



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INFORMÁTICA

PRÁCTICAS 1, 2 Y 3

Antonio García-Peñuela Molina-Prado

Pilar García Martín de la Puente

Dolores Sesmero Pozo

Laura Fernández Trujillo

Asignatura: Ingeniería del Software
Grupo: 3B.G1.05
Titulación: Grado en Ingeniería Informática
Fecha: 10/12/2015

ÍNDICE

1. Enunciado	3
2. Solución	4
2.1 Análisis de Requisitos	4
2.2 Casos de Uso	5
2.3 Modelado UML	6
2.4 Arquitectura del Sistema	7
2.5 Planificación	8

1. ENUNCIADO

Nuestro equipo de desarrollo está formado por cuatro personas: Laura, Antonio, Pilar y Dolores.

Laura será la jefa del proyecto y se encargará de realizar toda la planificación del mismo además de supervisar y controlar todo el desarrollo del proyecto. Por otro lado, Laura también se encargará de toda la parte de análisis de requisitos ya que será la encargada de hablar con el cliente y toda la parte del diseño del proyecto cuando pueda empezar teniendo en cuenta el trabajo que se necesita realizar previamente y será realizado por el resto de los encargados del proyecto, así como la parte de transición (documentación, integración, despliegue y pruebas).

Antonio será nuestro ingeniero de sistemas y se encargará de montar toda la infraestructura necesaria para poder realizar este proyecto. Además Antonio se encargará de toda la parte del testing, pero cómo hay mucho tiempo de un trabajo a otro hasta que sea requerido en la parte del testing se encargará de montar la infraestructura de otros proyectos que tengamos en desarrollo en la empresa y proporcionando soporte técnico en general en la empresa y a los clientes.

Pilar se encargará de realizar toda la parte de análisis y Dolores toda la parte de implementación, y en cuanto terminen este trabajo serán destinadas a otros proyectos de la empresa.

El sueldo de cada trabajador es:

- Ingeniero de requisitos: 60 €/hora
- Analista: 80€/hora
- Diseñador: 45 €/hora
- Implementador: 30€/hora
- Tester: 15 €/hora

(Los trabajadores que desempeñen tareas distintas tendrán el sueldo en función de las horas que haya realizado de cada tarea).

La jornada laboral es de 8 horas al día, de lunes a viernes excepto los días festivos.

Nuestra estructura monolítica consiste en la creación de una Academia de Inglés que permita administrar todo lo relacionado con los alumnos y empleados existentes en la academia. Las funciones principales que se deben de poder realizar son: autenticación del administrador para poder acceder al sistema y todo lo relacionado con la gestión, así como añadir, modificar, eliminar, listar o mostrar informe de todos y cada uno de las personas almacenadas en nuestra bbdd, tanto empleados como alumnos.

La metodología de desarrollo a seguir es el PUD (Proceso Unificado de Desarrollo) el cual se rige por tres características principales: Dirigido por Casos de Usos, Iterativo e Incremental y Centrado en Arquitectura.

Se pide realizar la planificación del proyecto en función de las prioridades, determinar la fecha final del proyecto y determinar el coste del proyecto suponiendo solo los costes de producción.

Elaborar también una implementación del ejercicio.

2. SOLUCIÓN

2.1. ANÁLISIS DE REQUISITOS

El sistema sólo puede ser usado por el administrativo de la academia de inglés, y siempre tiene que pedir la autenticación para poder acceder al sistema. Para autenticarse el administrativo debe introducir un usuario (será su dni) y contraseña.

El sistema debe permitir almacenar tanto empleados como alumnos de la escuela. De los empleados queremos conocer su nombre completo, DNI, número de teléfono, correo electrónico, puesto de trabajo en la academia (profesor, limpiador, administrativo), nómina, dirección, número de cuenta bancaria, fecha de nacimiento y estado donde pondremos si está activo o dado de baja. De los alumnos queremos conocer su nombre completo, DNI, teléfono, correo electrónico, nº de cuenta bancaria y el nivel de inglés al que está apuntado, además de un atributo de estado (donde al igual que los empleados pongamos si está activo o dado de baja).

