por la facilidad para alojar aplicaciones web, porque en su versión básica es gratuito y porque la puesta en producción es sencilla, se colocarán logos con símbolos de marca registrada, al igual que el nombre del sitio.

31

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 32/33 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Legales y reglamentarios

Se tiene que cumplir con la Ley de Transparencia y Protección de Datos Personales por el manejo y almacenamiento de los datos personales, de tal manera que se debe publicar un aviso de privacidad en un lugar asequible del sistema, así como un aviso de uso de cookies.

32

Elaboró: Giovanni Alberto García
Fecha de elaboración: 23/02/2015
Versión: 1.0

Ejemplo de *Especificación de Requerimientos de Software* 33/33 (Alberto, G. 2015).



Primera iteración
Diseño de Software
1.0
02/03/2015
Giovanni Alberto García

Ejemplo de *Diseño de Software* 1/11 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Contenido

Referencia al documento de *Especificación de Requerimientos de Software*
Diagrama de paquetes
Diagrama de distribución
Definición del ambiente de implementación
Diagramas de clases
Diagrama de secuencia
Diagramas de navegación
Diagrama entidad-relación de la base de datos

1
Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

Ejemplo de *Diseño de Software 2/11* (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Diseño de Software

Referencia al documento de *Especificación de Requerimientos de Software*

[Requerimientos, versión 1.0.](#)

Diagrama de paquetes

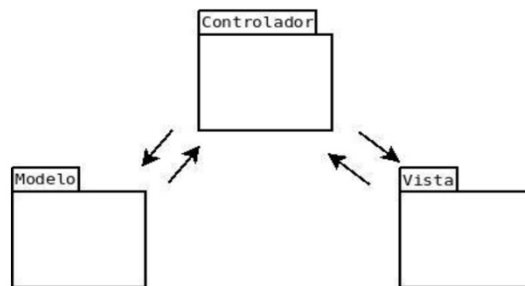
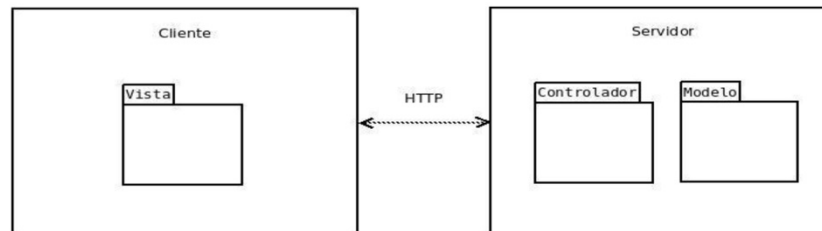


Diagrama de distribución



2

Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

Ejemplo de *Diseño de Software* 3/11 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Definición del ambiente de implementación

| Versión | Herramienta | Concepto |
|---------|-------------|-----------------------------|
| 2.3.7 | Play | Framework |
| 0.97.2 | Dia | Diagramador UML |
| 5.7 | MySQL | Manejador de bases de datos |
| NA | Heroku | Servidor de aplicaciones |
| NA | NA | IDE |
| 1.9.1 | Git | Control de versiones |

Diagramas de clases

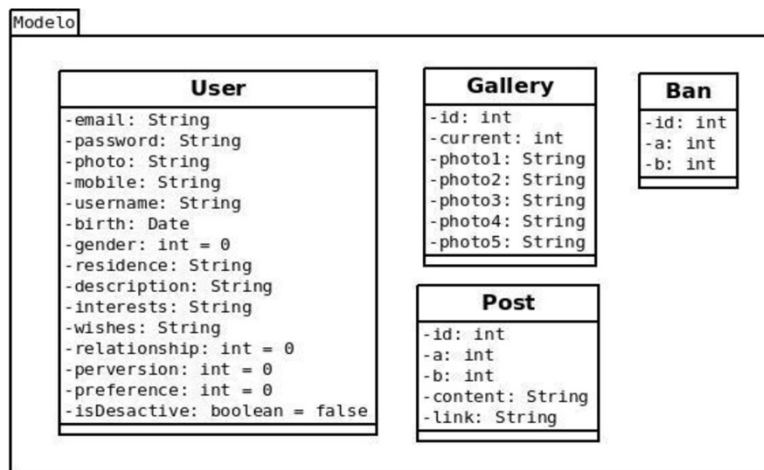


Diagrama de clases (Modelo) (Alberto, G. 2015).

3

Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

Ejemplo de *Diseño de Software* 4/11 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración

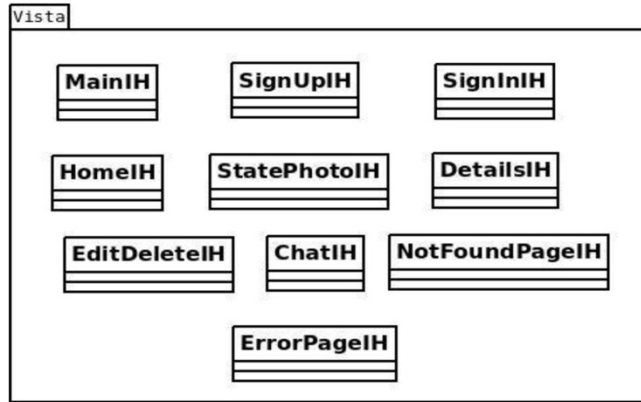


Diagrama de clases (Vista) (Alberto, G. 2015).

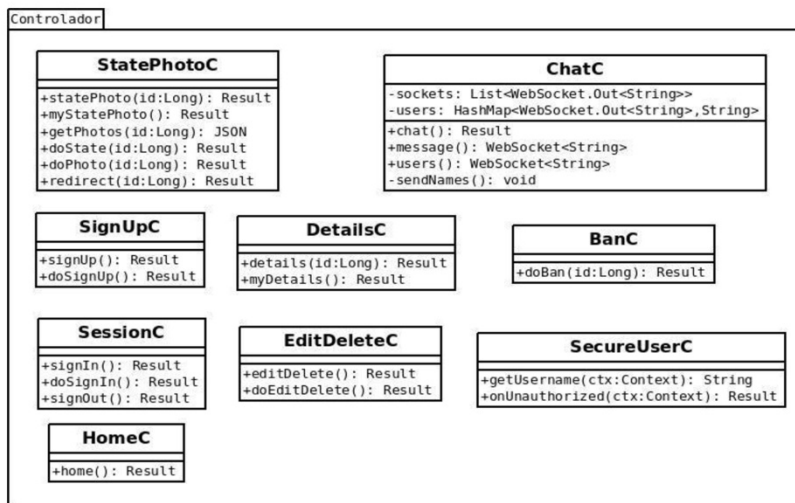


Diagrama de clases (Controlador) (Alberto, G. 2015).

4

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 02/03/2015
 Versión: 1.0

Ejemplo de *Diseño de Software* 5/11 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Diagrama de secuencia

Crear perfil

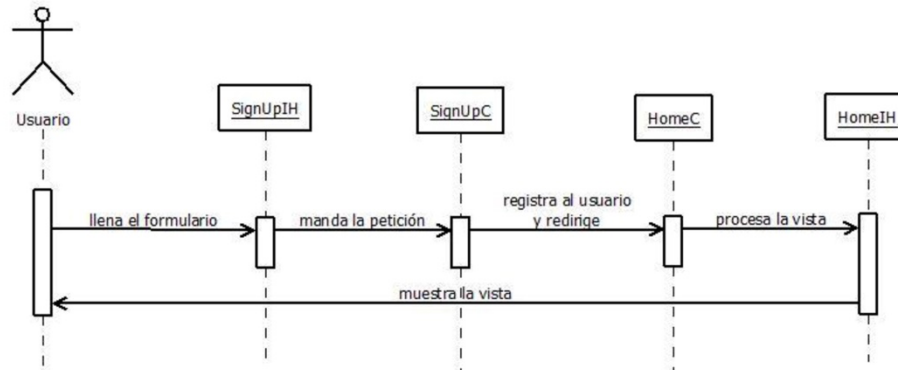


Diagrama de secuencia crear perfil (Alberto, G. 2015).

