



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 — 3 de junio de 2024
Tutor: nombre tutor

Índice general

Índice general	i
Índice de figuras	iii
Índice de tablas	v
Apéndice A Plan de Proyecto Software	1
A.1. Introducción	1
A.2. Planificación temporal	2
A.3. Estudio de viabilidad	10
Apéndice B Especificación de Requisitos	15
B.1. Introducción	15
B.2. Objetivos generales	15
B.3. Catálogo de requisitos	16
B.4. Especificación de requisitos	17
Apéndice C Especificación de diseño	19
C.1. Introducción	19
C.2. Diseño de datos	19
C.3. Diseño procedimental	19
C.4. Diseño arquitectónico	19
Apéndice D Documentación técnica de programación	21
D.1. Introducción	21
D.2. Estructura de directorios	21
D.3. Manual del programador	21

D.4. Pruebas del sistema	38
Apéndice E Documentación de usuario	39
E.1. Introducción	39
E.2. Requisitos de usuarios	39
E.3. Instalación	39
E.4. Manual del usuario	39
Apéndice F Anexo de sostenibilización curricular	41
F.1. Introducción	41
Bibliografía	43

Índice de figuras

A.1. Sprint 1 - Base del proyecto - Tareas	3
A.2. Sprint 1 - Base del proyecto - Gráfico	4
A.3. Sprint 2 - Evolución del prototipo - Tareas	4
A.4. Sprint 2 - Evolución del prototipo - Gráfico	5
A.5. Sprint 3 - Evolución del prototipo 2 - Tareas	6
A.6. Sprint 3 - Evolución del prototipo 2 - Gráfico	6
A.7. Sprint 4 - Evolución y finalización de la parte del cuestionario - Tareas	7
A.8. Sprint 4 - Evolución y finalización de la parte del cuestionario - Gráfico	7
A.9. Sprint 5 - Evolución y finalización de la parte del cuestionario 2 - Tareas	8
A.10.Sprint 5 - Base del proyecto - Gráfico	9
D.1. Pantalla principal de Docker Desktop.	23
D.2. Versión de docker en ejecución desde CMD de Windows.	23
D.3. Aplicación Git Bash.	24
D.4. Comando Git Clone.	25
D.5. Web de descarga de Spring Tools 4.	26
D.6. Spring Tools Suite 4 inicio.	27
D.7. Spring Tools Suite 4 espacio de trabajo.	27
D.8. Spring Tools Suite 4 - IDE.	28
D.9. Instalación del plugin de Thymeleaf en Spring Tools Suite.	29
D.10.Instalación del plugin de Eclipse Enterprise Java and Web en Spring Tools Suite.	30
D.11.Selección de características de Eclipse Enterprise Java and Web en Spring Tools Suite.	31
D.12.Plugin de Docker en Spring Tools Suite.	32

D.13.Importar proyecto en Spring Tool Suite.	33
D.14.Importación de proyecto Maven en Spring Tool Suite.	34
D.15.Ruta del proyecto indicada en Spring Tool Suite.	35
D.16.Proyecto importado en Spring Tools Suite.	36
D.17.Iniciar la aplicación en Spring Tools Suite.	37
D.18.Aplicación iniciada en Spring Tools Suite.	37

Índice de tablas

A.1. Coste de personal	11
A.2. Coste de recursos	12
A.3. Coste Total	12
B.1. CU-1 Nombre del caso de uso.	18

Apéndice A

Plan de Proyecto Software

A.1. Introducción

La fase de planificación es parte fundamental y necesaria en todos los proyectos.

Esta fase sirve para ir evaluando la viabilidad del proyecto a lo largo de su vida, pudiendo prevenir o adelantarse a posibles inconvenientes que surgan durante la fase de desarrollo. Tener medidas de forma correcta las tareas que se van realizando y su implicación en el total del proyecto es fundamental para un desarrollo sostenible, tanto económica como temporalmente.

Se ha decidido dividir esta fase en dos etapas:

- Planificación temporal.
- Estudio de viabilidad.