El alumno se puede dar de baja (en el caso de que vuelva a la escuela) poniendo el motivo por el cual se da de baja o eliminarlo completamente de la bbdd. En el caso de que el alumno se dé de baja sólo hay que modificar el atributo de estado. Si el alumno no se da de baja sino que simplemente pasa de nivel el atributo que habría que modificar es el del nivel apuntado. El empleado también se puede dar de baja poniendo el motivo por el cual se da de baja o puede ser eliminado completamente de la bbdd en el caso de despido.

Se debe poder modificar cualquier dato tanto de los alumnos como de los empleados almacenados en la bbdd.

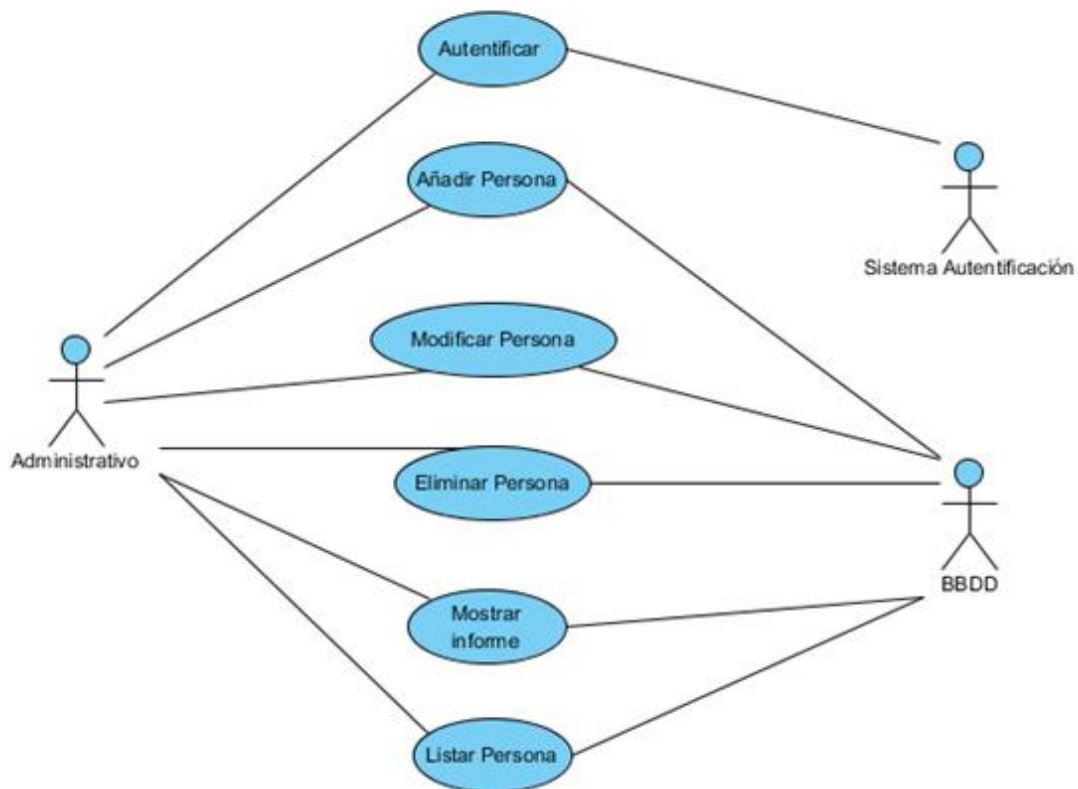
Se debe poder mostrar un listado con todos los empleados almacenados en la bbdd y el papel que desempeñan en la escuela y por otro lado también se debe listar también el nombre de los alumnos matriculados en cada nivel.

Por último, se quiere poder mostrar un informe con todos los atributos almacenados de un alumno o de un empleado buscándolo a través de su dni.

Se requiere implementar una arquitectura monolítica basada en el patrón MVC, en la que habrá además una capa de persistencia que utilizará un sistema gestor de bases de datos relacionales para almacenar los datos.

