



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



**TFG del Grado en Ingeniería  
Informática**

**título del TFG  
Documentación Técnica**



Presentado por nombre alumno  
en Universidad de Burgos — 12 de junio  
de 2023

Tutor: nombre tutor



---

# Índice general

---

|   |            |
|---|------------|
| <b>Índice general</b>                                   | <b>i</b>   |
| <b>Índice de figuras</b>                                | <b>iii</b> |
| <b>Índice de tablas</b>                                 | <b>iv</b>  |
| <b>Apéndice A Plan de Proyecto Software</b>             | <b>1</b>   |
| A.1. Introducción . . . . .                             | 1          |
| A.2. Planificación temporal . . . . .                   | 1          |
| A.3. Estudio de viabilidad . . . . .                    | 1          |
| <b>Apéndice B Especificación de Requisitos</b>          | <b>3</b>   |
| B.1. Introducción . . . . .                             | 3          |
| B.2. Objetivos generales . . . . .                      | 3          |
| B.3. Catalogo de requisitos . . . . .                   | 3          |
| B.4. Especificación de requisitos . . . . .             | 3          |
| <b>Apéndice C Especificación de diseño</b>              | <b>5</b>   |
| C.1. Introducción . . . . .                             | 5          |
| C.2. Diseño de datos . . . . .                          | 5          |
| C.3. Diseño procedimental . . . . .                     | 5          |
| C.4. Diseño arquitectónico . . . . .                    | 5          |
| <b>Apéndice D Documentación técnica de programación</b> | <b>7</b>   |
| D.1. Introducción . . . . .                             | 7          |
| D.2. Estructura de directorios . . . . .                | 7          |
| D.3. Manual del programador . . . . .                   | 7          |

|   |           |
|---|-----------|
| D.4. Compilación, despliegue y ejecución del proyecto . . . . . | 7         |
| D.5. Pruebas del sistema . . . . .                              | 11        |
| <b>Apéndice E Documentación de usuario</b>                      | <b>13</b> |
| E.1. Introducción . . . . .                                     | 13        |
| E.2. Requisitos de usuarios . . . . .                           | 13        |
| E.3. Instalación . . . . .                                      | 13        |
| E.4. Manual del usuario . . . . .                               | 13        |
| <b>Bibliografía</b>   | <b>15</b> |

---

## Índice de figuras

---

|   |   |
|---|---|
| D.1. Página de descarga de Docker Desktop . . . . . | 8 |
| D.2. Instalación de Docker Desktop en Mac . . . . . | 8 |
| D.3. Ventana principal de Docker Desktop . . . . .  | 9 |

---

# Índice de tablas

---

|   |   |
|---|---|
| B.1. CU-1 Nombre del caso de uso. . . . . | 4 |
|---|---|

## *Apéndice A*

---

# **Plan de Proyecto Software**

---

**A.1. Introducción**

**A.2. Planificación temporal**

**A.3. Estudio de viabilidad**

Viabilidad económica

Viabilidad legal





## *Apéndice B*

---

# **Especificación de Requisitos**

---

### **B.1. Introducción**

Una muestra de cómo podría ser una tabla de casos de uso:

### **B.2. Objetivos generales**

### **B.3. Catalogo de requisitos**

### **B.4. Especificación de requisitos**

| CU-1                        | Ejemplo de caso de uso  |
|-----------------------------|---|
| <b>Versión</b>              | 1.0   |
| <b>Autor</b>                | Alumno  |
| <b>Requisitos asociados</b> | RF-xx, RF-xx  |
| <b>Descripción</b>          | La descripción del CU   |
| <b>Precondición</b>         | Precondiciones (podría haber más de una)  |
| <b>Acciones</b>             | <ol style="list-style-type: none"> <li>1. Pasos del CU</li> <li>2. Pasos del CU (añadir tantos como sean necesarios)</li> </ol> |
| <b>Postcondición</b>        | Postcondiciones (podría haber más de una)   |
| <b>Excepciones</b>          | Excepciones   |
| <b>Importancia</b>          | Alta o Media o Baja...  |

Tabla B.1: CU-1 Nombre del caso de uso.

## *Apéndice C*

---

# **Especificación de diseño**

---

- C.1. Introducción
- C.2. Diseño de datos
- C.3. Diseño procedimental
- C.4. Diseño arquitectónico



## Apéndice *D*

---

# Documentación técnica de programación

---

### D.1. Introducción

### D.2. Estructura de directorios

### D.3. Manual del programador

### D.4. Compilación, despliegue y ejecución del proyecto

Para realizar la compilación, despliegue y ejecución del proyecto es imprescindible disponer de **Docker CLI** y **Docker Compose** instalados en el equipo. La mejor forma de instalarlos es mediante **Docker Desktop**, ya que se encarga de instalar en el equipo el *daemon* de Docker, Docker CLI, Docker Compose, y demás dependencias y herramientas que no vamos a utilizar para este proyecto, pero que nos pueden ayudar en el futuro. Docker Desktop es compatible con Windows, macOS y Linux, y soporta tanto arquitecturas x86 como ARM64 (Apple Silicon).