La planificación temporal tratará sobre la gestión y aprovechamiento del tiempo, así como la división del proyecto en pequeños hitos, siguiendo una metodología ágil.

El estudio de viabilidad se descompone a su vez en:

- Viabilidad económica: Inversión necesaria para acometer el proyecto y posible obtención de beneficios.
- Viabilidad legal: Todos los programas software conyevan unas implicaciones legales.

A.2. Planificación temporal

Para llevar a cabo una correcta planificación temporal se gestionado el proyecto siguiendo la metodología ágil que plantea Scrum, (poner enlace y bibliografía a Scrum) de forma reducida que el equipo de desarrolladores solo ha tenido un integrante, y las labores de "Scrum Master" y "Product Owner" han recaído en el tutor del TFG, el Sr. Raúl Marticorena.

Para el seguimiento del desarrollo incremental del proyecto software se ha dividido la planificación en una serie de Sprints (enlace y referencia).

Para la visualización de los sprint se tenía intención de incluir los gráficos de "Burndown" pero debido a que han sido unos sprint vivos, donde se han ido incluyendo o desglosando tareas más genéricas en partes más específicas, se ha decidido utilizar los gráficos "Burnup" que representan de forma más real el número total de tareas completadas en cada sprint.

Reunión de Kick-off

Del 12/02/2024 - 1 hora

La reunión de kick-off fue la toma de contacto con el proyecto, donde se trataron los aspectos básicos y se definieron varias herramientas para el desarrollo del mismo.

Temas tratados durante el Kick-off:

- Análisis inicial de la página de risk.real.eu.
- Borradores con la estructura de los cuestionarios.
- Valorar la carga de los cuestionarios en bases de datos o en ficheros de tipo json.
- Definición básica del proyecto:
 1. Web que lea los cuestionarios.
 2. Usuarios separados por sesiones.
 3. Creación de cuestionarios desde la web.
- Diseño responsive: Permite el renderizado para otros dispositivos.
- Documentación y gestión adecuada del proyecto.
- Uso de Spring Boot.
- Github o ZenHub para la gestión de tareas.

Sprint 1 - Base del proyecto

Del 27/02/2024 al 11/03/2024 - 20 horas

Se estima una dedicación total de 20 horas de trabajo. Se cumple con la estimación inicial de horas para completar todas las tareas.

Temas tratados durante el Sprint:

- Uso e integración de zube.io con GitHub.
- Registro en <https://app.riskreal.eu/> para analizar la parte "privada" de la aplicación.
- Schema json:
 - Posibilidad de imagen en cada respuesta
 - Atributo idioma en el cuestionario

Empezar a valorar y tener en cuenta la posibilidad de cuestionarios "Multi-idioma".

- Investigar sobre Thumeleaf y Spring Tools.
- Cargar un json de prueba en la app.
- Representación en una web del json.
- Posible estructura de directorios de la aplicación.
- Abordar la generación dinámica del contenido respecto al json.

Tareas tratadas cerradas a lo largo del sprint:

49	Crear "HoloMundo" web con Spring	49	Active	Spring 1 - Base del proyecto enhancement	Mar 9	Done	C-1	Updated: 16 hours ago	Closed Mar 10
48	Primera app que lee un json y muestre el contenido en una web	48	Active	Spring 1 - Base del proyecto enhancement	Mar 9	Done	C-1	Updated: 16 hours ago	Closed Mar 11
47	Registro en app.riskreal.eu para comprobar las funcionalidades avanzadas	47	Active	Spring 1 - Base del proyecto bug/enhancement/enhancement	Mar 9	Done	C-2	Updated: 16 hours ago	Closed Mar 9
46	Integración de GitHub con Zube.io	46	Active	Spring 1 - Base del proyecto enhancement	Mar 9	Done	C-2	Updated: 16 hours ago	Closed Mar 10
44	Hacer una aproximación inicial de la aplicación actual "https://app.riskreal.eu"	44	Active	Spring 1 - Base del proyecto documentation/enhancement	Feb 26	Done	C-1	Updated: 16 hours ago	Closed Mar 10
45	Importar al repositorio la plantilla de LaTut con la librería del TFG	45	Active	Spring 1 - Base del proyecto bug/enhancement	Feb 26	Done	C-1	Updated: 16 hours ago	Closed Mar 4
41	Definir estructura json para importar cuestionarios	41	Active	Spring 1 - Base del proyecto enhancement	Feb 26	Done	C-2	Updated: 16 hours ago	Closed Mar 9

