

# 1 Introducción

Son varios los caminos que han llevado al desarrollo de este proyecto. En este capítulo tendré la oportunidad de poder hablar de cada uno de ellos. Desde empezar a estudiar ingeniería informática hasta la mentalidad que he ido adquiriendo a lo largo de estos años gracias a ella.

## 1.1 Motivaciones

La motivación principal ha sido la ingeniería informática y es que sin ella no sería quién soy ahora. Para bien y para mal, ha ido siendo una evolución de pequeños pasos empezando en primero de carrera hasta llegar a saber una pizca de todo este mundo de tecnología ya acabando cuarto. Todo empezó desde pequeño, desde que vi esos monitores gigantes, los disquetes y el Age of Empires 1 a la edad de cuatro años.

Mi madre tuvo un novio, Martín se llamaba, que nos trajo el mundo de la tecnología a la casa. Ella también era otra fan tecnológica, pues al haberme tenido con tan solo 17 años siempre tuvo esa mentalidad más abierta con todo el tema de nuevas herramientas. Creo que esto me acabó ayudando bastante.

Luego de esta pequeña época vino una un poco más oscura que consistía en una vida de cero tecnología. Influyó bastante el aspecto de vivir en Argentina y que el precio de los aranceles fuera bastante alto influyendo en los productos tecnológicos en su mayoría. Pasar al lado de los escaparates y ver esas pantallas brillantes con juegos, películas, canciones... era un martirio para mí.

Otro aspecto muy importante y el cual también he de agradecer fue la concesión que hizo la Junta de Andalucía allá por 2009 de aquellos pequeños ordenadores verdes. Esos sí que fueron gasolina para mis aspiraciones y para terminar hoy aquí, dentro del grado de Ingeniería Informática de la Universidad de Almería.

Mi segunda motivación principal para el desarrollo de este proyecto fue la beca extracurricular que publicó el Departamento de Informática de la Universidad de Almería, cuyo director en el momento de la concesión y en la fecha actual de redacción de este documento es Juan Francisco Sanjuan Estrada, el tutor de este proyecto.

Gracias a ella me sentí motivado para poder afianzar la actividad desempeñada en aquel pequeño trayecto de tiempo, unos seis meses, a la redacción de este trabajo.

El trabajo no se basa en lo que exactamente realicé en aquella beca, sino en una refactorización/transformación de aquel proyecto para poder convertirla en algo “fresco” podríamos llamarlo.

Y aquí pasamos al tercer y último punto de este apartado. La transformación del

proyecto, el aplicar el concepto de ingeniería para poder transformar algo que era un software cerrado, con poca modularidad, difícil de entender y de ampliar en algo que merezca la pena tener. Un software que combine tecnologías y metodologías de hoy en día. Un ejemplo a seguir.

### 1.2 Objetivos

El objetivo principal de este proyecto es la creación de un sistema de gestión de inventario y creación de préstamos para el Departamento de Informática de la Universidad de Almería.

Este sitio web tiene que cumplir con unos requisitos principales, que son:

- Llevar un registro del inventario del Departamento de Informática ubicado en el edificio Científico Técnico III
- Que el estudiantado y el personal docente e investigador realicen solicitudes de préstamos para los distintos elementos ofertados dentro de la página
- Que los técnicos de servicio puedan gestionar estas solicitudes más el seguimiento del inventario dentro del edificio

Por estos puntos entendemos que la motivación principal de la herramienta es la de gestionar y organizar préstamos.

Luego de los requisitos principales que había que cumplir se nos presentaban los secundarios.

- La página web tenía que disponer de un modo adaptativo para las versiones móviles
- Había que incorporar la posibilidad de realizar un importado de datos gracias al procesamiento de una tabla de datos que es lo que antiguamente utilizaban los técnicos del departamento
- Había una serie de datos que tenían que almacenar los objetos añadidos a este sistema los cuales podían ser de tres tipos: inventarios, fungibles o kits
- Se permitiría ver un seguimiento de los préstamos realizados sobre los objetos.

En base a estos objetivos principales y secundarios realizaremos la construcción de la aplicación.