### Instalación de Docker Desktop

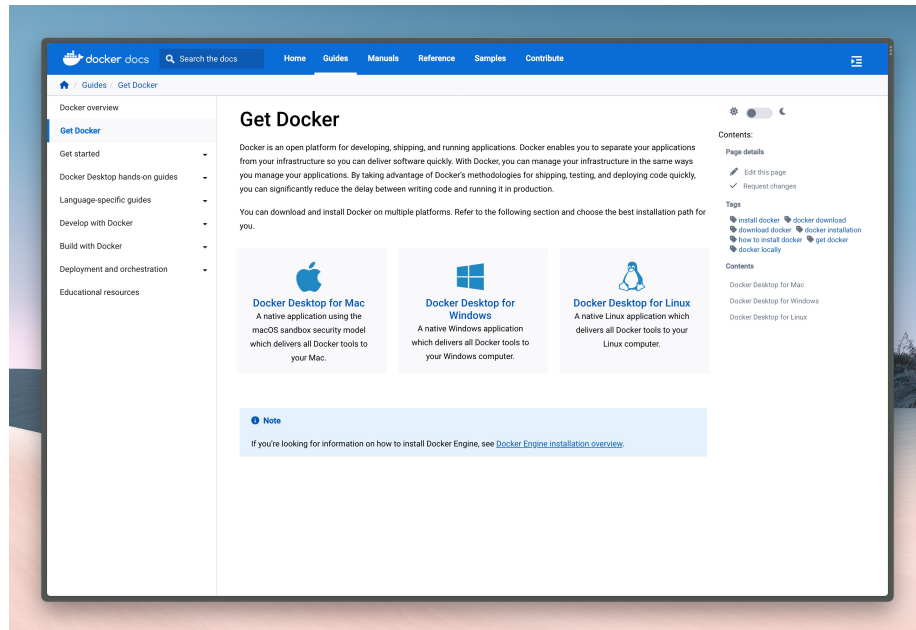


Figura D.1: Página de descarga de Docker Desktop

Para realizar la instalación de Docker Desktop tan sólo debemos dirigirnos a <https://docs.docker.com/get-docker/>, seleccionar la plataforma deseada, y descargar el instalador.

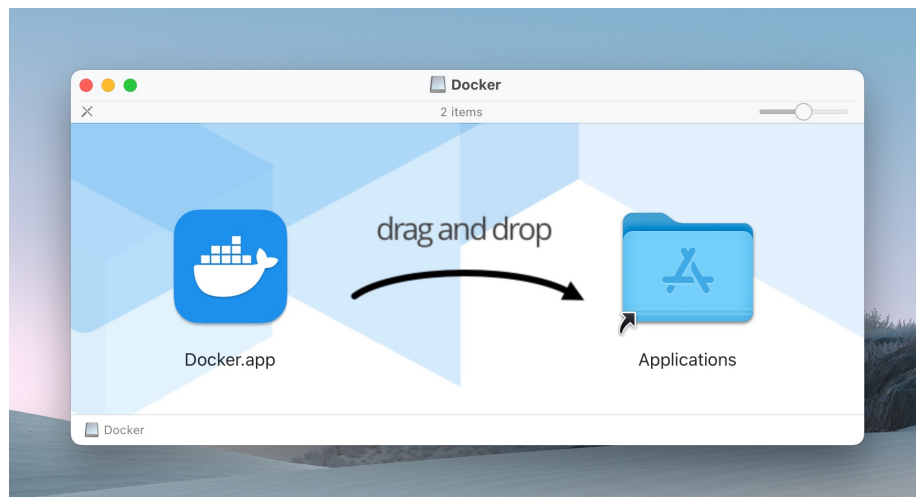


Figura D.2: Instalación de Docker Desktop en Mac

Una vez ejecutemos el instalador (en caso de macOS simplemente se debe arrastrar la aplicación a la carpeta de Aplicaciones), y hayamos seguido todos los pasos hasta finalizar la instalación, nos encontraremos con el panel principal de Docker Desktop.