Editar perfil

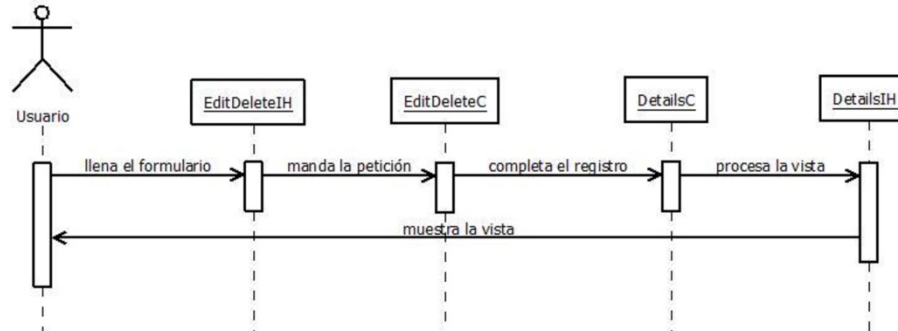


Diagrama de secuencia editar perfil (Alberto, G. 2015).

5

Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

Ejemplo de *Diseño de Software* 6/11 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Eliminar perfil

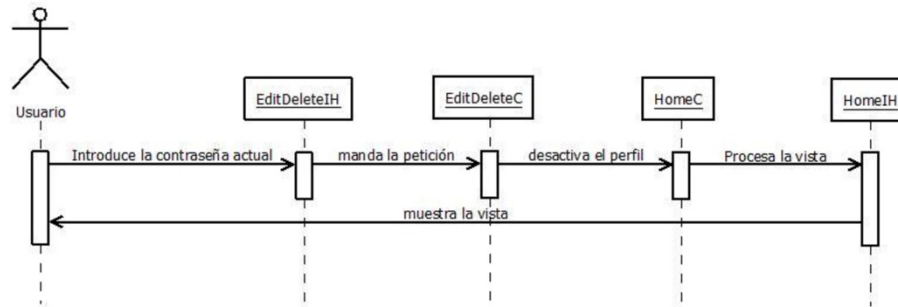


Diagrama de secuencia eliminar perfil (Alberto, G. 2015).

Iniciar sesión

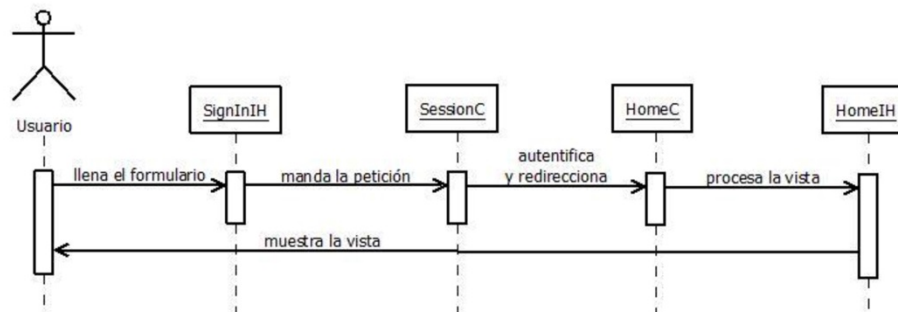


Diagrama de secuencia iniciar sesión (Alberto, G. 2015).

6

Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

Ejemplo de *Diseño de Software 7/11* (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Cerrar sesión

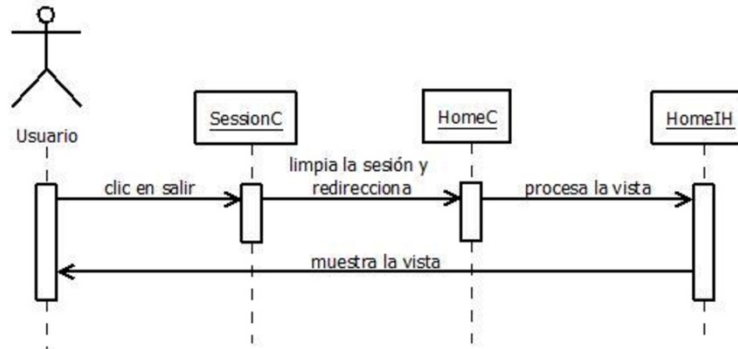


Diagrama de secuencia cerrar sesión (Alberto, G. 2015).

Publicar estados con hipervínculo

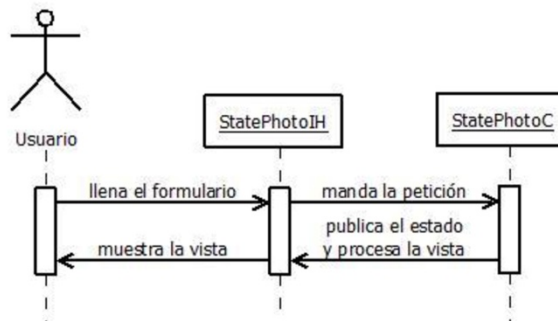


Diagrama de secuencia publicar estados con hipervínculo (Alberto, G. 2015).

7

Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

Ejemplo de *Diseño de Software* 8/11 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Subir hasta 5 fotos en una galería

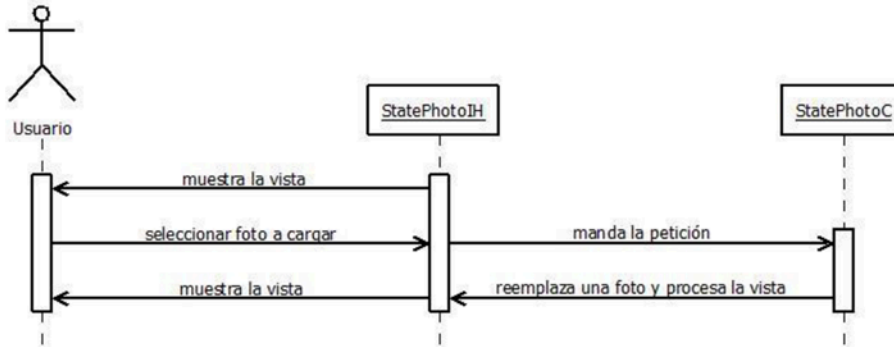


Diagrama de secuencia subir hasta cinco fotos en una galería (Alberto, G. 2015).

Chatear con otros usuarios

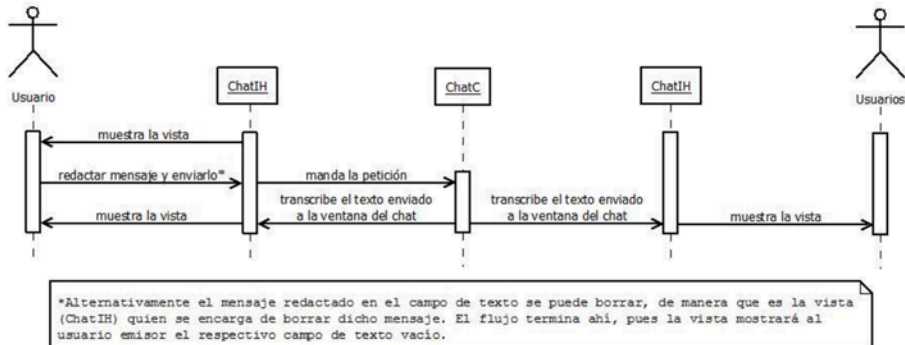


Diagrama de secuencia chatear con otros usuarios (Alberto, G. 2015).

