



UNIVERSIDAD DE BURGOS
ESCUELA POLITÉCNICA SUPERIOR
Grado en Ingeniería en Informática



TFG del Grado en Ingeniería Informática
DOCUMENTACIÓN TÉCNICA



Presentado por Adrian Aguado
en Universidad de Burgos — 3 de junio de 2017
Tutor: Luis R.Izquierdo

Índice general

| | |
|---|-----------|
| Índice general | I |
| Índice de figuras | II |
| Apéndice A Plan de Proyecto Software | 1 |
| A.1. Introducción | 1 |
| A.2. Planificación temporal | 2 |
| A.3. Estudio de viabilidad | 2 |
| Apéndice B Especificación de Requisitos | 5 |
| B.1. Introducción | 5 |
| B.2. Objetivos generales | 6 |
| B.3. Catálogo de requisitos | 6 |
| B.4. Especificación de requisitos | 6 |
| Apéndice C Especificación de diseño | 7 |
| Apéndice D Documentación técnica de programación | 8 |
| D.1. Introducción | 8 |
| D.2. Programas mínimos necesarios | 8 |
| D.3. AngularCLI | 9 |
| D.4. Manual del programador | 9 |
| Apéndice E Documentación de usuario | 11 |
| E.1. Introducción | 11 |
| E.2. Instalación | 11 |
| E.3. Manual del usuario | 11 |

Índice de figuras

Plan de Proyecto Software

A.1. Introducción

En este capítulo se detalla la planificación del proyecto. Como gestor de tareas se comenzó utilizando *Trello+Github* pero más tarde se pasó a utilizar *Zenhub*, extensión de Google Chrome que permiten integrar los *boards* dentro del mismo repositorio de código alojado en *github*. Ya se han dado más detalles en la memoria del proyecto.

Se ha utilizado metodologías ágiles para el desarrollo del proyecto y de este modo, se ha realizado un desarrollo dividido en iteraciones. Terminada una iteración empezaba la siguiente y se agregaban a las tareas planeadas las que no habían sido completado de la iteración precedente. Las iteraciones del proyecto estaban pensadas para durar una semana aproximadamente. No obstante, hay alguna excepción en la que la iteración duró más tiempo.

La fase de planificación se puede dividir a su vez en:

- Planificación temporal.
- Estudio de viabilidad.

La primera parte me centro en la programación y desarrollo de la aplicación. Es decir elaboro un programa de tiempos con una serie de tareas a seguir para cumplimentar el proyecto.

La segunda parte se centra en el estudio de viabilidad. De la misma manera desde la segunda semana de marzo vengo realizando un plan de empresa con el programa Yuzz por lo que ello me va a facilitar el estudio de viabilidad de mi proyecto. Se desarrollará tanto la viabilidad legal como la económica.

A.2. Planificación temporal

Desde inicio del proyecto se planteó utilizar una metodología ágil como *Scrum* para la gestión del proyecto. Aunque no se ha seguido al 100 % la metodología al tratarse de un proyecto para la Universidad, sí que se ha aplicado en líneas generales una filosofía ágil y metódica.

A continuación se describen los diferentes *sprints* que se han realizado. Dentro de *github* cada *milestone* recibe el número del sprint asignado y dentro de cada uno de ellos existen diferentes tareas que describiré a continuación

Sprint 0: 18/02/2017 - 28/02/2017

Tareas:

- Terminar formación.
- Aprender *Sonarqube*.
- Inicio Back-End.
- Inicio Decidir base de datos a emplear.

| Completed Issues and Pull Requests | Story points |
|--|--------------|
| Terminar formación courses barterAPP #1 III New Issues ↑ Sprint 0 | 8 |
| Aprender uso https://www.sonarqube.org/ research various barterAPP #2 III New Issues ↑ Sprint 0 | 1 |
| Decidir base de datos a usar. research various barterAPP #3 III New Issues ↑ Sprint 0 | 1 |
| Comenzar desarrollo Back-End. web app (Back-End) barterAPP #5 III New Issues ↑ Sprint 0 | 8 |
| Información sobre bases de datos NoSQL courses research barterAPP #6 III New Issues ↑ Sprint 0 | 5 |

Figura A.1: Detalle sprint 0

La primera vez que hice uso de *ZenHub* ya llevaba algún tiempo formándome, de ahí el primer *Issue* del Sprint 0. Esta primera toma de contacto fue para comenzar a desarrollar el Back-End de la aplicación, además se consultaron varias fuentes para decidir que tipo de base de datos emplear.