2.2. CASOS DE USO

1. Autenticar
2. Añadir persona
3. Modificar persona
4. Eliminar persona
5. Mostrar informe
6. Listar persona

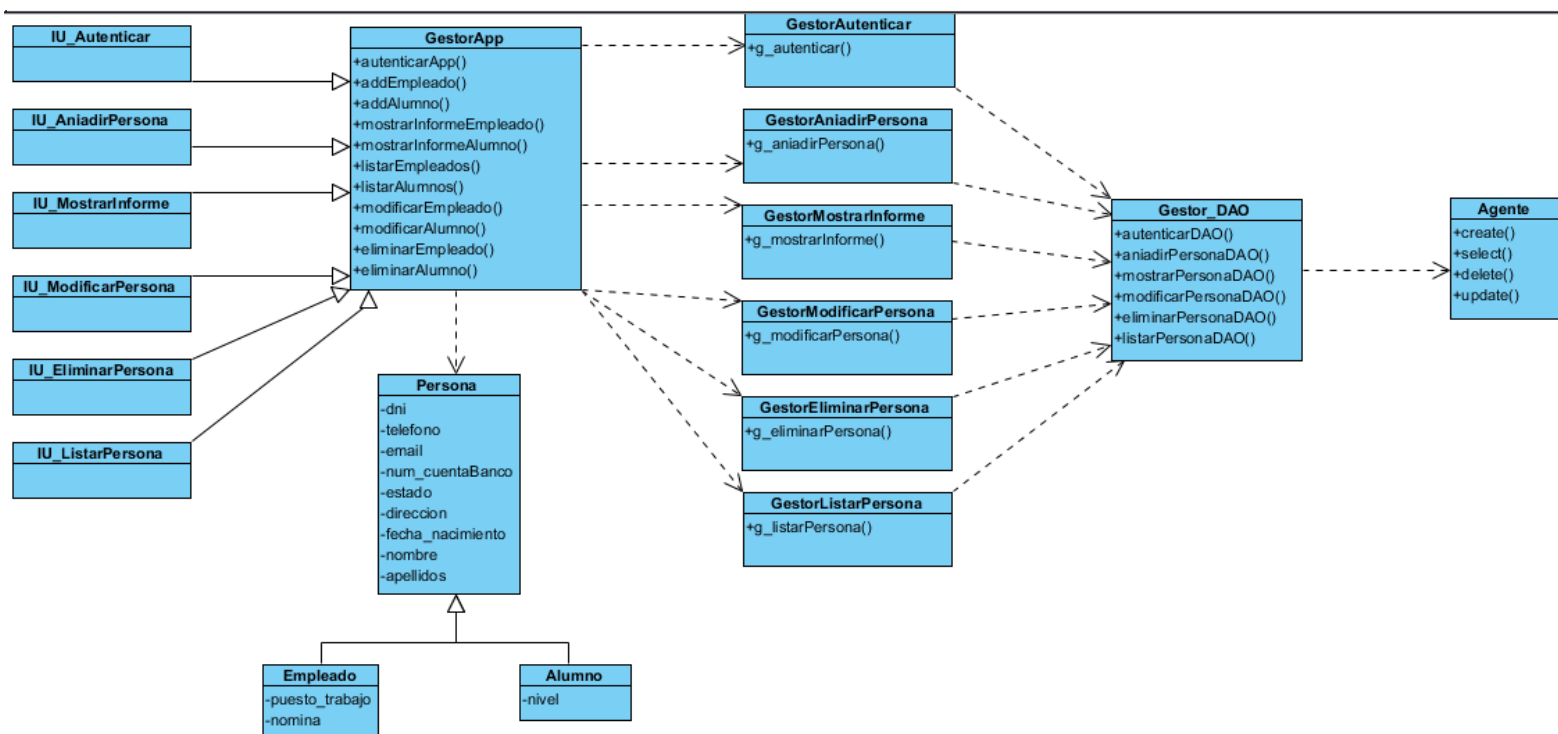


Según nuestra metodología de desarrollo, cada requisito funcional se corresponde con un caso de uso y con una iteración: 1:1:1.

ESCENARIOS ALTERNATIVOS

En la autenticación en el caso de que el administrativo introduzca alguno de los datos mal debe de aparecer un mensaje de error mostrando que alguno de los datos introducidos es erróneo y borrar los campos de escritura para poder introducir los datos de nuevo.

