



---

# CLASS CONTROL

---

## PLAN DE DESARROLLO DE *SOFTWARE*, VERSIÓN 1.2

Desarrollo de una aplicación para la plataforma social educativa *edmodo*.

---

TFG1314-40

---



21 DE FEBRERO DE 2014

ALUMNO: CRISTIAN TEJEDOR GARCÍA

TUTORA: YANIA CRESPO GONZÁLEZ-CARVAJAL



Escuela Técnica Superior  
de **Ingeniería Informática**

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

## Historial de Revisiones

Fecha	Versión	Descripción	Autor
24/12/2013	0.9	Versión preliminar del Plan de Desarrollo de <i>Software</i> .	Cristian T.G.
01/01/2014	1.0	Versión propuesta para aprobación al final de la fase de Inicio.	Cristian T.G.
25/01/2014	1.1	Revisión tras la reunión con la tutora del proyecto.	Cristian T.G.
21/02/2014	1.2	Revisión del diseño del documento.	Cristian T.G.

**TABLA 1: HISTORIAL DE REVISIONES**

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

## Contenido

Índice de tablas .....	4
Índice de ilustraciones .....	4
<b>1. Introducción</b> .....	5
1.1. Propósito .....	5
1.2. Ámbito .....	5
1.3. Definiciones, acrónimos y abreviaturas .....	5
1.4. Referencias .....	5
1.5. Visión global .....	6
<b>2. Visión general del proyecto</b> .....	7
2.1. Objetivos y ámbito del proyecto .....	7
2.2. Suposiciones y restricciones .....	7
2.3. Entregables .....	7
2.4. Evolución del plan de Desarrollo de <i>Software</i> .....	8
<b>3. Organización del proyecto</b> .....	9
3.1. Estructura de la organización .....	9
3.2. Interfaces externas .....	9
3.3. Roles y responsabilidades .....	9
3.3.1. Responsabilidades del proyecto .....	9
<b>4. Proceso de gestión</b> .....	10
4.1. Estimaciones del proyecto .....	10
4.2. Plan del proyecto .....	10
4.2.1. Plan de fases e iteraciones .....	10
4.2.2. Entregas .....	11
4.2.3. Calendarización del proyecto .....	12
4.2.4. Recursos humanos del proyecto .....	12
4.3. Seguimiento y control del proyecto .....	12
4.3.1. Gestión de requisitos .....	12
4.3.2. Control de calidad .....	12
4.3.3. Informes y medidas .....	13
4.3.4. Gestión de riesgos .....	13
4.3.5. Gestión de Configuraciones .....	13
<b>5. Anexo</b> .....	14
5.1. Documento de Solicitud de Cambio .....	14

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

## Índice de tablas

Tabla 1: Historial de revisiones .....	2
Tabla 2: Definición de roles.....	9
Tabla 3: Responsabilidades del proyecto.....	9
Tabla 4: Fases y duración estimada .....	11
Tabla 5: Entregas .....	12

## Índice de ilustraciones

Ilustración 1: Roles durante el proyecto.....	9
Ilustración 2: Fases Del Desarrollo en OpenUP .....	12
Ilustración 3: Ficha de Solicitud De Cambio.....	14

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

## 1. Introducción

Este Plan de Desarrollo de *Software* es un documento a incluir en la propuesta elaborada como respuesta a la asignatura *Trabajo Fin de Grado* del Grado en Ingeniería Informática de Sistemas de la Escuela Técnica Superior de Ingeniería Informática<sup>[1]</sup> de la Universidad de Valladolid<sup>[2]</sup>. Este documento provee una visión global del enfoque de desarrollo propuesto.

El proyecto ha sido ofertado por la profesora Yania Crespo Carvajal, y se llevará a cabo basándose en la metodología OpenUP<sup>[3]</sup>.