Sprint 1: 18/02/2017 - 28/02/2017

Tareas:

- Primera tarea.

- Segunda tarea.

Breve texto describiendo las tareas

Sprint 2:

Sprint 3:

Sprint 4:

Sprint 5:

Sprint 6:

Sprint 7:

A.3. Estudio de viabilidad

En esta sección se lleva a cabo un estudio para comprobar la viabilidad del proyecto realizado. Paralelamente al desarrollo de la aplicación, como ya se ha nombrado en la memoria, el proyecto formó parte del programa YUZZ para jóvenes emprendedores en el que durante cinco meses realicé un plan de empresa completo. Se detalla por tanto en un documento que adjuntaré al proyecto un estudio de viabilidad exhaustivo y muy completo en el que se incluyen entre otras cosas: plan de marketing, plan de financiación, estudio de viabilidad o plan de puesta en marcha del negocio a cinco años vista.

Por lo tanto en esta sección voy a realizar un resumen del documento descrito en el párrafo anterior en el que como conclusión definitiva tendremos un boceto de lo que supondría transformar un proyecto fin de carrera y que pase a formar parte del mercado. Así mismo voy a intentar adaptarlo a las condiciones que se exigen en el proyecto dado que el plan de empresa completo es un estudio de viabilidad completo de aquí a cinco años por lo que resulta ser más extenso y detallado. Se intentará por tanto aquí realizar una estimación.

Viabilidad económica

La viabilidad económica es la parte donde lograremos detectar si el proyecto es o no rentable económicamente hablando.

Análisis de costes

Económica

Coste de personal Se considerará que el proyecto ha sido desarrollado en un periodo de cinco meses. Considerando que se ha trabajado unas 6 horas a día cada semana, y que el programador, que en este caso es una

sola persona, ha percibido un sueldo de 13 e/hora, el coste del personal por lo tanto se resume en la siguiente tabla:

| | Total |
|------------------------------|-----------------------|
| 13 €/hora * 6 horas/día | 78 €/día |
| 78 €/día * 5 días/semana | 390 €/semana |
| 390 €/semana * 4 semanas/mes | 1560 €/mes |
| Coste total salario | 7800 €/5 meses |

Tabla A.1: Tabla salarios.

Coste de seguridad social más información sobre el segundo item.

Coste de software más información sobre el segundo item.

Coste de Hardware más información sobre el segundo item.

Viabilidad legal

La viabilidad legal se centra principalmente en el estudio de las licencias software utilizadas y en la licencia que se le va a ser asignada a las diferentes aplicaciones desarrolladas.

Especificación de Requisitos

B.1. Introducción

Este anexo recoge la especificación de requisitos que define el comportamiento del sistema desarrollado. El objetivo principal de la Especificación de Requisitos del Sistema (*ERS*) es servir como medio de comunicación entre clientes, usuarios, ingenieros de requisitos y desarrolladores.

La ERS es correcta si y sólo si todo requisito que figura en ella refleja alguna necesidad real. La corrección de la ERS implica que el sistema implementado será el sistema deseado. Se han seguido las recomendaciones del estándar IEEE 830 según la última versión del estándar IEEE 830. Las características deseables para una especificación de requisitos son:

1. No ambigua.
2. Completa
3. Verificable
4. Consistente
5. Clasificada
6. Modificable
7. Explorable
8. Utilizable durante las tareas de mantenimiento y uso

| CU-02 | Crear cuestionarios |
|-----------------------------|---|
| Versión | 1.0 |
| Autor | Adrián Aguado |
| Requisitos asociados | RF-1.1, RF-1.2 |
| Descripción | Permite al usuario logearse |
| Precondición | LAs precondiciones deben estar disponibles |
| Secuencia | 1. El usuario rellena el formulario para 2. El usuario 3. Si se |
| Postcondición | Se añade el usuario a la base de datos. |
| Excepciones | Si no se introducen los campos obligatorios se muestra texto resaltando indicándolo |
| Frecuencia | Alta |
| Importancia | Alta |
| Comentarios | Esto es un comentario sobre este requisito. |