### 1.3 Planificación

La planificación del proyecto se dividirá en cinco grandes grupos: reuniones iniciales con los clientes, planificación y elaboración del desarrollo del proyecto, preparación del entorno de trabajo, construcción de la aplicación, testeo y comprobación de la aplicación y comprobación de errores.

### 1.3.1 Reuniones iniciales con los clientes

A pesar de basarse en la reconstrucción de una aplicación las reuniones con los clientes son iniciales. Estas se hicieron en un principio del desarrollo y se mantuvieron unas pocas más a lo largo del mismo. Terminando añadiendo nuevas funcionalidades o descripciones de los productos.

### 1.3.2 Planificación y elaboración del desarrollo del proyecto

En el desarrollo del proyecto no solo se considera la planificación de las diferentes etapas para poder ver y realizar un seguimiento del proyecto sino que también se considera la redacción y especificación de las diferentes actividades realizadas en cada una de estas. Este apartado de la planificación se desarrolla junto a todo el resto de apartados.

### 1.3.3 Preparación del entorno de trabajo

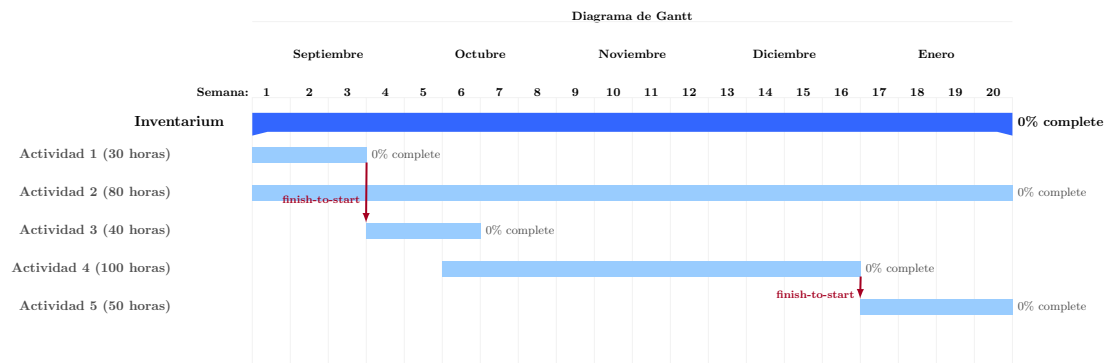
Este apartado me hace especial ilusión ya que considero que gran parte de la actividad que realizas depende del entorno donde te manejas. Los mayores casos de éxito tanto en producción como en mejoras de producto de una empresa se debe a la mejora de su entorno de producción. Gracias a algunas asignaturas del grado, a la investigación, a poder haber estado alrededor de un año en el entorno laboral y a la magnífica herramienta que es internet y todo lo que nos brinda he podido perfeccionar esta creación de entornos de trabajo y cada día siento que puedo ir mejorando un poquito más en cada aspecto que rodea a esta intencionalidad.

### 1.3.4 Construcción de la aplicación

Uno de los apartados más difíciles en cierto momento, por lo que conllevaba la creación de la lógica de la aplicación, las interacciones que realiza la interfaz con el usuario y las distintas funcionalidades que tiene que presentar la misma. Esta sección unos dos años atrás pudo haber sido de las que más tiempo podrían llevar pero gracias a los distintos frameworks que tenemos hoy en día para poder reutilizar componentes software y ahorrar pasos intermedios se ha ido volviendo más pequeña, que no quiere decir menos importante. Gracias a Angular 12 y a algunas tecnologías más que utilizaremos veremos como la construcción no es tan complicada como en un principio parecía.

### 1.3.5 Testeo y comprobación de la aplicación y comprobación de errores

¿Qué es de una aplicación bien planificada, con una interfaz bien elaborada y unos requisitos satisfechos que presente errores? No es nada, y es por ello que un testeo intensivo en el momento que la aplicación finalmente haya sido publicada hará que mejore en esos apartados que anteriormente podría haber presentado fallos.