2.3. MODELADO UML



Como podemos ver, tenemos una interfaz gráfica por cada funcionalidad del proyecto, es decir, por cada caso de uso, para que el usuario pueda manejar la aplicación fácilmente.

Las IUs que se comunicarán internamente con GestorApp que tendrá los métodos de las funcionalidades de la aplicación de la Academia de Inglés. Son seis métodos, uno para autenticarse, añadir una persona, mostrar un informe, listar las personas, modificar las personas y eliminar las personas, serán AutenticarApp(), AñadirPersonaApp(), MostrarInformeApp(), ListarPersonaApp(), ModificarPersona() y EliminarPersona() respectivamente.

La clase GestorApp necesita a la clase Persona que contiene los atributos de cada tipo de persona y así modificará (creará, borrará) los atributos de ésta. También llama a cada uno de los gestores (GestorAutenticar, GestorAñadirPersona, GestorMostrarInforme, GestorListarPersona, GestorModificarPersona, GestorEliminarPeronsa).

La clase Persona es llamada también por los gestores de añadir, buscar, borrar y modificar. Y estos cuatro gestores llaman a GestorDAO que finalmente se comunica con Agente que tiene los métodos necesarios para conectarse con la base de datos según las operaciones que queramos hacer.

2.4. ARQUITECTURA DEL SISTEMA

Para implementar esta arquitectura tendremos un ordenador PC Sobremesa - Acer Aspire AXC705, i3-4160, NVIDIA GT705 1GB y 1TB, mas Logitech Wireless Combo MK520 - Juego de teclado y ratón y un monitor - Acer V196HQL HD con VGA.

Para que funcione nuestra arquitectura tendremos un sistema operativo windows 7, con la versión de Java: 7.0, más la aplicación desarrollada (aplicación de la academia) ya instalada y además se instalará un xampp (XAMPP es una distribución de Apache completamente gratuita y fácil de instalar que contiene MariaDB, PHP y Per) que interactúa con la base de datos.

Los costes totales del montaje del sistema son: 649,3€

2.5. PLANIFICACIÓN

A continuación se muestra la asignación de los casos de uso a las iteraciones:

Inicio	Elaboración		Construcción				Transición
It0	It1	It2	It3	It4	It5	It6	It7
Plan de Proyecto	CDU1 Autenticar	CDU2 Añadir Persona	CDU5 Mostrar informe	CDU6 Listar Persona	CDU3 Modificar Persona	CDU4 Eliminar Persona	Documentación Integración Despliegue Entrega

La siguiente tabla, muestra la asignación de prioridades a los casos de uso y la estimación del esfuerzo para cada tarea:

Req#	CDU	Prioridad	Requisitos	Análisis	Diseño	Implementación	Pruebas
1	1	1	15	30	60	100	30
2	2	2	20	40	70	120	40
3	5	3	15	30	50	80	30
4	6	3	15	30	60	110	40
5	3	4	15	30	60	120	40
6	4	4	15	30	60	120	40

*Los datos de tiempo están expresados en minutos

Consideramos que para la iteración 0 se necesitan 8 horas de trabajo. Las horas para este trabajo se pagan con 100 €/hora.