El enfoque de desarrollo propuesto constituye una configuración del proceso OpenUP de acuerdo a las características del proyecto, seleccionando los roles de los participantes, las actividades a realizar y los artefactos que serán generados. Este documento de desarrollo, es a su vez uno de los artefactos de OpenUP.

### 1.1. Propósito

El propósito de este Plan de Desarrollo de *Software* es ofrecer toda la información necesaria para controlar el desarrollo del proyecto *Class Control*. Es una visión al más alto nivel de abstracción que facilita al equipo de trabajo la organización del desarrollo: el alumno y la tutora.

Los miembros del equipo son dos: el alumno, que se encargará de planificar, analizar y desarrollar la aplicación según OpenUP, y el tutor, encargado de revisar todo el desarrollo y aportar conocimiento e ideas.

### 1.2. Ámbito

Este Plan de Desarrollo de *Software* presenta el plan global que será seguido en el proyecto *Class Control*. En él se incluyen el plan de fases y se hace referencia a los planes de iteraciones, documentos que se aportan en forma separada.

### 1.3. Definiciones, acrónimos y abreviaturas

Las definiciones, acrónimos y abreviaturas utilizadas a lo largo del proyecto se incluyen en el *Documento de Glosario de Términos*.

### 1.4. Referencias

- [1] <<Escuela Técnica Superior de Ingeniería Informática>>. [En línea] Disponible en: <https://www.inf.uva.es/> [Accedido: 25-diciembre-2013]
- [2] <<Universidad de Valladolid>>. [En línea] Disponible en: <http://www.uva.es/> [Accedido: 20-mayo-2014]
- [3] <<OpenUP. From Wikipedia, the free encyclopedia>>. [En línea] Disponible en: [http://en.wikipedia.org/wiki/Unified\\_Process](http://en.wikipedia.org/wiki/Unified_Process) [Accedido: 24-diciembre-2014]
- [4] <<Bienvenido a Edmodo>>. [En línea] Disponible en: <https://www.edmodo.com/> [Accedido: 25-diciembre-2014]
- [5] <<Descarga de Java>>. [En línea] Disponible en: <http://www.java.com/es/download/> [Accedido: 25-diciembre-2014]
- [6] <<SQL. From Wikipedia, the free encyclopedia>>. [En línea] Disponible en: [http://en.wikipedia.org/wiki/Unified\\_Process](http://en.wikipedia.org/wiki/Unified_Process) [Accedido: 25-diciembre-2014]
- [7] <<Heroku, Cloud Application Platform>>. [En línea] Disponible en: <https://www.heroku.com/> [Accedido: 25-diciembre-2014]

Class Control	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

- [8] <<Pivotal Tracker>>. [En línea] Disponible en: <http://www.pivotaltracker.com/> [Accedido: 25-diciembre-2014]
- [9] <<Colegio San Agustín>>. [En línea] Disponible en: <http://www.colegiosanagustin.net/> [Accedido: 25-diciembre-2014]
- [10] <<Git>>. [En línea] Disponible en: <http://git-scm.com/> [Accedido: 26-diciembre-2014]
- [11] <<Dropbox>>. [En línea] Disponible en: <https://www.dropbox.com/> [Accedido: 27-diciembre-2014]

## 1.5. Visión global

Después de esta introducción, el resto del documento está organizado en las siguientes secciones:

- Visión general del proyecto: proporciona una descripción de los objetivos, el alcance y las restricciones del proyecto. También define los entregables que se deben ir proporcionando al cliente y la evolución del proyecto.
- Organización del proyecto: describe la estructura de la organización del equipo de trabajo, incluyendo roles y responsabilidades.
- Proceso de gestión: detalla los costes estimados y la calendarización, indicando las fases principales, sus correspondientes hitos y los artefactos a entregar. También describe el proceso de seguimiento y control del proyecto teniendo en cuenta el control de calidad así como la gestión de riesgos.