- **Actividad 1** (30 horas) Reuniones iniciales con los clientes
  - Reunión inicial con el director del proyecto
  - Reunión con los técnicos para investigar el dominio de la aplicación
  - Presentar primer diseño de la aplicación junto a una interpretación del dominio
- **Actividad 2** (80 horas) Planificación y elaboración del desarrollo del proyecto
- **Actividad 3** (40 horas) Preparación del entorno de trabajo
  - Creación del entorno en la nube Google Cloud
  - Creación del repositorio en GitHub
  - Creación del entorno virtual en Visual Studio Code
  - Contenerización de los distintos sistemas que se utilizarán en el desarrollo de la aplicación
- **Actividad 4** (100 horas) Construcción de la aplicación
  - Elaboración y creación de la base de datos
  - Desarrollo de la API
  - Desarrollo del sitio web
- **Actividad 5** (50 horas) Testeo, comprobación de la aplicación y comprobación de errores

## 1.4 Estructuración del documento

La estructura de este documento de trabajo fin de grado es la siguiente:

### 1.4.1 Herramientas utilizadas

Donde haremos un pequeño resumen de todas las herramientas tanto físicas como tecnológicas utilizadas para la construcción de este software.

### **1.4.2 Desarrollo de las fases del proyecto**

Donde se describirán y detallarán todas las fases de desarrollo del proyecto. Esta es la más extensa.

### **1.4.3 Conclusiones y posibles mejoras**

Evaluaremos las distintas fases de desarrollo, su desempeño en ellas y posibles mejoras que se hubieran podido realizar.

### **1.4.4 Bibliografía**

Hablaremos sobre la bibliografía en la que se han fundamentado cada uno de los pasos de elaboración de este proyecto.



## 2 Herramientas utilizadas

En este capítulo hablaré sobre las herramientas utilizadas durante el desarrollo del proyecto. Me parece bastante interesante debido a la inclusión de nuevas herramientas que irán en conjunción a nuevas metodologías de trabajo.

Al igual que las tecnologías van avanzando las empresas también lo hacen, y la amplia gama de tecnologías gratuitas que están a nuestro alcances conlleva que el conocimiento de las mismas sea de vital importancia en el momento de construcción de una aplicación.

### 2.1 Hardware

Dentro del apartado de hardware disponemos de dos ordenadores. Su procesador no conlleva relevancia en el desarrollo de la aplicación debido a la utilización de servicios en la nube.

#### 2.1.1 Torre de PC

La cual contiene como procesador un Xeon E5-2620 V3. 16GB de RAM DDR3. Una tarjeta gráfica RTX 570 de 4GB DDR5. Tiene 256GB de memoria SSD y 1TB de memoria HDD.

#### 2.1.2 Un portátil

Es un MacBook Air M1 de 2020 con 8GB de RAM y 256GB de almacenamiento.

### 2.2 Software

#### 2.2.1 Entorno de desarrollo

Nuestro entorno de desarrollo y desde donde haremos casi absolutamente todo será desde Visual Studio Code. Este es un editor de código desarrollado por Microsoft que soporta varias distribuciones de sistemas operativos, entre ellas: Windows, Mac Os y Ubuntu.

Una de las características de esta herramienta que la hacen la predilecta de varios desarrolladores es el gran soporte que tiene por parte de la comunidad. Tiene un mercado de plugins bastante grande que apoya la creación continua de código para todos los desarrolladores.

Dispone de integraciones con Git, resaltado en errores de sintaxis, finalización de código y hasta conexión remota a otros entornos de trabajo mediante SSH.

Otra enorme ventaja que presenta es el consumo de memoria que tiene, bastante pequeño. Es un programa para ordenadores de todos los tamaños y precios, un software gratuito y una herramienta increíblemente potente al alcance de todos.

### 2.2.2 Redacción del documento

Estas líneas están siendo escritas ahora mismo desde LaTeX. LaTeX es un sistema de composición de textos que está formado mayoritariamente por órdenes construidas a partir de comandos TeX. En un principio no estaba seguro de qué herramienta utilizar, ya que la posición de varios profesores respecto a esta herramienta era bastante férrea pero Word siempre había ido agarrado a mi mano desde comienzos del instituto.