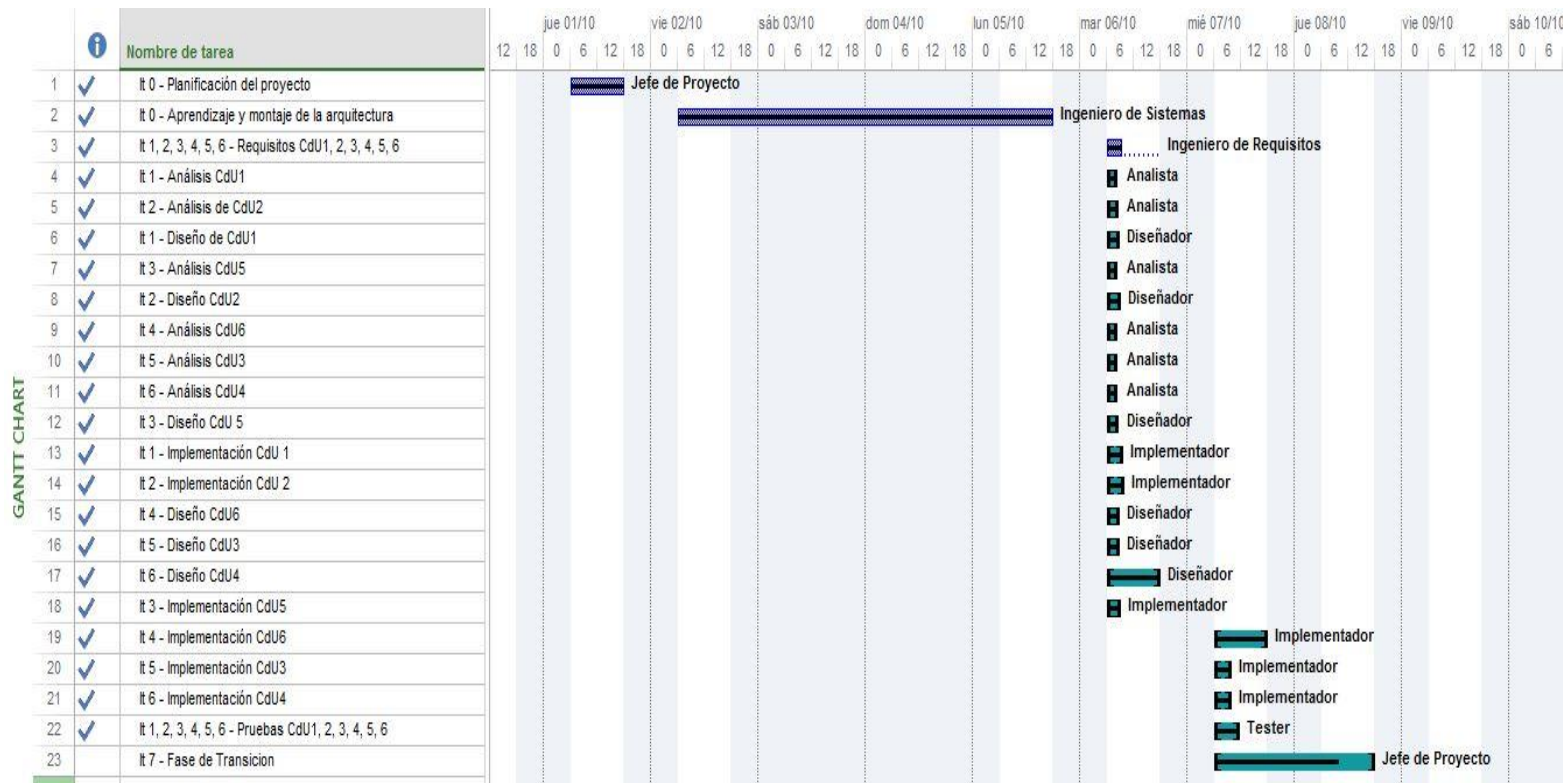
Es necesario un ingeniero de sistemas para montar la infraestructura que necesita 8 horas de aprendizaje y otras 8 horas de montar el sistema. Cobra 30 euros/hora, lo que supone un coste total de 480 euros.

Para la iteración 7 son necesarias 10 horas que se pagan a 40 €/hora.

A continuación vemos cada recurso con su tasa del coste en € por horas:

	Nombre del	Tipo	Etiqueta de	Iniciales	Grupo	Capacidad	Tasa	Tasa horas	Costo/Use	Acumu	Calendario
	Ingeniero de Requisitos	Trabajo		I		100%	\$60,00/hr	\$60,00/hr	\$0,00	Prorrateo	Standard
	Jefe de Proyecto	Trabajo		J		100%	\$100,00/hr	\$100,00/hr	\$0,00	Prorrateo	Standard
	Ingeniero de Sistemas	Trabajo		I		100%	\$25,00/hr	\$25,00/hr	\$0,00	Prorrateo	Standard
	Analista	Trabajo		A		100%	\$80,00/hr	\$80,00/hr	\$0,00	Prorrateo	Standard
	Diseñador	Trabajo		D		100%	\$45,00/hr	\$45,00/hr	\$0,00	Prorrateo	Standard
	Implementador	Trabajo		I		100%	\$30,00/hr	\$30,00/hr	\$0,00	Prorrateo	Standard
	Tester	Trabajo		T		100%	\$15,00/hr	\$15,00/hr	\$0,00	Prorrateo	Standard

La planificación de nuestro proyecto teniendo en cuenta que comenzaría el 1 de octubre. En la siguiente línea de tiempo podemos ver de forma clara el solape de las tareas, su inicio y finalización. Además, al final de cada tarea, se aprecia el recurso que la realiza.