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

## 2. Visión general del proyecto

### 2.1. Objetivos y ámbito del proyecto

*Class Control* es una aplicación Web que facilitará a los usuarios de la plataforma *edmodo*<sup>[4]</sup> (profesores, alumnos y padres) mejorar el uso de dicha plataforma y poder llevar un seguimiento más controlado y exhaustivo de los alumnos en el período escolar.

Será accesible única y exclusivamente desde la plataforma *edmodo*, y su finalidad es complementar a dicha plataforma, nunca sustituirla ni repetir funcionalidad.

A los profesores les permitirá crear asignaturas, calificaciones de diversa índole, pruebas y realizar un control de la conducta y actitud de éstos.

A los alumnos les permitirá visualizar sus datos de asistencia, conducta, actitud, entregas, calificaciones e informes.

A los padres les permitirá visualizar los datos de sus hijos en cuanto a asistencia, conducta, actitud, entregas, calificaciones e informes.

### 2.2. Suposiciones y restricciones

El proyecto debe cumplir las siguientes restricciones:

#### 1. Restricciones de presupuesto:

- Sólo se usarán herramientas FLOSS o versiones educativas que la Universidad de Valladolid disponga.

#### 2. Restricciones de recursos:

- Los miembros del equipo de trabajo deben ser de un máximo de dos personas durante todo el proyecto, tutor y alumno (con los roles de gestor de proyecto, analista, desarrollador y probador).

#### 3. Restricciones de la aplicación:

- El sistema debe tener un compilador del lenguaje de programación Java<sup>[5]</sup>.
- La aplicación debe utilizar un sistema de gestión de base de datos de tipo SQL<sup>[6]</sup>.
- La aplicación web se mantendrá alojada en el servidor de *Heroku*<sup>[7]</sup> para su despliegue.

#### 4. Restricciones de planificación:

- Se debe realizar un análisis y seguimiento de los riesgos asociados al desarrollo del sistema.

### 2.3. Entregables

Mediante la plataforma *Pivotal Tracker*<sup>[8]</sup> se pondrá a disposición de la tutora los diferentes entregables del proyecto.

A continuación se enumeran cada uno de los artefactos que serán generados y utilizados durante el proyecto y que constituyen los entregables. Los artefactos aparecerán listados según la fase de desarrollo a la que pertenecen:

#### 1. Fase de Inicio

- Plan de Desarrollo de *Software*.
- Documento de Gestión de Riesgos.
- Plan de la 1ª iteración de la fase de Inicio.
- Plan de la 1ª iteración de la fase de Elaboración.

#### 2. Fase de Elaboración

- Documento de Especificación de Requisitos *Software*.
- Especificación inicial de casos de uso.

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

- Modelo de análisis.
- Modelo de diseño (incluye la arquitectura del sistema).
- Modelo de datos.
- Prototipo de interfaz de usuario.
- Versión alfa del producto.
- Plan de la 2ª iteración de la fase de Elaboración.
- Plan de la 1ª iteración de la fase de Construcción.

### 3. Fase de Construcción

- Modelo de despliegue.
- Fase de Transición
- Documento de Casos de Prueba.
- Plan de la 2ª iteración de la fase de Construcción.
- Versión beta del producto.
- Versión inicial del manual de usuario.
- Documento de resultados de las pruebas.
- Plan de la 1ª iteración de la fase de Transición.

### 4. Fase de Transición

- Versión final del producto.
- Manual de instalación y despliegue.
- Manual de usuario.
- Manual de administración

Cabe destacar que todos los artefactos son objeto de cambios a lo largo del proceso de desarrollo, por lo que, únicamente al finalizar dicho proceso se dispondrá de una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de la lista de artefactos.

El *Glosario de términos* se incluirá en un documento independiente ya que se irá ampliando con cada fase.

## 2.4. Evolución del plan de Desarrollo de *Software*

El Plan de Desarrollo del *Software* se revisará semanalmente y se refinará antes del comienzo de cada iteración con el objetivo de comprobar que la realización del proyecto cumple lo planificado para que, en caso contrario, se actúe consecuentemente.