8

Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

Ejemplo de *Diseño de Software 9/11* (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Bloquear perfiles

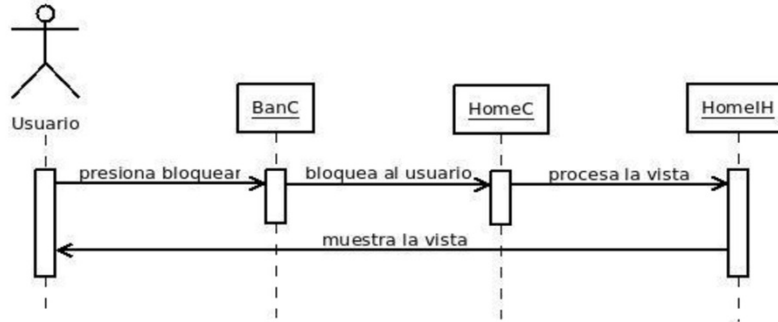
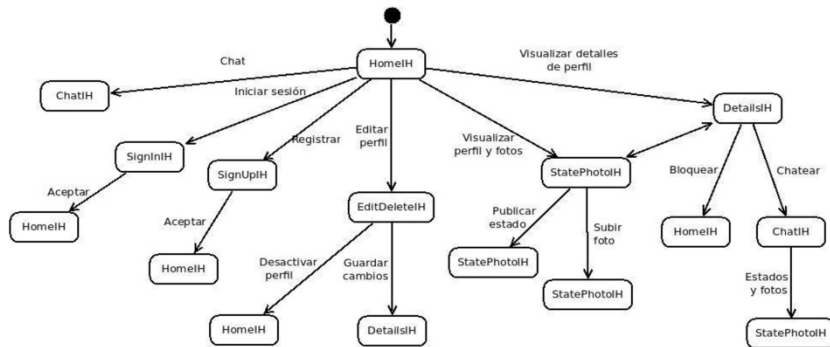


Diagrama de secuencia bloquear perfiles (Alberto, G. 2015).

Diagramas de navegación



Desde cualquier página se puede regresar a la página principal, si ha llegado a un nodo repetido reemplacelo por el nodo antecesor idéntico y sus descendientes, esto es con el fin de tener todas las posibilidades y diagramar un árbol finito.

Diagrama de navegación (Alberto, G. 2015).

9

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 02/03/2015
 Versión: 1.0

Ejemplo de *Diseño de Software* 10/11 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Diagrama entidad-relación de la base de datos

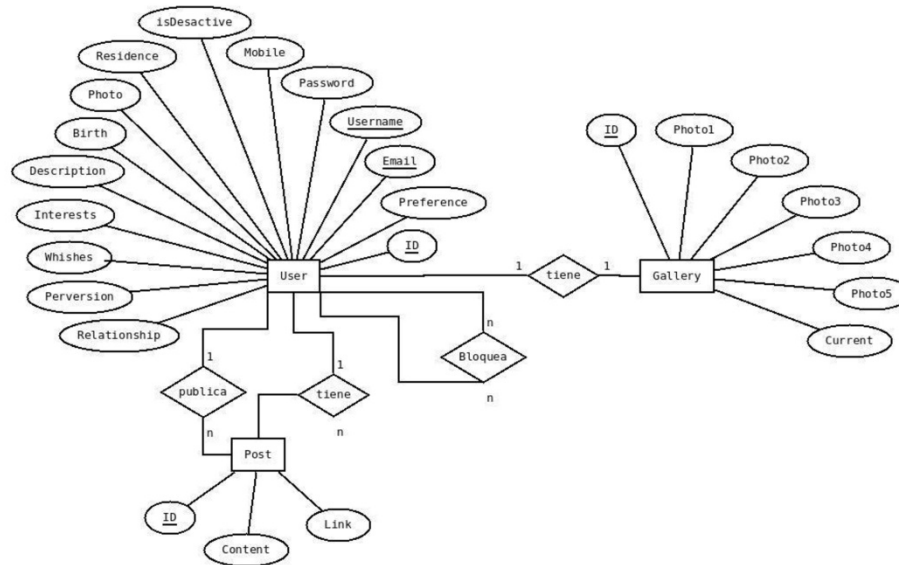


Diagrama entidad-relación de la base de datos (Alberto, G. 2015).

Elaboró: Giovanni Alberto García
Fecha de elaboración: 02/03/2015
Versión: 1.0

10

Ejemplo de *Diseño de Software* 11/11 (Alberto, G. 2015).



Primera iteración
Construcción de Software
1.0
09/03/2015
Giovanni Alberto García

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Contenido

Referencia al documento de *Especificación de Requerimientos de Software*.
Referencia al documento de *Diseño de Software*
Especificación del ambiente de desarrollo
Estándar de documentación y codificación en java
Lista de casos de uso construidos
Registro de defectos encontrados
Repositorio de código

1
Elaboró: Giovanni Alberto García
Fecha de elaboración: 09/03/2015
Versión: 1.0

Ejemplo de *Construcción de Software 2/6* (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Construcción de Software

Referencia al documento de *Especificación de Requerimientos de Software*

[Requerimientos, versión 1.0.](#)

Referencia al documento de *Diseño de Software*

[Diseño, versión 1.0.](#)

Especificación del ambiente de desarrollo

| Concepto | Herramienta | Versión |
|-----------------------------|-------------|---------|
| Framework | Play | 2.3.7 |
| Diagramador UML | Dia | 0.97.2 |
| Manejador de bases de datos | MySQL | 5.7 |
| Servidor de aplicaciones | Heroku | NA |
| IDE | NA | NA |
| Control de versiones | Git | 1.9.1 |

2

Elaboró: Giovanni Alberto García
Fecha de elaboración: 09/03/2015
Versión: 1.0

Ejemplo de *Construcción de Software* 3/6 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Estándar de documentación y codificación en java

[Lista de documentos con estándares para codificación con Java](#)

Lista de casos de uso construidos

Crear perfil

Modelo User 1.0
Vista SignUpIH 1.0, HomeIH 1.0
Controlador SignUpC 1.0, HomeC 1.0

Editar perfil

Modelo User 1.0
Vista EditDeleteIH 1.0, DetailsIH 1.0
Controlador EditDeleteC 1.0, DetailsC 1.0

Eliminar perfil

Modelo User 1.0
Vista EditDeleteIH 1.0, HomeIH 1.0
Controlador EditDeleteC 1.0, HomeC 1.0

Iniciar sesión

Modelo User 1.0
Vista SignInIH 1.0, HomeIH 1.0
Controlador SessionC 1.0, HomeC 1.0

Cerrar sesión

Modelo NA
Vista HomeIH 1.0
Controlador SessionC 1.0, HomeC 1.0

Publicar estados con hipervínculo

Modelo Post 1.0
Vista StatePhotoIH 1.0
Controlador StatePhotoC 1.0

3

Elaboró: Giovanni Alberto García
Fecha de elaboración: 09/03/2015
Versión: 1.0

Ejemplo de *Construcción de Software* 4/6 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Subir hasta 5 fotos en una galería

Modelo Gallery 1.0
Vista StatePhotoH 1.0
Controlador StatePhotoC 1.0

Chatear con otros usuarios

Modelo NA
Vista ChatH 1.0
Controlador ChatC 1.0

Bloquear perfiles

Modelo Ban 1.0
Vista DetailsIH 1.0, HomeIH 1.0
Controlador DetailsC 1.0, BanC 1.0, HomeC 1.0

Registro de defectos encontrados

| Caso de uso | Número de casos de prueba | Defectos encontrados | Defectos corregidos |
|---------------------------------------|---------------------------|----------------------|---------------------|
| Crear Perfil | 3 | 12 | 12 |
| Editar perfil | 3 | 10 | 10 |
| Eliminar perfil | 2 | 4 | 4 |
| Iniciar sesión | 3 | 3 | 3 |
| Cerrar sesión | 2 | 5 | 5 |
| Publicar estados con hipervínculo | 3 | 15 | 15 |
| Publicar hasta 5 fotos en una galería | 3 | 4 | 4 |
| Chatear con otros | 3 | 2 | 2 |

4

Elaboró: Giovanni Alberto García
Fecha de elaboración: 09/03/2015
Versión: 1.0

Ejemplo de *Construcción de Software* 5/6 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



| Caso de uso | Número de casos de prueba | Defectos encontrados | Defectos corregidos |
|-------------------|---------------------------|----------------------|---------------------|
| usuarios | | | |
| Bloquear perfiles | 2 | 6 | 6 |
| Total | 24 | 61 | 61 |

Calidad de software = $61/24 = 2.54$

Repositorio de código

Visualiza el código en Github (<https://github.com/yovasx2/media-naranja>).