Figura A.1: Sprint 1 - Base del proyecto - Tareas

Gráfico de Burnup del sprint:

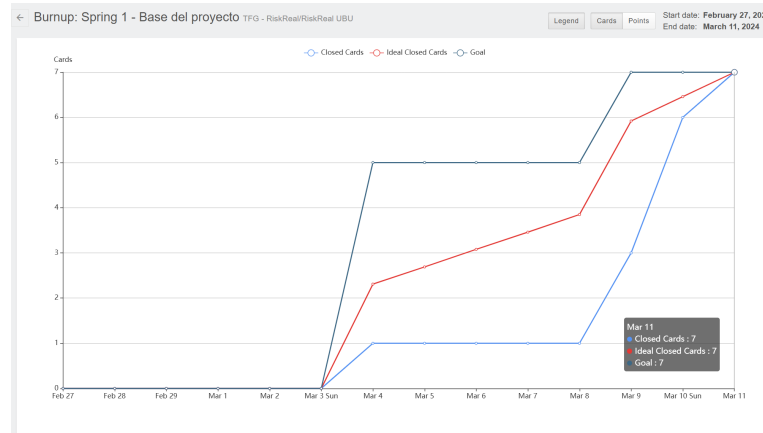


Figura A.2: Sprint 1 - Base del proyecto - Gráfico

Sprint 2 - Evolución del prototipo

Del 11/03/2024 al 25/03/2024 - 20 horas

Se estima una dedicación total de 20 horas de trabajo. Se cumple con la estimación inicial de horas para completar todas las tareas.

Temas tratados durante el Sprint:

- Inclusión de más preguntas y respuestas en el cuestionario.
- Tratar las preguntas como independientes en la web (permitir transición entre preguntas).
- Recoger y mostrar valoraciones al terminar el cuestionario.
- Concepto de sesión web, para que no se mezclen datos de 2 usuarios.
- Primer acercamiento a un framework de estilo, bootstrap o similar.

Tareas tratadas cerradas a lo largo del sprint:

#15	Problema al ir a preguntas que no existen	#15	Activar	Sprint 2 - Evolución del prototipo	Mar 24	Done	C10	Updated 10 hours ago	Closed Mar 24
#13	Mezcladura de preguntas	#13	Activar	Sprint 2 - Evolución del prototipo	Mar 11	Done	C10	Updated 10 hours ago	Closed Mar 24
#12	Independencia de preguntas	#12	Activar	Sprint 2 - Evolución del prototipo	Mar 11	Done	C11	Updated 10 hours ago	Closed Mar 24
#11	Independencia de preguntas del cuestionario	#11	Activar	Sprint 2 - Evolución del prototipo	Mar 11	Done	C11	Updated 10 hours ago	Closed Mar 24
#10	Procesar cuestionarios con más preguntas	#10	Activar	Sprint 2 - Evolución del prototipo	Mar 11	Done	C11	Updated 10 hours ago	Closed Mar 24

Figura A.3: Sprint 2 - Evolución del prototipo - Tareas

Gráfico de Burnup del sprint:

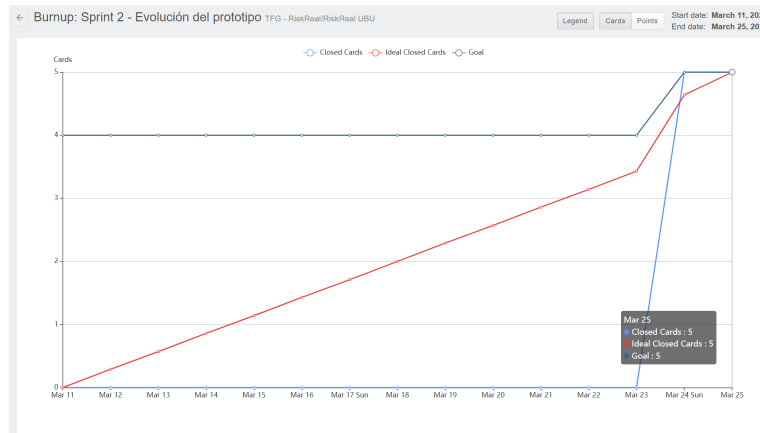


Figura A.4: Sprint 2 - Evolución del prototipo - Gráfico

Sprint 3 - Evolución del prototipo 2

Del 25/03/2024 al 24/04/2024 - 60 horas

Se estima una dedicación total de 20 horas de trabajo. Tras encontrar problemas diversos problemas a la hora de integrar la seguridad de Spring Security referencia.^{en} la app, no solo no se cumple con la planificación inicial sino que se extiende el mismo spring dos semanas más, ampliando la carga de trabajo con más tareas y teniendo que emplear finalmente 60 horas de trabajo para cumplir con casi todas las tareas. La tarea que queda pendiente se solucionará en el sprint 5.

Temas tratados durante el Sprint:

- Independizar las sesiones de los usuarios.
- Integración de la seguridad en la aplicación.
- Integrar la funcionalidad de envío de correo electrónico desde la web.
- Enviar la información del cuestionario con el método POST para que no vayan los datos en claro.
- Internacionalización de los distintos menús de la app web.
- Integración de framework CSS para utilizar un diseño responsive.^{en} la web.
- Recoger datos de puntuación de las preguntas.

- Mostrar un score tras finalizar el cuestionario.

Tareas tratadas cerradas a lo largo del sprint:

#35	Integrar Seguridad en la Aplicación	#34	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Apr 17	Closed	C1 1	Updated: 10 hours ago	Closed Apr 17
#27	Integrar envío de correo en Sprint Boot	#30	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Apr 11	Closed	C1 1	Updated: 10 hours ago	Closed Apr 15
#24	Enviar el nombre del cuestionario dentro de la sesión en modo "Pilot" para que no se	#34	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Apr 11	Closed	C1 1	Updated: 10 hours ago	Closed Apr 13
#20	Interacción con Spring	#30	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Mar 26	Closed	C1 6	Updated: 10 hours ago	Closed Apr 13
#19	Mostrar un "score" final tras completar el cuestionario	#19	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Mar 26	Closed	C1 1	Updated: 10 hours ago	Closed Apr 21
#18	Recoger datos de la puntuación de las preguntas	#16	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Mar 26	Closed	C1 1	Updated: 10 hours ago	Closed Apr 21
#17	Cuestionario Real	#17	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Mar 24	Closed	C1 1	Updated: 10 hours ago	Closed Apr 7
#16	Integración de algún framework de CSS para hacer el diseño de la web "responsive"	#16	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Mar 24	Closed	C1 2	Updated: a few seconds ago	Closed Apr 7
#14	Sesiones independientes al realizar los cuestionarios	#14	Activar	# Sprint 3 - Evolución del prototipo 2 entonces	Mar 11	Closed	C1 2	Updated: 10 hours ago	Closed Apr 13

Figura A.5: Sprint 3 - Evolución del prototipo 2 - Tareas

Gráfico de Burnup del sprint:



Figura A.6: Sprint 3 - Evolución del prototipo 2 - Gráfico

Sprint 4 - Evolución y finalización de la parte del cuestionario

Del 24/04/2024 al 06/05/2024 - 23 horas

Se estima una duración inicial de 20 horas de trabajo, aunque finalmente se emplean 23 horas para cumplir con 4 de las 6 tareas previstas. La tarea de implementar el registro de usuarios a través de una base de datos llevó más tiempo del esperado, de ahí la necesidad de horas adicionales y el no poder cumplir con todas las tareas previstas. Las 2 tareas pendientes se completarán en el sprint 5.