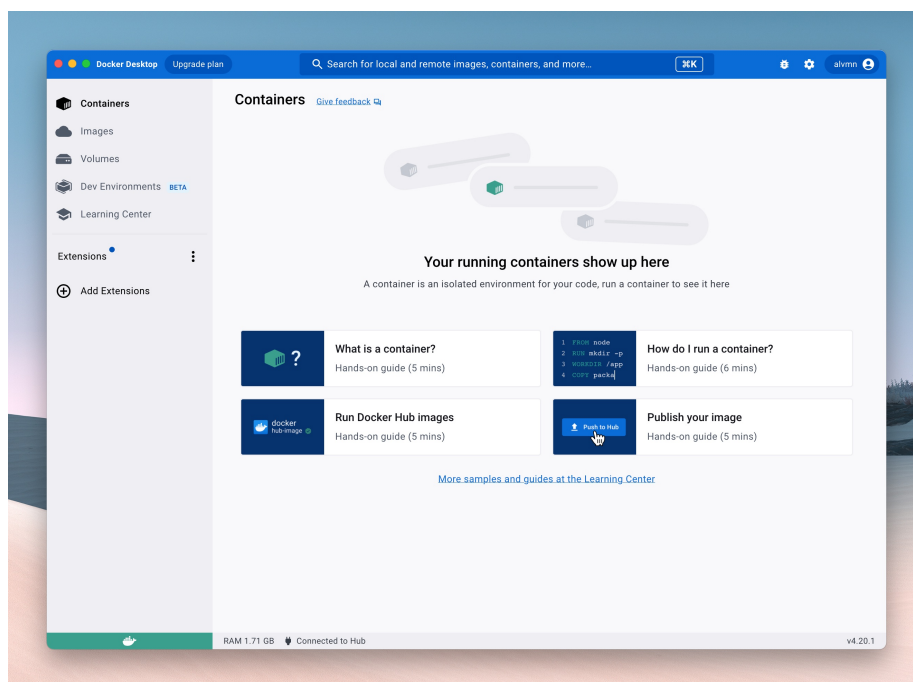


Figura D.3: Ventana principal de Docker Desktop

En esta ventana podemos ver todos los contenedores que están ejecutándose actualmente en el sistema, así como las imágenes descargadas y los volúmenes creados.

## Preparación del entorno de desarrollo

Para hacer más sencillo el cambio de valores en distintos parámetros de la aplicación, como el mapeado de puertos o la gestión de secretos, y facilitar así el despliegue de nuevos entornos o la escalabilidad de los servicios, estos valores se han externalizado en un sólo archivo `.env`, que va a ser un archivo que simplemente va a contener las variables junto a sus valores.

Al ser un fichero que contiene secretos, siguiendo lo que dictan las buenas prácticas no se sincronizará con el repositorio, ya que si se tratase de un entorno de producción o el código se hiciera público, cualquier usuario podría obtener acceso a los datos de los usuarios de la aplicación.

Por lo tanto, antes de desplegar los contenedores, para que la aplicación funcione correctamente el archivo se ha de crear dentro del directorio `/docker-compose`, y se deben de definir las siguientes variables dentro de él:

---

```
# Variables del backend de la parte de los clientes
CLIENT_WEB_DOCKER_PORT=8081
CLIENT_WEB_LOCAL_PORT=8081
CLIENT_RELOAD_DOCKER_PORT=35730
CLIENT_RELOAD_LOCAL_PORT=35730
CLIENT_DEBUG_DOCKER_PORT=5006
CLIENT_DEBUG_LOCAL_PORT=5006

# Variables del backend de la parte de gestion
MANAGE_WEB_DOCKER_PORT=8080
MANAGE_WEB_LOCAL_PORT=8080
MANAGE_RELOAD_DOCKER_PORT=35729
MANAGE_RELOAD_LOCAL_PORT=35729
MANAGE_DEBUG_DOCKER_PORT=5005
MANAGE_DEBUG_LOCAL_PORT=5005

# Variables de la base de datos
DATABASE_HOST=mysqlldb
MYSQL_DATABASE=nutri_db
MYSQL_USER=<MySQL user> # Introduce el nombre del usuario de MySQL
MYSQL_PASSWORD=<MySQL password> # Introduce la pass escogida para
    el usuario creado en el paso anterior
MYSQL_ROOT_PASSWORD=<MySQL root password> # Introduce la pass
    escogida para el usuario root de MySQL
DB_DOCKER_PORT=3306
DB_LOCAL_PORT=3306
```

---

Los valores pueden ser modificados si es necesario, pero se recomienda usar los valores dados (indicando los valores escogidos en `MYSQL_USER`, `MYSQL_PASSWORD` y `MYSQL_ROOT_PASSWORD`). Con estos valores lo que estamos indicando es:

- Los puertos en los que se van a ejecutar cada una de las aplicaciones, que en este caso van a ser el **8080** para la aplicación de gestión y el **8081** para la aplicación de los clientes.
- Los puertos en los que va a estar escuchando el plugin *LiveReload* de Spring, que es el encargado de recargar el servidor en caliente cada vez



que se produzcan cambios en el código, y no tener así que recompilar todo el código cada vez que cambiemos algo. En este caso los puertos van a ser el **35729** para la aplicación de gestión y el **35730** para la aplicación de los clientes.

- Los puertos que se van a usar para permitir conectar un debugger remoto a la aplicación, y poder así hacer debug desde el IDE que usemos para el desarrollo. Los puertos serán el **5005** para la aplicación de gestión y el **5006** para la aplicación de los clientes.
- Todas las variables correspondientes a la base de datos, como el **nombre de la instancia**, **nombre de la base de datos**, **credenciales de los usuarios de MySQL**, y el puerto escogido para la base de datos, que en este caso va a ser el usado por defecto, **3306**.

## D.5. Pruebas del sistema



## *Apéndice E*

---

# **Documentación de usuario**

---

- E.1. Introducción
- E.2. Requisitos de usuarios
- E.3. Instalación
- E.4. Manual del usuario



---

## **Bibliografía**

---