5
Elaboró: Giovanni Alberto García
Fecha de elaboración: 09/03/2015
Versión: 1.0

Ejemplo de *Construcción de Software* 6/6 (Alberto, G. 2015).



Primera iteración
Integración y pruebas de Software
1.0
16/03/2015
Giovanni Alberto García

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Contenido

Referencia al documento de *Especificación de Requerimientos* de Software
Referencia al documento de *Diseño de Software*
Referencia al documento de *Construcción de Software*
Especificación de la estrategia de integración
Configuración del software
Registro de defectos encontrados
Tabla de casos de prueba aplicados

1
Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 2/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Integración y pruebas de Software

Referencia al documento de *Especificación de Requerimientos de Software*

[Requerimientos, versión 1.0](#)

Referencia al documento de *Diseño de Software*

[Diseño, versión 1.0](#)

Referencia al documento de *Construcción de Software*

[Construcción, versión 1.0](#)

Especificación de la estrategia de integración

La estrategia de integración que se seguirá será adicionar uno a uno los casos de uso desarrollados, de manera que el sistema sea visiblemente funcional en todo momento de la integración, ya que de otra manera, el sistema sería funcional pero no tendría manera de ser comprobado por el usuario. De igual manera se harán pruebas de código para ir verificando la correcta funcionalidad de los casos de uso ya integrados.

Configuración del software

Crear perfil

Modelo User 1.0

2

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 3/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Vista SignUpIH 1.0, HomeIH 1.0
Controlador SignUpC 1.0, HomeC 1.0

Editar perfil

Modelo User 1.0
Vista EditDeleteIH 1.0, DetailsIH 1.0
Controlador EditDeleteC 1.0, DetailsC 1.0

Eliminar perfil

Modelo User 1.0
Vista EditDeleteIH 1.0, HomeIH 1.0
Controlador EditDeleteC 1.0, HomeC 1.0

Iniciar sesión

Modelo User 1.0
Vista SignInIH 1.0, HomeIH 1.0
Controlador SessionC 1.0, HomeC 1.0

Cerrar sesión

Modelo NA
Vista HomeIH 1.0
Controlador SessionC 1.0, HomeC 1.0

Publicar estados con hipervínculo

Modelo Post 1.0
Vista StatePhotoH 1.0
Controlador StatePhotoC 1.0

Subir hasta 5 fotos en una galería

Modelo Gallery 1.0
Vista StatePhotoH 1.0
Controlador StatePhotoC 1.0

Chatear con otros usuarios

Modelo NA
Vista ChatIH 1.0
Controlador ChatC 1.0

3

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Bloquear perfiles

Modelo Ban 1.0
Vista DetailsIH 1.0, HomeIH 1.0
Controlador DetailsC 1.0, BanC 1.0, HomeC 1.0

Registro de defectos encontrados

| Prueba del caso de uso | Responsable de la prueba | Identificador del caso de prueba | Defecto identificado |
|----------------------------|--------------------------|----------------------------------|---|
| Crear perfil | Giovanni Alberto García | N-1 | No abría la página principal tras el registro |
| Editar perfil | Giovanni Alberto García | A-1 | Permitía subir cualquier archivo que no fuera imagen |
| Eliminar perfil | Giovanni Alberto García | N-1 | El usuario seguía accediendo a su perfil y detalles |
| Subir fotos | Giovanni Alberto García | N-1 | Sólo se actualizaba una foto de las cinco |
| Publicar estado | Giovanni Alberto García | N-1 | No servía el hipervínculo |
| Chatear con otros usuarios | Giovanni Alberto García | N-1 | No aparecía la lista de usuarios conectados |
| Bloquear perfiles | Giovanni Alberto García | N-1 | Se bloqueaba el perfil, pero podían acceder todavía a sus perfiles y detalles |
| Total | 7 | | |

4

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

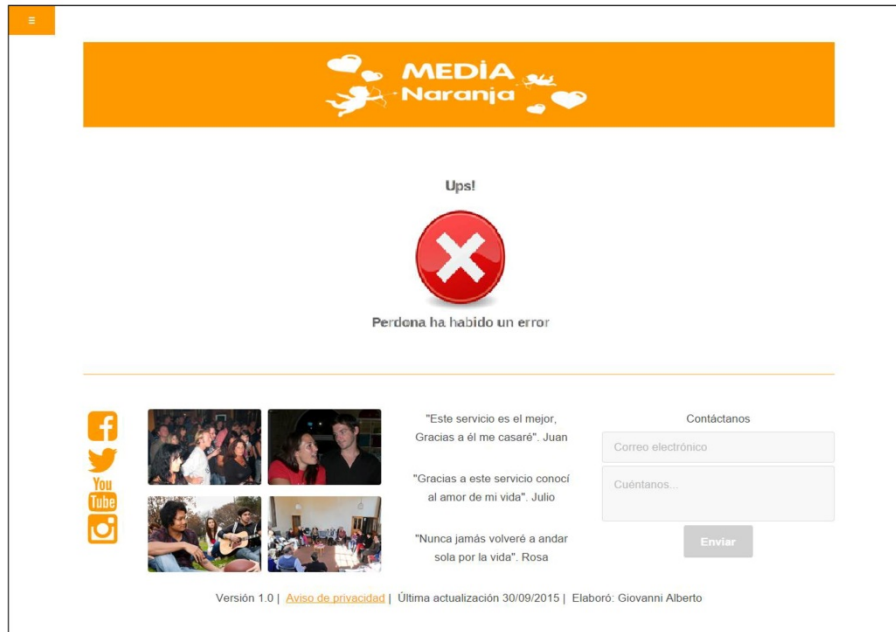
Ejemplo de *Integración y pruebas de Software* 5/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Tabla de casos de prueba aplicados

Todos los casos excepcionales por pérdida de conexión con la base de datos muestran la siguiente pantalla:



Captura de pantalla errores 500's en el servidor (Alberto, G. 2015).

5

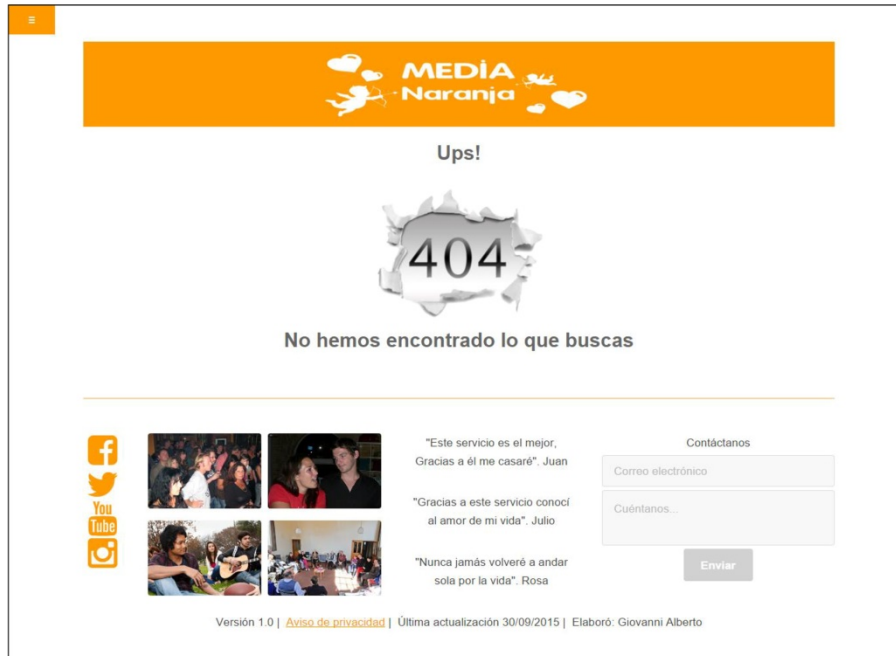
Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 6/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Y toda página no encontrada muestra la siguiente pantalla:



Captura de pantalla error 404 en el servidor (Alberto, G. 2015).