Temas tratados durante el Sprint:

- Gestión de usuarios y roles.
- Mostrar información del usuario logueado.
- Creación de un formulario de registro de usuarios.
- Almacenar los datos de score en un fichero, para futuros análisis.
- Introducción al uso de roles para acceder a distintas partes de la aplicación.

Tareas tratadas cerradas a lo largo del sprint:

#43	Separación por ROLES	#42	Active	# Spring 4 - Evolución y finalización de la parte del cuestionario entramente	May 1	Done	C2	Updated: 16 hours ago	Closed: May 6
#40	Mostrar información del usuario logueado	#39	Active	# Spring 4 - Evolución y finalización de la parte del cuestionario entramente	May 1	Done	C1	Updated: 16 hours ago	Closed: May 1
#37	Registro de Usuarios	#36	Active	# Spring 4 - Evolución y finalización de la parte del cuestionario entramente	Apr 17	Done	C2	Updated: 16 hours ago	Closed: May 6
#36	Gestión de Usuarios y Roles	#35	Active	# Spring 4 - Evolución y finalización de la parte del cuestionario entramente	Apr 17	Done	C1	Updated: 16 hours ago	Closed: May 6

Figura A.7: Sprint 4 - Evolución y finalización de la parte del cuestionario - Tareas

Gráfico de Burnup del sprint:

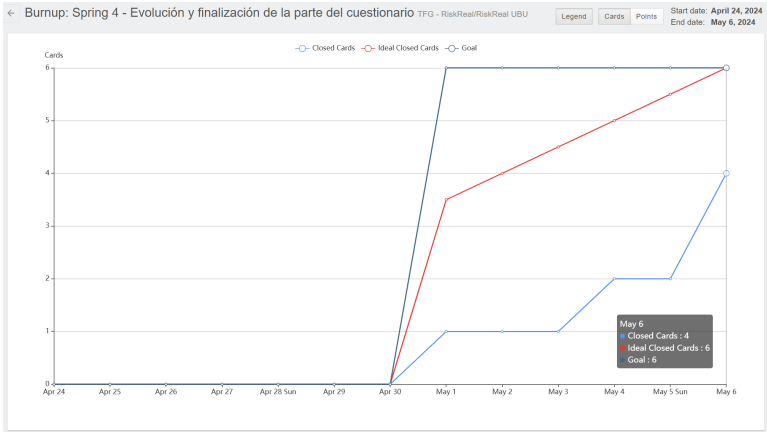


Figura A.8: Sprint 4 - Evolución y finalización de la parte del cuestionario - Gráfico

Sprint 5 - Evolución y finalización de la parte del cuestionario 2

Del 06/05/2024 al 22/05/2024 - 40 horas

Se estima una dedicación total de 40 horas de trabajo. Se cumple con la estimación inicial de horas para completar todas las tareas. En este sprint se han completado las tareas que quedaron pendientes del sprint 3 y 4, por eso la carga de trabajo y de horas ha sido mayor que en los anteriores.

Temas tratados durante el Sprint:

- Investigar sitio para alojar un test de la aplicación → <https://www.heroku.com/>. Guardar el score por pregunta y el total.
- Descarga de resultados en CSV.
- Array de imágenes en las preguntas.
- Diseño de datos bbdd y json. Código de la aplicación.
- Revisar información para incluir en el arquitectónico.
- Revisar diagramas para incluir en el diseño procedimental.

Tareas tratadas cerradas a lo largo del sprint:

#58	Contraseñas encriptadas para los usuarios	#58	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 20	Done	0/0	Updated 16 hours ago	Closed May 20
#56	Filtrado de opciones disponibles en función de usuario registrado o invitado	#56	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 20	Done	0/0	Updated 16 hours ago	Closed May 20
#55	Selección de idioma en la NavBar	#54	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 20	Done	0/0	Updated 16 hours ago	Closed May 20
#54	Campo "Tipo" en el Quiz para diferenciar	#53	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 14	Done	0/0	Updated 16 hours ago	Closed May 14
#53	Descarga de resultados en CSV	#52	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 14	Done	0/0	Updated 16 hours ago	Closed May 22
#52	Array de imágenes en las preguntas	#51	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 14	Done	0/0	Updated 16 hours ago	Closed May 14
#42	Almacenar los datos de "Score" en un fichero para futuros análisis	#41	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 1	Done	0/1	Updated 16 hours ago	Closed May 22
#41	Formulario de registro en la aplicación similar a de appRealReal	#40	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	May 1	Done	0/0	Updated 16 hours ago	Closed May 20
#38	Servicio de restauración de Contraseñas	#37	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	Apr 17	Done	0/0	Updated 16 hours ago	Closed May 21
#25	Definir los requisitos para el registro de usuario	#25	Archive	Spring 5 - Evaluación y finalización de la parte del cuestionario	12	Apr 11	Done	0/0	Updated 16 hours ago	Closed May 21

Figura A.9: Sprint 5 - Evolución y finalización de la parte del cuestionario 2 - Tareas

Gráfico de Burnup del sprint:

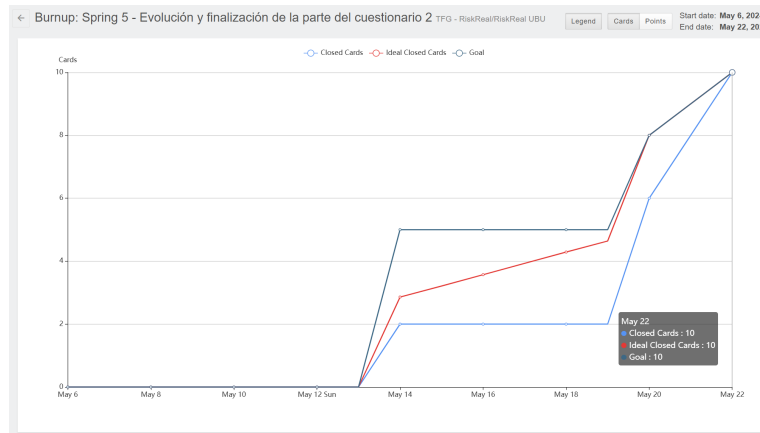


Figura A.10: Sprint 5 - Base del proyecto - Gráfico

Sprint 6 - Carga de cuestionarios y documentación

Del 22/05/2024 al 30/05/2024 - 25 horas

Temas tratados durante el Sprint:

- Carga de cuestionarios desde una carpeta definida en configuración.
- Carga de cuestionarios de forma "viva" desde la propia web de la aplicación.
- Completar la memoria del proyecto.
- Completar los anexos del proyecto.
- Documentar de forma adecuada y en inglés el código de java.

Tareas tratadas cerradas a lo largo del sprint:

Gráfico de Burnup del sprint:

Sprint 7 -

Del 30/05/2024 al 03/06/2024 - 20 horas

Temas tratados durante el Sprint:

- Lista de tareas.

Tareas tratadas cerradas a lo largo del sprint:

Gráfico de Burnup del sprint:

Sprint 8 -

Del 03/05/2024 al 10/06/2024 - 30 horas

Temas tratados durante el Sprint:

- Lista de tareas.

Tareas tratadas cerradas a lo largo del sprint:

Gráfico de Burnup del sprint:

A.3. Estudio de viabilidad

Viabilidad económica

A lo largo de este apartado se van a simular los costes/beneficios del proyecto, si hubiese sido llevado a cabo en una empresa real.

Costes

Los costes se han desglosado en dos categorías:

- Costes de personal
- Costes de recursos

Costes de personal

Se ha simulado la contratación de un programador web durante los 5 meses que ha durado el proyecto.

En la siguiente tabla se desglosan los datos de retenciones referenciaz tributación de la empresa a la seguridad social referencia"para hacer la estimación.