Luego de pasar de Word a LaTeX y de LaTeX a Word bastantes veces no fue hasta que mi profesora Rosa, en una de mis visitas matinales a su despacho me dijo: Yo hice mi TFG en LaTeX.

No me lo podía creer y al comprobar la fecha de publicación de este programa de procesamiento de textos me sorprendí al ver que su lanzamiento oficial fue en 1980. “Si el programa ha durado tanto es que algo de importante tendrá” pensé. Y aquí me hallo redactando este documento con un programa que facilita el control de versiones de Git de una manera asombrosa. Facilita también los procesos de documentación y disposición de las diferentes subsecciones. Y, lo que más me gusta sin lugar a dudas, que puedo realizar una separación de cada capítulo por documentos separados y es que a mí, el tener las cosas descompuestas, me puede.

### 2.2.3 Diseño y creación de la base de datos

La base de datos ha sido diseñada con MySQL Workbench. Esta es una herramienta visual de diseño de bases de datos, capaz de administrar, diseñar, gestionar y mantener bases de datos. Su primera versión fue publicada en 2005.

En un principio la base de datos fue exportada para que el deploy también se realizaría en un servidor MySQL pero el versionado de estos scripts para la creación de las bases de datos me llevan dando problemas tres años. Grata fue mi sorpresa al descubrir MariaDB y que ofrece una compatibilidad perfecta con MySQL, además, es software libre.

MariaDB es un sistema de gestión de bases de datos derivado de MySQL con licencia GPL. Fue escrito por el mismo creador que MySQL, y esto ¿por qué? Porque vendió su producto (MySQL) a Oracle, dejando este de ser software libre.

Para la gestión de la base de datos he utilizado PHPMyAdmin. Un gestor de bases de datos que se utiliza desde páginas web. Facilita cualquier tipo de inspección que un técnico tenga que realizar por no poder hacerlo desde la aplicación web de inventarium.

### 2.2.4 NodeJS

NodeJS es un entorno de tiempo de ejecución de JavaScript. Este entorno de tiempo de ejecución en tiempo real incluye todo lo que se necesita para ejecutar un programa escrito en JavaScript. Gracias a él podremos utilizar varias herramientas que explicaré a continuación.



### 2.2.5 Levantamiento del servidor web y de la Interfaz de Programación de Aplicaciones(API)

El servicio Web se levanta desde Express. Express es framework back-end para NodeJS que está diseñado para levantar sitios webs y APIs. Es la herramienta predilecta para levantar servicios web dentro del entorno de Node.

### 2.2.6 Desarrollo del sitio web

El desarrollo del sitio web se realizará con Angular 12. Angular es sin duda el punto más importante de esta subsección.

Angular es un framework para aplicaciones web desarrollado en TypeScript, de código abierto, mantenido por Google, que se utiliza para crear y mantener aplicaciones web de una sola página. Su objetivo es aumentar las aplicaciones basadas en navegador con capacidad de Modelo Vista Controlador (MVC), en un esfuerzo para hacer que el desarrollo y las pruebas sean más fáciles.

Angular se basa en clases tipo Componentes, cuyas propiedades son las usadas para hacer el binding de los datos.

Angular es la evolución de AngularJS (la versión de Angular que usaba JavaScript) aunque incompatible con su predecesor.

Este framework para mí ha sido como el descubrimiento de América. Y es que en la anterior versión de la aplicación había utilizado únicamente PHP, Javascript y CSS. Es más, había conseguido desarrollar un modelo basado en jQuery para que el sitio web se ubicara en una única página y fuera refrescando los datos de forma interactiva y fluida. Pero Angular fue un cambio total de mentalidad, la gestión de componentes es lo que le da la vida en su totalidad y es que, ¿qué sería un desarrollo software sin que presente una modularidad en sus componentes? Angular te lo ofrece, y te da más. Podemos definir todo el dominio de la aplicación dentro de ella y crear los servicios para que estén listos para usarse para contactar con la API. Luego podemos definir el sistema de rutas que deba tener la aplicación junto al resto de componentes que tienen que interactuar con el usuario.