6

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software 7/35* (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Crear perfil

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|------------------------------|---|
| Esperar un registro exitoso. | Introducir el correo, contraseña, confirmación y fecha de nacimiento siguientes: test@test.com password password 12/03/1989 |

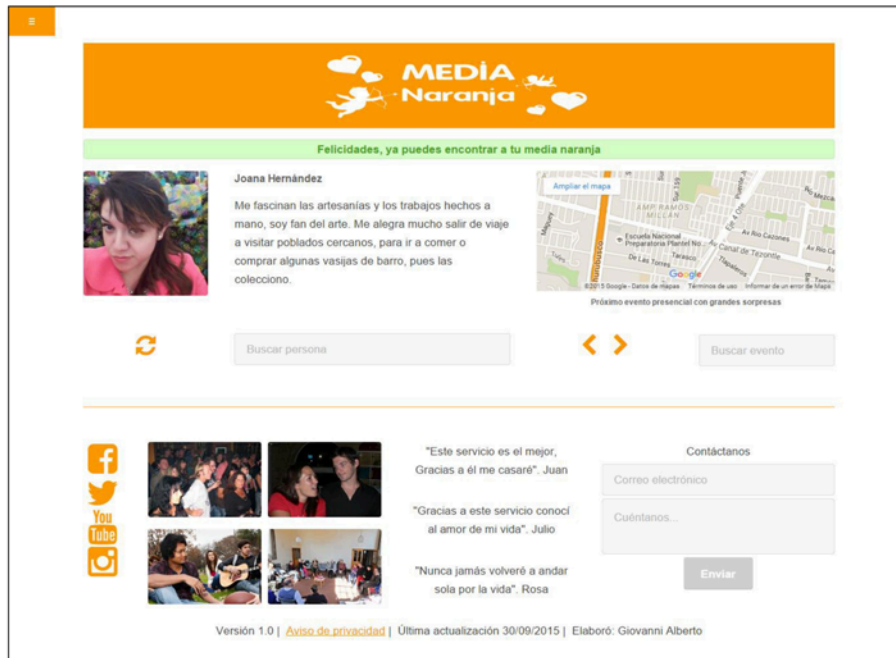
Captura de pantalla crear perfil (flujo normal) 1/2 (Alberto, G. 2015).

7

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 8/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla crear perfil (flujo normal) 2/2 (Alberto, G. 2015).

Caso de prueba 2 (flujo alternativo)

| Resultado esperado | Entradas |
|--|--|
| Esperar una advertencia sobre el correo no válido y una contraseña muy corta | Introducir el correo, contraseña y confirmación siguientes: testtest.com pas pa 12/03/1989 |

8

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de Integración y pruebas de Software 9/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla crear perfil (flujo alternativo) 1/2 (Alberto, G. 2015).

9

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 10/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla crear perfil (flujo alternativo) 2/2 (Alberto, G. 2015).

10

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 11/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Editar perfil

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|---|--|
| El sistema editará exitosamente los datos del registro y llevará al usuario a la página de detalles | Ingresar datos válidos en todos los campos solicitados a excepción de la contraseña y la confirmación que deberán estar vacías (para no cambiar la contraseña); recuerda seleccionar una foto de perfil (JPG). |

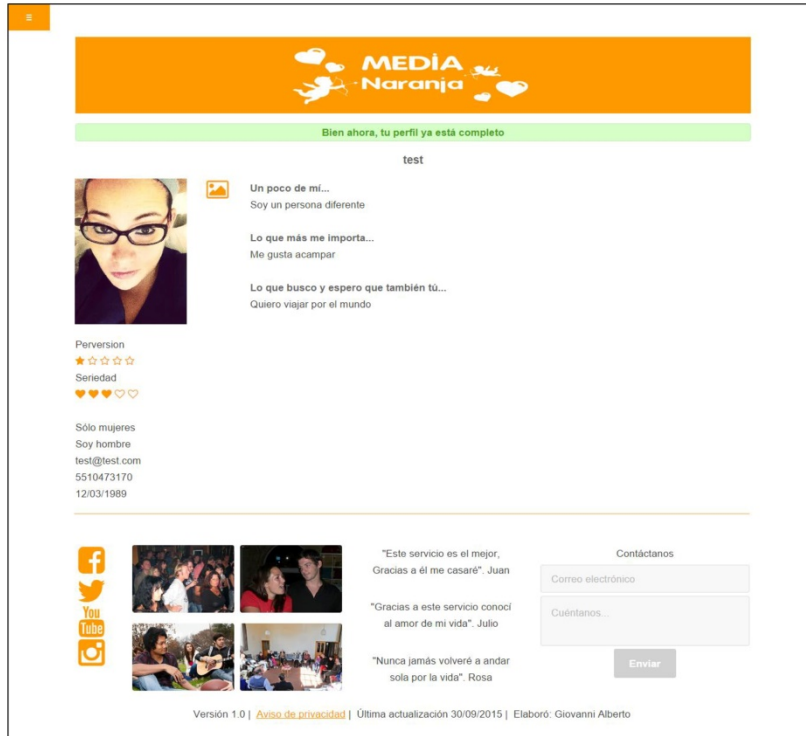
Captura de pantalla editar perfil (flujo normal) 1/2 (Alberto, G. 2015).

11

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 12/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla editar perfil (flujo normal) 2/2 (Alberto, G. 2015).

Caso de prueba 2 (flujo alternativo)

| Resultado esperado | Entradas |
|--|--|
| El sistema indicará dónde están los errores para que el usuario los corrija de inmediato | El usuario ingresará datos erróneos como contraseñas sin coincidir o valores mal construidos o no correspondientes al tipo que se pide (letras donde se requieran números) |

12

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de Integración y pruebas de Software 13/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Logo: MEDIA Naranja

Datos de acceso *

test@test.com

....

**

Desactivar perfil

Datos de contacto *

test

México DF

Gustos, intereses y deseos *

Soy un persona diferente

Me gusta acampar

Quiero viajar por el mundo

Datos de la cuenta *

test

Seleccionar archivo No se eligió archivo

Preferencias *

Busco sólo mujeres

Quiero pasarla bien

Soy demasiado cursi

Hombre Mujer

Limpiar Guardar

Contáctanos

Correo electrónico

Cuéntanos...

Enviar

Versión 1.0 | [Aviso de privacidad](#) | Última actualización 30/09/2015 | Elaboró: Giovanni Alberto

Captura de pantalla editar perfil (flujo alternativo) 1/2 (Alberto, G. 2015).

13

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 14/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



MEDIA Naranja

Datos de acceso

test@test.com

Contraseña nueva

Introduce al menos 8 caracteres

Confirmación

Las contraseñas no coinciden

Desactivar perfil

Datos de contacto *

2342432432

México DF

Preferencias *

Busco sólo mujeres

Quiero pasarla bien

Soy demasiado cursi

Hombre Mujer

Gustos, intereses y deseos *

Soy un persona diferente

Me gusta acampar

Quiero viajar por el mundo

Limpiar Guardar

Contáctanos

Correo electrónico

Cuéntanos...

Enviar

Este servicio es el mejor, Gracias a él me casaré". Juan

"Gracias a este servicio conocí al amor de mi vida". Julio

"Nunca jamás volveré a andar sola por la vida". Rosa

Versión 1.0 | [Aviso de privacidad](#) | Última actualización 30/09/2015 | Elaboró: Giovanni Alberto

Captura de pantalla editar perfil (flujo alternativo) 2/2 (Alberto, G. 2015).

14

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 15/35 (Alberto, G. 2015).