Concepto	Coste
Salario mensual bruto	2.000,00 €
Retención IRPF (13,55 %)	217,00 €
Seguridad social (36,25 %)	1.450,00 €
Salario mensual neto	1.602,00 €
Coste total para la empresa	3.323,00 €
Total 5 meses	16.615,00 €

Tabla A.1: Coste de personal

Costes de recursos

Solo se ha tenido en cuenta el material necesario para el nuevo empleado, no se considera ni el alquiler del espacio, ni el acceso a internet ni el mobiliario, ya que gracias al trabajo híbrido (presencial + teletrabajo) todo esto no ha supuesto ningún gasto adicional para la empresa.

A continuación se detalla de forma breve el material hardware que se ha tenido en cuenta:

- Ordenador portátil: Dell Inspiron 14 Plus 1.098,99 € - 908,26 € sin IVA
- Monitor adicional: Dell S Series S2721HS 27"LED IPS FullHD 229,36€ - 189,55 € SIN IVA
- Combo teclado y ratón: Logitech MK235 39,99 € - 33,05€ SIN IVA
- Licencia Windows 11 pro: Incluida en el precio del portátil

Debido a que las empresas se deducen el IVA, este no se incluye en el cálculo del coste.

Concepto	Coste
Ordenador portátil	908,26 €
Monitor adicional	189,55 €
Combo teclado y ratón	33,05€
S.O. Windows 11 Pro	- €
Total	1.130,86 €

Tabla A.2: Coste de recursos

Costes Total

La siguiente tabla recoge el coste total que supondría a la empresa la realización del proyecto.

Concepto	Coste
Coste de personal	16.615,00 €
Coste de material	1.130,86 €
Total	17.745,86 €

Tabla A.3: Coste Total

Beneficios

La obtención de beneficio con el producto software se realizaría de la siguiente forma.

1. Venta como servicio de pago mensual por cuestionario: Se ofrece el acceso a la plataforma y se tiene que abonar una cuota mensual de 5€ por cuestionario que se incluya, con 1 usuario administrador y 10 usuarios normales de base. Se pagaría un suplemento de 5€ por cada usuario administrador necesario y otros 5€ por cada paquete de 10 usuarios normales extra.
2. Solución *standalone* con soporte y actualizaciones: Se ofrecería un docker en el que estaría corriendo la aplicación y se tendrían diferentes tarifas en nivel del soporte y actualizaciones que se contratase.

Viabilidad legal

Esta sección tratará sobre el tema de la licencia, tanto del *software* como de la documentación.

Se han recopilado las siguientes tecnologías junto con la licencia que aplica en cada una:

- Spring Boot, Spring Data JPA, Spring Security, Spring Web, Thymeleaf: **Apache License 2.0**
- Docker Engine: **Apache License 2.0**
- Eclipse IDE: **Eclipse Public License 2.0**
- Bootstrap: **MIT License**
- MySQL: **GNU General Public License (GPL) 2.0**

Del listado anterior, la licencia más restrictiva es la GPL de MySQL, por lo tanto, para la distribución del software se podrían dar 2 escenarios:

1. Ofrecer la aplicación sin la parte de MySQL y establecerlo como requisito para funcionar pero sin incluirlo.
2. Ofrecer la aplicación como un todo, con MySQL integrado.

En el primer caso, se podría optar por una licencia MIT , que es la más permisiva y engloba a todas las demás. El uso de esta licencia permitiría que cualquiera pueda añadir mejoras o modificar el código de la aplicación.

En el segundo caso, se optaría por la licencia GPL 2.0, teniendo en cuenta que cualquier otra versión de la aplicación también sería considerado software libre.

Apéndice *B*

Especificación de Requisitos

B.1. Introducción

En este anexo se recoge la especificación de requisitos de la aplicación web que se ha desarrollado.

Este documento tiene el objetivo doble de servir de *contrato* con el cliente y como base de documentación para el análisis de la aplicación.

B.2. Objetivos generales

El proyecto tiene los siguientes objetivos generales:

- Desarrollar una aplicación web que permita la realización de cuestionarios para evaluar *Soft Skills*.
- Permitir la carga de usuarios tanto desde una carpeta del sistema como desde la interfaz web.
- Almacenar los resultados de los cuestionarios de forma anónima con fines estadísticos para los creadores.
- La descarga completa de los resultados en un documento correctamente estructurado y que sea sencillo de modelar.
- La aplicación web tiene que tener un diseño *responsive* que extienda su funcionalidad a diferentes dispositivos.