Tabla B.1: CU-02 Crear cuestionarios

B.2. Objetivos generales

Los objetivos generales que se perseguían con el proyecto han sido:

- Desarrollar una aplicación web para solventar el problema de la gestión de turnos.
- Formarse en tecnologías web.
- Aprender a realizar una aplicación móvil híbrida a través de diversas plataformas como por ejemplo PhoneGapp.

B.3. Catálogo de requisitos

A continuación, se enumeran los requisitos específicos :

Requisitos funcionales

B.4. Especificación de requisitos

Apéndice C

Especificación de diseño

Apéndice *D*

Documentación técnica de programación

D.1. Introducción

En este capítulo vamos a dividirlo en dos partes, por un lado los condiciones específicas para la aplicación y por otro vamos a aprender de manera breve y sencilla a crear una aplicación gracias a *Angular CLI*, la interfaz de línea de comandos de Angular.

El por qué he decidido incluir la parte de *Angular CLI* es por que considero que va aclarar muchas dudas acerca de cómo funciona esta tecnología y también para intentar comprender mejor como he realizado el proyecto. En diversas ocasiones para aprender a usar una tecnología tan solo hace falta tiempo pero en otras muchas el tiempo es limitado por lo que quizás con esta pequeña guía podemos comenzar a utilizar un framework cliente en muy pocos pasos y de una manera sencilla. Si bien es cierto que yo he necesitado mucho tiempo para comprender su funcionamiento y no lo he entendido gracias a la interfaz de comandos, ésta te permite agilizar los trámites de creación en un tanto por ciento considerable.

D.2. Programas mínimos necesarios

Especificación de requisitos usuario

Especificación de requisitos programador

Este anexo tiene como objetivo analizar y documentar las necesidades funcionales que deberán ser soportadas por el sistema a desarrollar, es decir, en qué condiciones ha sido desarrollado, en qué condiciones se debe usar y cuáles son los requerimientos mínimos para que funcione.

D.3. AngularCLI

Sabiendo los requisitos mínimos que son necesarios para proceder

¿Qué es typescript?

Esto se ha nombrado antes pero resulta esencial para entender el código de Angular

TypeScript is a typed superset of JavaScript that compiles to plain JavaScript. Any browser. Any host. Any OS. Open source.

([?])

Comandos básicos

Puesta en marcha

D.4. Manual del programador

En esta sección hay que tener en cuenta que el autor de este trabajo fin de grado a escogido unas herramientas, tanto para desplegar la app, como la base de datos como para desarrollar la aplicación pero que de ninguna manera resultan ser ni las únicas ni las mejores simplemente son unas herramientas que ha considerado utilizar pero existen muchas más que no son ni peores ni peores.

Para desplegar la app hemos elegido heroku, para la base de datos mlab (que dentro tiene servidores AWS, Google Cloud o Azure)

Pasos para montar en tu propio ordenador y desarrollar tu propia API (tener en cuenta que es software desde el que se realiza es un MACBOOK PRO, por lo que pueden existir cambios respecto a otros sistemas operativos. Trataré sin embargo ajustarme y dar detalles para instalarlo en cualquier entorno.

Compilación, instalación y ejecución del proyecto

Voy a tratar de explicar un desarrollo completo desde la instalación en local hasta la carga en un servidor.

*APÉNDICE D. DOCUMENTACIÓN TÉCNICA DE PROGRAMACIÓN*10

Modo desarrollador

Modo producción

Estructura de directorios del proyecto

Base de datos

Avisos

Documentación de usuario

E.1. Introducción

En este capítulo se detalla como un usuario puede comenzar a usa la aplicación deberemos diferenciar dos aspectos diferentes:

- Aplicación Web.
- Aplicación móvil.

E.2. Instalación

Aplicación Web

Aplicación Móvil

E.3. Manual del usuario

Aplicación Web

Aplicación Móvil