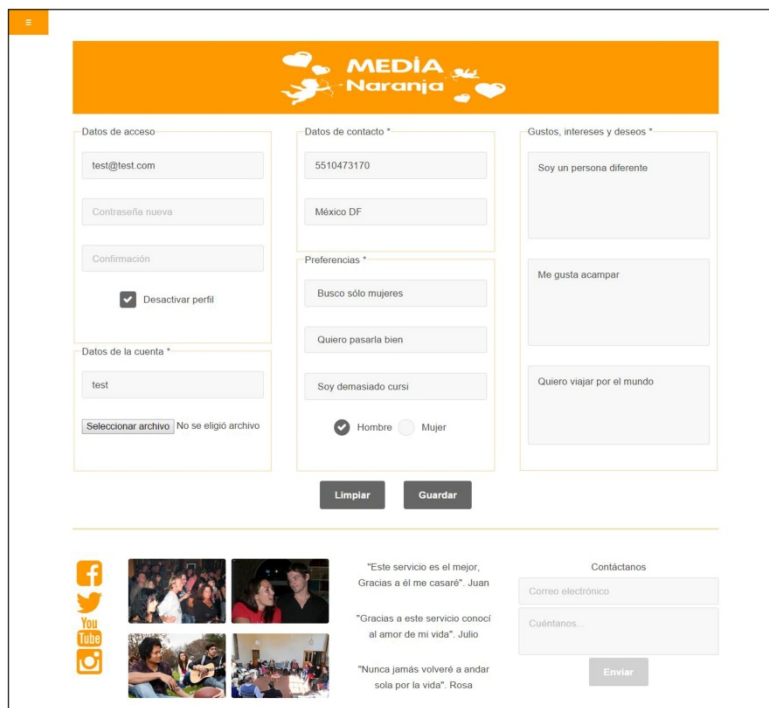
Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Eliminar perfil

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|---|--|
| Esperar que el sistema abra la página principal indicando que la acción deseada fue exitosa | Marcar la casilla desactivar perfil y presionar el botón guardar |



Captura de pantalla eliminar perfil (flujo normal) 1/2 (Alberto, G. 2015).

15

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 16/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla eliminar perfil (flujo normal) 2/2 (Alberto, G. 2015).

Iniciar sesión

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|---------------------------|--|
| Esperar un acceso exitoso | Introducir el correo y contraseña siguientes: test@test.com password |

16

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de Integración y pruebas de Software 17/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla iniciar sesión (flujo normal) 1/2 (Alberto, G. 2015).

17

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 18/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla iniciar sesión (flujo normal) 2/2 (Alberto, G. 2015).

Caso de prueba 2 (flujo alternativo)

| Resultado esperado | Entradas |
|--|---|
| Esperar una advertencia de correo y/o contraseña inválidos | Introducir el correo y contraseña siguientes: testtest.com pass |

18

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de Integración y pruebas de Software 19/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla iniciar sesión (flujo alternativo) 1/2 (Alberto, G. 2015).

19

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 20/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla iniciar sesión (flujo alternativo) 2/2 (Alberto, G. 2015).

Cerrar sesión

Caso de prueba 1 (flujo normal)

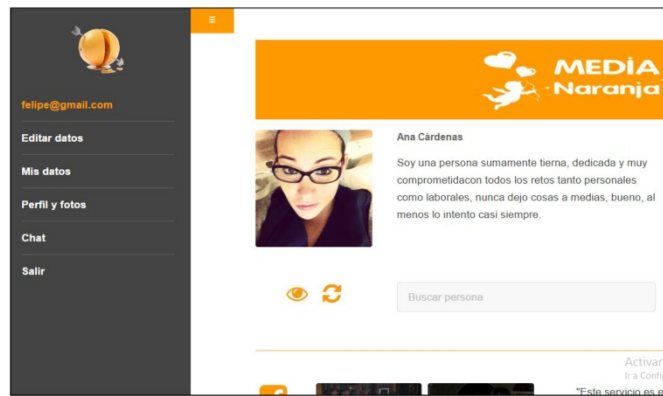
| Resultado esperado | Entradas |
|--|---------------------------------------|
| Ser redirigido a la página principal y no tener privilegios de usuario | Hacer clic en el hipervínculo "Salir" |

20

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de Integración y pruebas de Software 21/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla cerrar sesión (flujo normal) 1/2 (Alberto, G. 2015).



Captura de pantalla cerrar sesión (flujo normal) 2/2 (Alberto, G. 2015).

21

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de Integración y pruebas de Software 22/35 (Alberto, G. 2015).

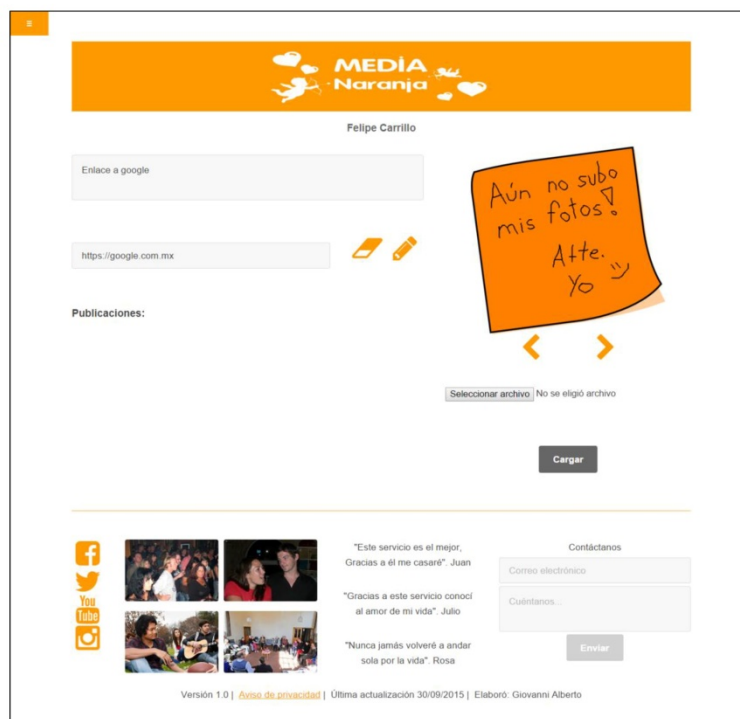
Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Publicar estados con hipervínculo

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|--|--|
| Esperar una publicación de un enlace a google en el perfil correspondiente | Introducir el texto: "Enlace a google" y en el hipervínculo: "https://google.com.mx" |



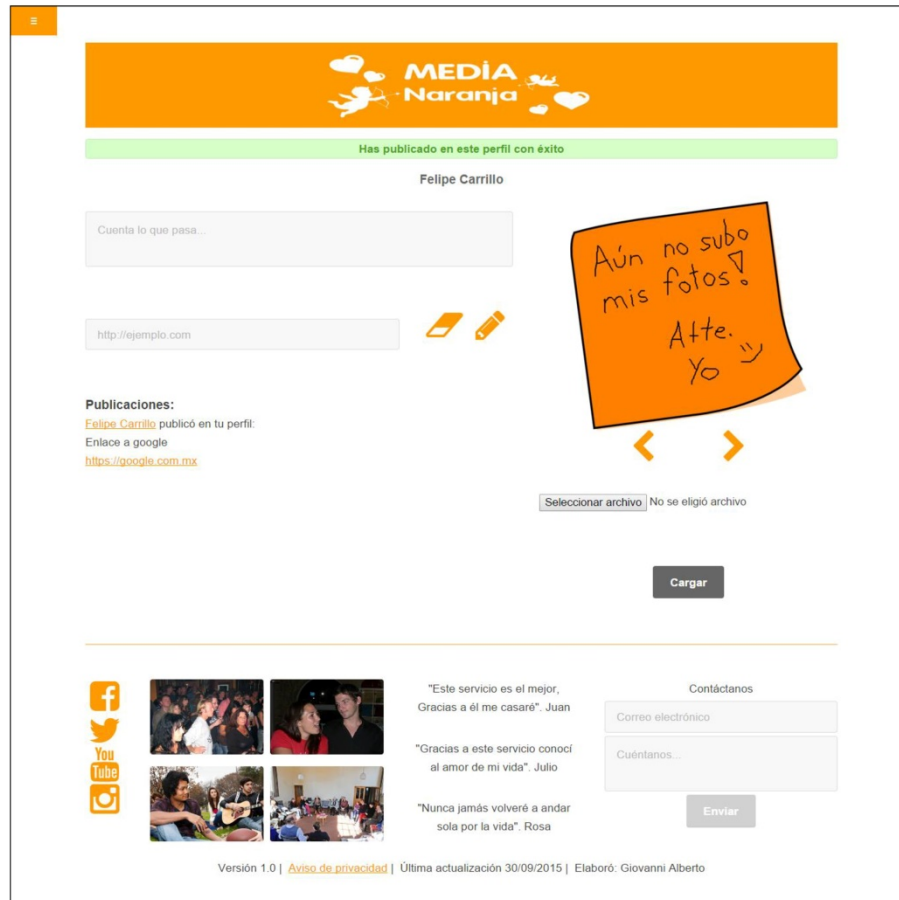
Captura de pantalla publicar estados con hipervínculo (flujo normal) 1/2 (Alberto, G. 2015).