### 3. Organización del proyecto

#### 3.1. Estructura de la organización

Los roles que se desempeñarán a lo largo del desarrollo del proyecto son:

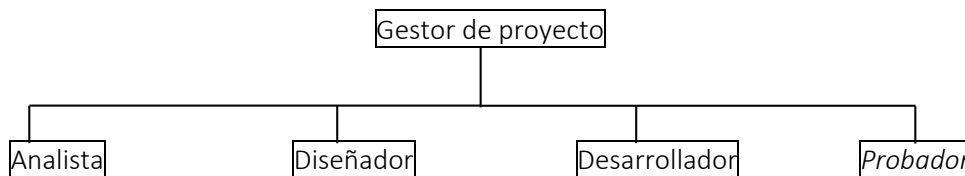


ILUSTRACIÓN 1: ROLES DURANTE EL PROYECTO

#### 3.2. Interfaces externas

Un profesor del colegio San Agustín<sup>[9]</sup> de Valladolid desempeñará el rol de cliente que requiere la aplicación.

#### 3.3. Roles y responsabilidades

La función de cada uno de los miembros dentro de la estructura del equipo es la siguiente:

Rol	Competencias
<b>Gestor de proyecto</b>	Encargado de planificar, dirigir, realizar el seguimiento y controlar el proceso de desarrollo del proyecto.
<b>Analista</b>	Encargado del análisis del sistema y los requisitos de éste, con una garantía de calidad, para garantizar las expectativas del cliente y la gestión de la configuración, para controlar el caos a medida que el software crece.
<b>Diseñador</b>	Encargado del diseño de la arquitectura del sistema y de la base de datos.
<b>Desarrollador</b>	Encargado de la implementación de la aplicación a partir de los modelos generados por los diseñadores.
<b>Probador (Tester)</b>	Encargado de definir, llevar a cabo el plan de pruebas del sistema y de corregir los fallos encontrados.

TABLA 2: DEFINICIÓN DE ROLES

##### 3.3.1. Responsabilidades del proyecto

	Inicio	Elaboración	Construcción	Transición
<b>Cristian T.G.</b>	Analista	Diseñador	Desarrollador	Probador
<b>Yania C.G.C.</b>		Gestor de proyecto*		

TABLA 3: RESPONSABILIDADES DEL PROYECTO

\* **Nota:** en algunas fases Cristian T.G. se encargará del rol de gestor de proyecto, para la planificación de la fase. Cabe aclarar, que el rol de *Gestor de proyecto* de Yania C.G.C. es más orientado a “guía” o “consejero” del proyecto.

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

## 4. Proceso de gestión

### 4.1. Estimaciones del proyecto

En la estimación de costes y tiempos de este proyecto se ha de tener en cuenta que el alumno es un estudiante del Grado de Ingeniería Informática de Sistemas, por lo tanto, no existirá un coste ya que no va a recibir retribución alguna.

Del mismo modo, no habrá ningún coste indirecto en este proyecto. No obstante este es un ejercicio académico. Se hace una estimación del coste basado en el número de horas hombre planificadas y un coste medio de la hora de un Ingeniero Informático supuesto en XXX + 33% de costes sociales. Adicionalmente un supuesto de amortización del *hardware* empleado ya que no hay costes en *software*.

### 4.2. Plan del proyecto

En esta sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

#### 4.2.1. Plan de fases e iteraciones

El desarrollo se llevará a cabo en base al modelo de OpenUP. Se dividirá en fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra la distribución de tiempos y el número de iteraciones de cada fase así como los hitos que determinan el final de cada una de ellas:

Class Control	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

Fase	Iteración	Duración (horas/hombre)	Iteraciones e hitos	Fecha aproximada
Inicio	1	10	Entrega del Plan de Desarrollo y del Documento de Gestión de Riesgos.	Diciembre de 2013
Elaboración	1	80	Validación por parte de la tutora de los modelos de análisis, diseño y datos.	Diciembre de 2013 y enero de 2014
	2	10	Versiones definitivas de los modelos de análisis y diseño, así como entrega del prototipo de la interfaz de usuario (entrega de la versión alfa del sistema).	Enero y febrero de 2014
Construcción	1	110	Desarrollo de <i>software</i> .	Marzo y abril de 2014
	2	30	Entrega de la versión beta del sistema y de la versión inicial del manual de usuario, redacción del documento de resultados de los casos de prueba y validación del modelo de despliegue.	Abril y mayo de 2014
Transición	1	60	Entrega de la versión final del sistema junto a los manuales de instalación y de usuario final.	Mayo y junio de 2014
Horas/hombre totales		300		

**TABLA 4: FASES Y DURACIÓN ESTIMADA**

#### 4.2.2. Entregas

El desarrollo se llevará a cabo en base al modelo de OpenUP. Se dividirá en fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra la distribución de tiempos y el número de iteraciones de cada fase así como los hitos que determinan el final de cada una de ellas:

Software	Descripción	Entrega
Prototipo de interfaz de usuario y versión alfa	Interfaz gráfica de usuario básica sin ningún tipo de funcionalidad. Describe forma de uso.	Fin de la fase de Elaboración
Versión beta	Segundo prototipo funcional posterior a la realización de pruebas.	Fin de la 2ª iteración de la fase de Construcción
Versión final	Producto final.	Fin de la fase de Transición

TABLA 5: ENTREGAS

#### 4.2.3. Calendarización del proyecto

El desarrollo se llevará a cabo en base al modelo de OpenUP. Se dividirá en fases con una o más iteraciones en cada una de ellas.

Como se ha comentado, el proceso iterativo e incremental de OpenUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase de Iteración del proyecto. La siguiente figura ilustra este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina (*workflow*) en un momento determinado del desarrollo.

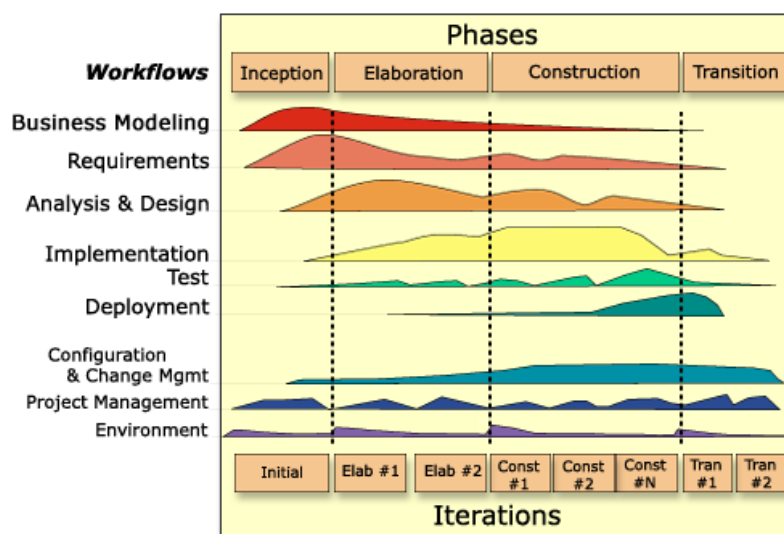


ILUSTRACIÓN 2: FASES DEL DESARROLLO EN OPENUP

#### 4.2.4. Recursos humanos del proyecto

Los recursos humanos disponibles a lo largo de todo el proceso de desarrollo son, el autor del presente documento, Cristian Tejedor García, que cursa los estudios de Grado en Ingeniería Informática de Sistemas y con experiencia laboral limitada; y la tutora Yania Crespo González-Carvajal, encargada de la supervisión y guía del proyecto.