No podría imaginarme un desarrollado planificado, con sus diagramas de requisitos, de base de datos, diseño de componentes y estructuración de vistas sin este framework.

## 2.3 Servicios

### 2.3.1 Sistema de control de versiones

El sistema de control de versiones se realizará gracias a GitHub. GitHub es una plataforma de desarrollo colaborativo para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de aplicaciones.

El software que opera GitHub fue escrito en Ruby on Rails. Desde enero de 2010, GitHub opera bajo el nombre de GitHub, Inc. Anteriormente era conocida como Logical Awesome LLC. El código de los proyectos alojados en GitHub se almacena normalmente de

forma pública.

El 4 de junio de 2018 Microsoft compró GitHub por 7500 millones de dólares. Al principio, el cambio de propietario generó preocupaciones y la salida de algunos proyectos de este repositorio, sin embargo no fueron representativos. GitHub continúa siendo la plataforma más importante de colaboración para proyectos Open Source.

### 2.3.2 Gestión y creación de máquinas virtuales

Este apartado no creí que lo llegaría a utilizar. En la Universidad de Almería tenemos OpenStack un servicio que nos ofrece el poder crear y destruir máquinas virtuales además de poder ubicarlas en una infraestructura de red. El problema que tuve con OpenStack fue la propia configuración de entorno de red de la universidad. Y es que había unos determinados puertos que por mucho que los abriese en la máquina virtual el cortafuegos de la Universidad me impedía acceder a ellos. Como veremos más adelante a lo largo de estas páginas esto no tuvo por qué haber sido un problema pero lo descubrí más tarde. Debido a los problemas anteriormente mencionados con OpenStack me dispuse a buscar las soluciones que me brindaban las distintas “empresas top” del sector.

Las iré evaluando por precio, rendimiento e interfaz. Las he probado todas.

#### Amazon Web Services

En precio se encuentra en un rango medio. Creo que de las tres es la que mejor se enfocaba en lo que buscaba. Disponía de buena documentación y la interfaz era compleja pero fácil de usar. La conectividad y el acceso a las máquinas era rápido pero los planes y servidores eran algo limitados.

#### Microsoft Azure

El precio es alto, bastante a mi parecer. El manejo y creación de base de datos se complica por el plan de precios que presentan que están almacenados en unos tipos de monederos. No creo que sean malas ideas pero sí que están mal implementadas dificultando el aprendizaje inicial. Los servidores y la configuración de los mismos son más limitados que los de Amazon Web Services.


#### Google Cloud

Ofrecen el mejor precio de la competencia. El problema es que la primera vez que entras te encuentras con muchas soluciones que ofrece Google con servicios en la nube. Es una sobreinformación de cosas que realmente no necesitaba para desplegar una simple máquina virtual. Una vez te ubicas que tienes que moverte en el entorno de “Compute Engine” todo se vuelve mucho más fácil. Las máquinas virtuales son fáciles de desplegar y de personalizar y presentan una alta gama de servidores disponibles, al fin y al cabo Google está en todo el mundo.


Mi elección fue Google Cloud. Sin duda para el proyecto iba a ser la mejor opción.


### 3 Estructura y desarrollo del proyecto


El proyecto se divide en varias etapas.





[Pull requests](#) [Issues](#)

 **Inventarium**


 New






 In progress

 Done

[+ New view](#)

 Title

1	 Creación del entorno de látex
2	 Creación del entorno web
3	 Instalación de aplicaciones
+ You can use <span>Ctrl</span> + <span>Space</span> to add an item	



# References

- Biot, M. A. (1962). Mechanics of deformation and acoustic propagation in porous media. *Journal of applied physics*, 33(4), 1482–1498.
- Heinz, M., Carsten, & Hoffmann, J. (2014, March). *The listings package, march 2014*. <http://texdoc.net/texmf-dist/doc/latex/listings/listings.pdf>. Retrieved 12/12/2014, from <http://texdoc.net/texmf-dist/doc/latex/listings/listings.pdf>