22

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 23/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla publicar estados con hipervínculo (flujo normal) 2/2 (Alberto, G. 2015).

23
Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

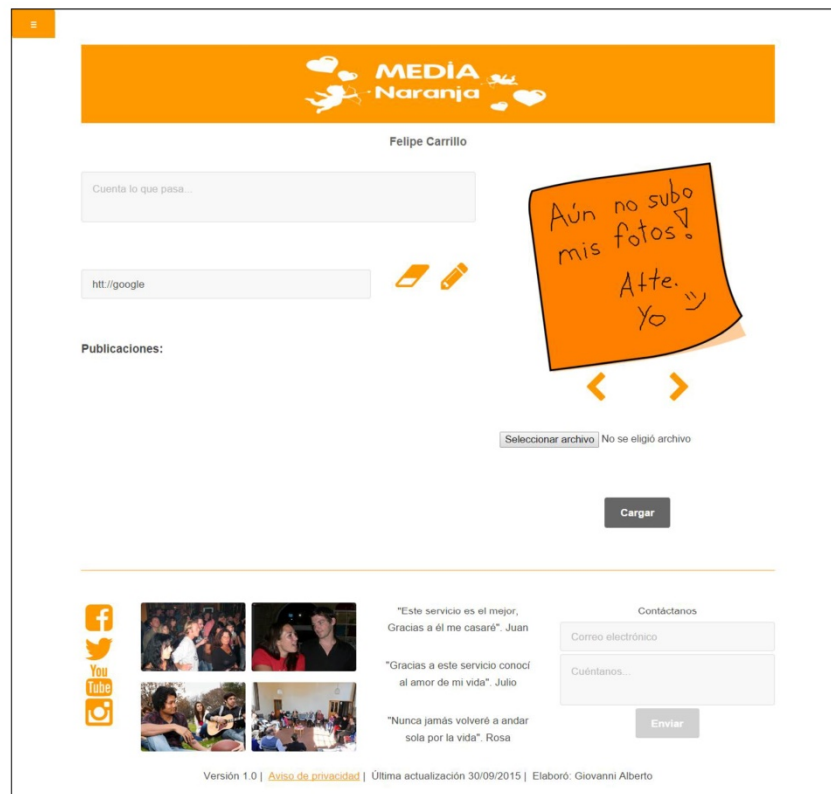
Ejemplo de *Integración y pruebas de Software* 24/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Caso de prueba 2 (flujo alternativo)

| Resultado esperado | Entradas |
|---|---|
| Esperar una advertencia de texto dejado en blanco e hipervínculo inválido | Ningún texto e hipervínculo: "htt://google" |



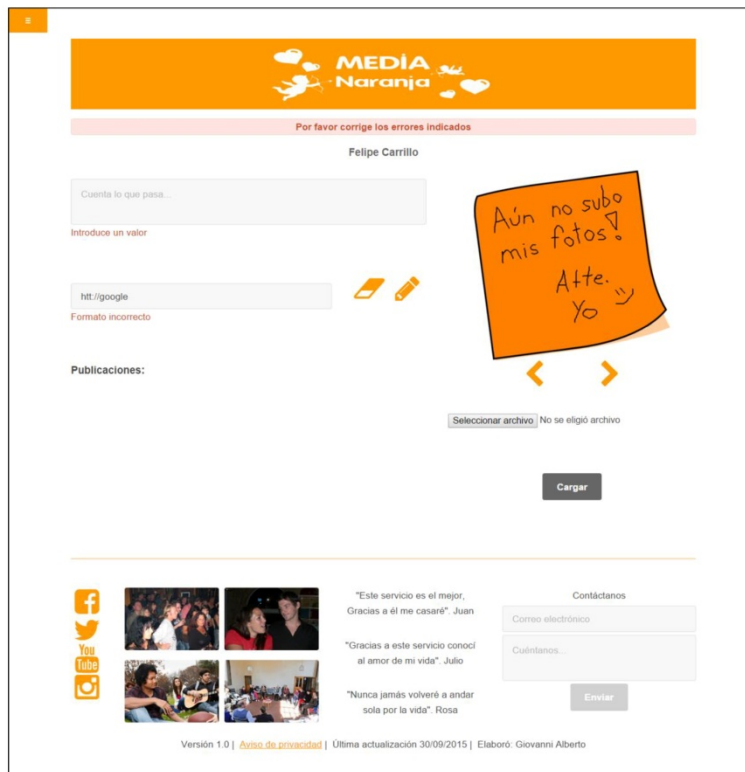
Captura de pantalla publicar estados con hipervínculo (flujo alternativo) 1/2 (Alberto, G. 2015).

24

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 25/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla publicar estados con hipervínculo (flujo alternativo) 2/2 (Alberto, G. 2015).

Subir hasta 5 fotos en una galería

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|---|--|
| Cargar la foto e insertarla en la galería | Seleccionar un archivo con formato jpg |

25

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de Integración y pruebas de Software 26/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



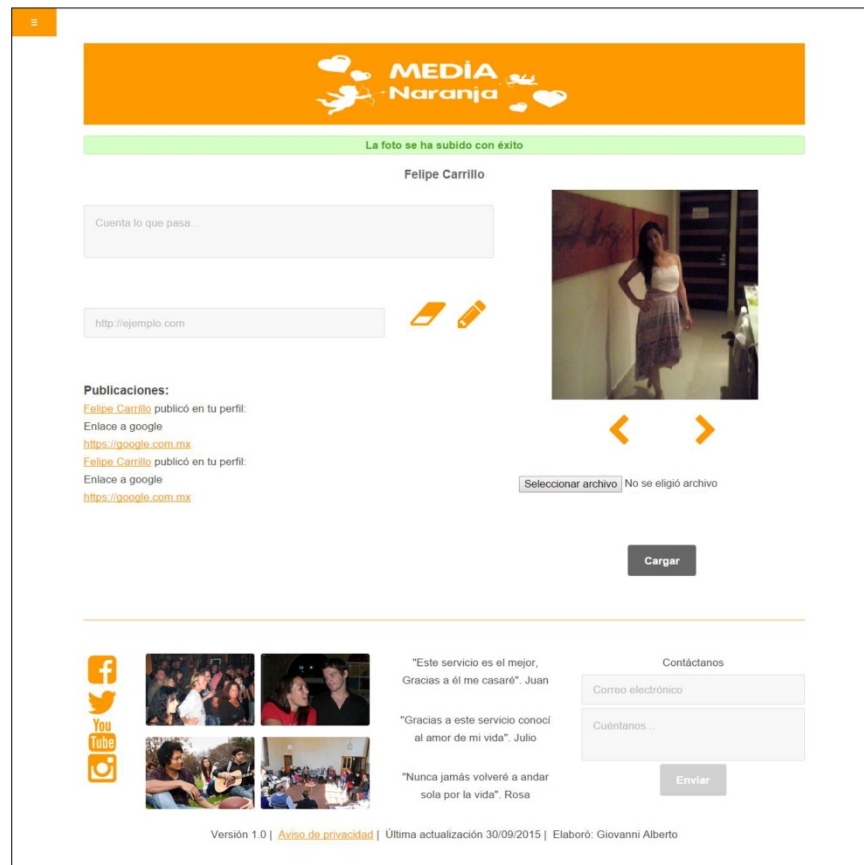
Captura de pantalla subir hasta 5 fotos en una galería (flujo normal) 1/2 (Alberto, G. 2015).

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

26

Ejemplo de Integración y pruebas de Software 27/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla subir hasta 5 fotos en una galería (flujo normal) 2/2 (Alberto, G. 2015).