### 4.3. Seguimiento y control del proyecto

#### 4.3.1. Gestión de requisitos

Los requisitos del sistema son especificados en el artefacto "Documento de Especificación de Requisitos *Software*". Cada requisito tendrá un conjunto de atributos que permitirán realizar su seguimiento a lo largo de las diferentes iteraciones del proceso de desarrollo.

En el caso de que sea necesaria la introducción de cambios en los requisitos, éstos serán evaluados y aprobados en un *Documento de Solicitud de Cambio* (ver Anexo) como parte del proceso de gestión de configuraciones y serán introducidos en versiones posteriores del *Documento de Especificación de Requisitos Software*.

#### 4.3.2. Control de calidad

Todos los entregables deben pasar por un proceso de revisión, en el que participará el cliente al final de cada uno, que permita asegurar que los distintos artefactos

<i>Class Control</i>	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

producidos tenga una calidad y aporten valor añadido al proyecto. Para ello se seguirán las guías de revisión y listas de verificación propuestas en OpenUP para la parte de análisis y gestión del proyecto, así como el uso de *tests* unitarios para la parte de software. Los defectos encontrados en los diferentes procesos de revisión serán documentados para permitir su seguimiento y facilitar su corrección.

#### 4.3.3. Informes y medidas

Las medidas del esfuerzo y del tiempo serán tenidas en cuenta a la hora de realizar tanto la planificación como el control y seguimiento de las diferentes actividades involucradas en el proyecto, permitiendo de esta forma hacer una estimación del progreso del mismo.

Los *Planes de Iteraciones* sucesivos servirán a modo de informes para conocer el estado actual del desarrollo. En ellos se enumerarán los criterios para considerar cumplidos los objetivos de cada iteración así como se llevará a cabo una evaluación de la situación actual del proceso de desarrollo y del producto. Esta evaluación se basa en las medidas anteriormente comentadas y en los riesgos.

A partir de esta información se tomarán las decisiones oportunas para mejorar y ajustar el proceso de desarrollo.

#### 4.3.4. Gestión de riesgos

Durante la fase de Inicio se creará un documento con una lista de riesgos asociados al proyecto y sus respectivos planes de contingencia para poder mitigarlos o controlarlos. Esta lista será evaluada al menos una vez en cada iteración.

Para más información sobre este aspecto del desarrollo del proyecto, se adjunta por separado el *Documento de Gestión de Riesgos*.

#### 4.3.5. Gestión de Configuraciones

El objetivo de la gestión de configuraciones es mantener la integridad de los artefactos que se generan durante el proceso de desarrollo, garantizando que no se realizan cambios incontrolados y que se dispone de la versión adecuada de los productos que manejan.

Para ello se realizará la gestión de solicitudes de cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles.

Para el desarrollo del código fuente se utilizará un control de versiones e incidencias con los gestores de versiones distribuidos git<sup>[10]</sup>. Además para agilizar esta tarea se hará uso de la herramienta gratuita Dropbox<sup>[11]</sup>.

Al finalizar cada iteración se actualizará un registro del estado y versión de cada artefacto durante el proceso de evaluación de la situación actual del proceso de desarrollo y del producto que se realiza en el plan de iteración correspondiente.

Class Control	Versión: 1.2
Plan De Desarrollo De Software	21/02/2014

## 5. Anexo

### 5.1. Documento de Solicitud de Cambio

Como ya se explicó en el apartado [4.3.1 Gestión de requisitos](#), se utilizará el siguiente modelo de documento cuando se produzca algún cambio de requisitos una vez establecidos los iniciales de la fase de Elaboración.

<b>Id documento:</b>		
<b>Fecha:</b>		
<b>Creador:</b>		
<b>Aprobado por:</b>		
<b>Título del documento:</b>		
<b>Ref. id. Documento:</b>		
<b>Versión del documento:</b>		
<b>Página</b>	<b>Sección</b>	<b>Razón de cambio</b>

ILUSTRACIÓN 3: FICHA DE SOLICITUD DE CAMBIO