En el siguiente Diagrama de Gantt mostramos la planificación de cada iteración. La primera columna indica con un “tic” si ya está terminada esa tarea. En la tercera columna indicamos que iteración y que tarea es realizada por cada fila. La cuarta columna nos dice las horas que se va a llevar a cabo terminar cada tarea. Y en la quinta y sexta, la fecha de inicio y de finalización. La última columna nos dice qué recurso desempeña la tarea.

DIAGRAMA DE GANTT

		Modo de	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	✓		It 0 - Planificación del proyecto	8 hrs	jue 01/10/15	jue 01/10/15		Jefe de Proyecto
2	✓		It 0 - Aprendizaje y montaje de la arquitectura	16 hrs	vie 02/10/15	lun 05/10/15		Ingeniero de Sistemas
3	✓		It 1, 2, 3, 4, 5, 6 - Requisitos CdU1, 2, 3, 4, 5, 6	95 mins	mar 06/10/15	mar 06/10/15		Ingeniero de Requisitos
4	✓		It 1 - Análisis CdU1	30 mins	mar 06/10/15	mar 06/10/15		Analista
5	✓		It 2 - Análisis de CdU2	40 mins	mar 06/10/15	mar 06/10/15		Analista
6	✓		It 1 - Diseño de CdU1	1 hr	mar 06/10/15	mar 06/10/15		Diseñador
7	✓		It 3 - Análisis CdU5	30 mins	mar 06/10/15	mar 06/10/15		Analista
8	✓		It 2 - Diseño CdU2	70 mins	mar 06/10/15	mar 06/10/15		Diseñador
9	✓		It 4 - Análisis CdU6	30 mins	mar 06/10/15	mar 06/10/15		Analista
10	✓		It 5 - Análisis CdU3	30 mins	mar 06/10/15	mar 06/10/15		Analista
11	✓		It 6 - Análisis CdU4	30 mins	mar 06/10/15	mar 06/10/15		Analista
12	✓		It 3 - Diseño CdU 5	50 mins	mar 06/10/15	mar 06/10/15		Diseñador
13	✓		It 1 - Implementación CdU 1	100 mins	mar 06/10/15	mar 06/10/15		Implementador

DIAGRAMA DE GANTT

		Modo de	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
14	✓		It 2 - Implementación CdU 2	120 mins	mar 06/10/15	mar 06/10/15		Implementador
15	✓		It 4 - Diseño CdU6	60 mins	mar 06/10/15	mar 06/10/15		Diseñador
16	✓		It 5 - Diseño CdU3	60 mins	mar 06/10/15	mar 06/10/15		Diseñador
17	✓		It 6 - Diseño CdU4	1 día	mar 06/10/15	mar 06/10/15		Diseñador
18	✓		It 3 - Implementación CdU5	80 mins	mar 06/10/15	mar 06/10/15		Implementador
19	✓		It 4 - Implementación CdU6	1 día	mié 07/10/15	mié 07/10/15		Implementador
20	✓		It 5 - Implementación CdU3	120 mins	mié 07/10/15	mié 07/10/15		Implementador
21	✓		It 6 - Implementación CdU4	120 mins	mié 07/10/15	mié 07/10/15		Implementador
22	✓		It 1, 2, 3, 4, 5, 6 - Pruebas CdU1, 2, 3, 4, 5, 6	230 mins	mié 07/10/15	mié 07/10/15		Tester
23			It 7 - Fase de Transición	16 hrs	mié 07/10/15	jue 08/10/15		Jefe de Proyecto

Según las especificaciones y la planificación del proyecto se ha estimado que el día final del proyecto es el 8 de octubre (teniendo en cuenta que la jornada laboral es de 8 horas).

Teniendo en cuenta todos los datos el coste total del proyecto sería: 3327,63 €