27
Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

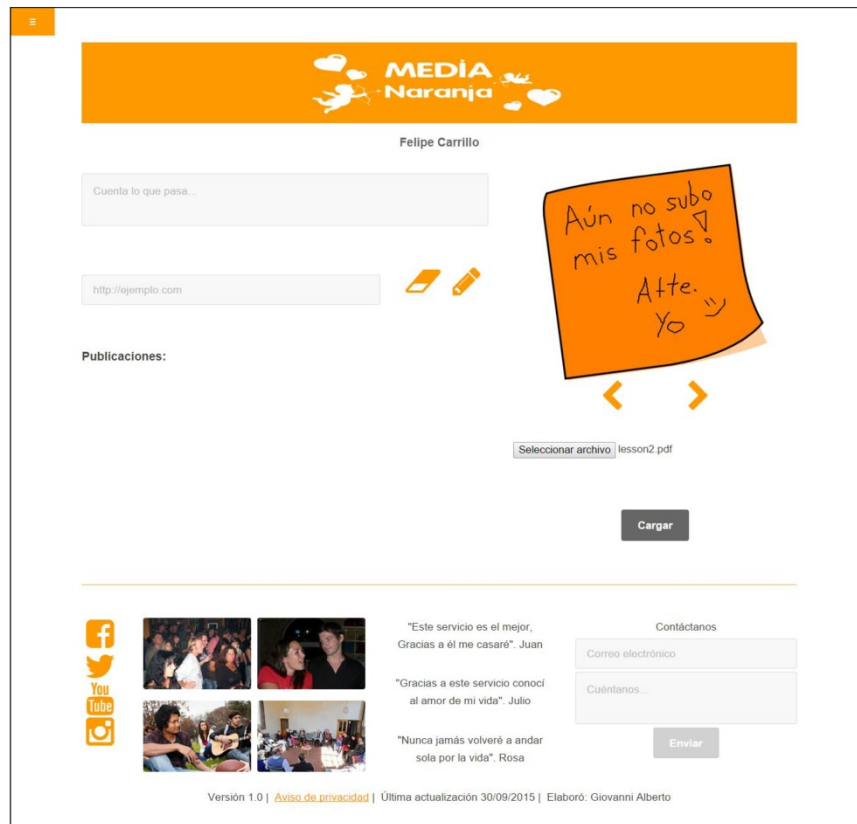
Ejemplo de *Integración y pruebas de Software* 28/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Caso de prueba 2 (flujo alternativo)

| Resultado esperado | Entradas |
|--------------------------------------|--|
| Esperar un aviso de formato inválido | Seleccionar un archivo con formato pdf |



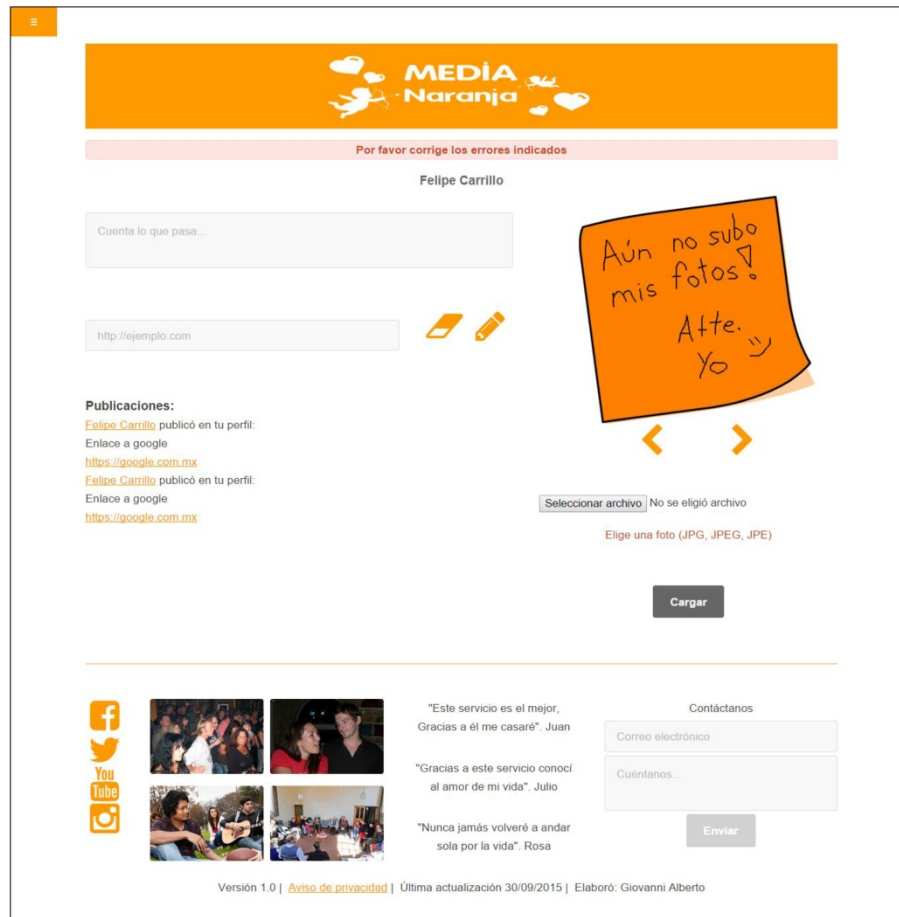
Captura de pantalla subir hasta 5 fotos en una galería (flujo alternativo) 1/2 (Alberto, G. 2015).

28

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de Integración y pruebas de Software 29/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla subir hasta 5 fotos en una galería (flujo alternativo) 2/2 (Alberto, G. 2015).

29
Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de Integración y pruebas de Software 30/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Chatear con otros usuarios

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|--|--|
| Esperar a que el otro usuario reciba el mensaje y responda | Introducir el texto: "Hola" y presionar enviar |



Captura de pantalla chatear con otros usuarios (flujo normal) 1/2 (Alberto, G. 2015).

30

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 31/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla chatear con otros usuarios (flujo normal) 2/2 (Alberto, G. 2015).

Caso de prueba 2 (flujo alternativo)

| Resultado esperado | Entradas |
|--|-------------------------|
| Enviar una advertencia de mensaje inválido | Enviar un mensaje vacío |

31

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 32/35 (Alberto, G. 2015).

Equipo: Diablo Systems
 Sistema: Media Naranja
 Iteración: Primera iteración



Captura de pantalla chatear con otros usuarios (flujo alternativo) 1/1 (Alberto, G. 2015).

Bloquear perfiles

Caso de prueba 1 (flujo normal)

| Resultado esperado | Entradas |
|---|---------------------------------------|
| Esperar a que se abra la página principal | Seleccionar algún perfil, bloquearlo. |

32

Elaboró: Giovanni Alberto García
 Fecha de elaboración: 16/03/2015
 Versión: 1.0

Ejemplo de *Integración y pruebas de Software* 33/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla bloquear perfiles (flujo normal) 1/2 (Alberto, G. 2015).

33

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de Integración y pruebas de Software 34/35 (Alberto, G. 2015).

Equipo: Diablo Systems
Sistema: Media Naranja
Iteración: Primera iteración



Captura de pantalla bloquear perfiles (flujo normal) 2/2 (Alberto, G. 2015).

34

Elaboró: Giovanni Alberto García
Fecha de elaboración: 16/03/2015
Versión: 1.0

Ejemplo de Integración y pruebas de Software 35/35 (Alberto, G. 2015).

Bibliografía

- SWEBOK 3.0 Guide to the Software Engineering Body of Knowledge*. (2014). IEEE Computer society.
- Ibargüengoitia, G., Oktaba, H. (s.f.). *Primer Curso de Ingeniería de Software*. DF, México: En publicación.
- Morales M., O. H. (2012). *KUALI-BEH Software Project Common Concepts*. Object Management Group. Retrieved from <http://www.kuali-kaans.mx/proyectos/kuali-beh>
- Pressman, R. (2010). *Software Engineering: A Practitioner's Approach* (7° ed.). New York, USA: Mc Graw Hill.
- Sommerville, I. (2011). *Software Engineering* (9° ed.). Boston, Massachusetts, USA: Addison Wesley.