- La aplicación tiene que tener ser multi-idioma, para poder llegar a más usuarios.

B.3. Catálogo de requisitos

Se van a enumerar los requisitos separándolos en dos categorías: requisitos funcionales y no funcionales.

Requisitos funcionales

- **RF-1 Gestión de cuestionarios:** La aplicación web tiene que ser capaz de gestionar cuestionarios.
 - **RF-1.1 Carga de cuestionario:** Se tienen que poder cargar nuevos cuestionarios en la aplicación web.
 - **RF-1.2 Realización de cuestionario:** Los usuarios deben poder realizar los cuestionarios disponibles.
- **RF-2 Gestión de usuarios:** La aplicación web tiene que ser capaz de gestionar usuarios.
 - **RF-2.1 Registro de usuario:** Los usuarios deben poder registrarse en la aplicación.
 - **RF-2.2 Modificación de datos:** Los usuarios deben poder editar sus datos.
 - **RF-2.3 Eliminación de usuario:** Los usuarios tienen que poder darse de baja en la aplicación.
- **RF-3 Gestión de resultados:** La aplicación web tiene que permitir la gestión de los resultados obtenidos.
 - **RF-3.1 Almacenamiento de resultados:** Los resultados se tienen que almacenar de forma persistente en la aplicación.
 - **RF-3.2 Descarga de resultados:** Los resultados se tienen que poder descargar de la aplicación.
- **RF-4 Diseño internacional:** La aplicación web tiene que ser capaz de adaptarse a varios idiomas.
 - **RF-4.1 Selección de idioma:** La aplicación debe permitir la selección del idioma.

- **RF-4.2 Idioma de los cuestionarios:** La aplicación debe permitir la carga de cuestionarios en varios idiomas.
- **RF-5 Diseño responsable:** La aplicación tiene que ser capaz de adaptarse a diferentes dispositivos.

B.4. Especificación de requisitos

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

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

En el anexo actual se va a describir toda la documentación técnica, así como la estructura de directorios que se ha seguido para almacenar el proyecto completo. Una de las partes más importantes es la sección de *Manual del Programador* ya que en ella se describen los pasos necesarios para instalar el entorno de desarrollo.

D.2. Estructura de directorios

El repositorio del proyecto está formado por los siguientes directorios:

D.3. Manual del programador

El manual del programador es la guía *paso a paso* de las diferentes herramientas, configuraciones y peculiaridades que se deben seguir para poder montar el entorno de desarrollo, trabajar con el código fuente o recompilar una nueva versión de la aplicación.

Las imágenes que se pueden ver en los siguientes pasos han sido tomadas desde un portátil con el sistema operativo Windows 11 y con las versiones aquí indicadas de las diferentes aplicaciones, por lo tanto, en distintas versiones el contenido o la ubicación de algunas opciones pueden estar en diferentes apartados.

Herramientas necesarias

Para poder desplegar el entorno de desarrollo se necesita instalar las siguientes aplicaciones:

- Java SE JDK 17.
- Docker.
- Git.
- Spring Tools 4 for Eclipse.

A continuación, se explica cómo instalar y configurar cada uno de ellos de forma correcta.

Java SE JDK 17

Es el lenguaje de programación utilizado para la realización de la aplicación web. Se debe acceder al siguiente enlace [3], seleccionar el sistema operativo y arquitectura correspondiente y seguir el asistente de instalación.

Docker

En el momento actual en el que se encuentra la informática y el interés por la optimización de recursos, Docker se ha posicionado como la plataforma principal de *microservicios*.

Docker Desktop se puede descargar desde el enlace [1]. Tiene versión para windows, linux y mac, por lo que hay que elegir el sistema operativo correcto y seguir los pasos del asistente de instalación.

Una vez instalado, la aplicación sería similar a la siguiente:

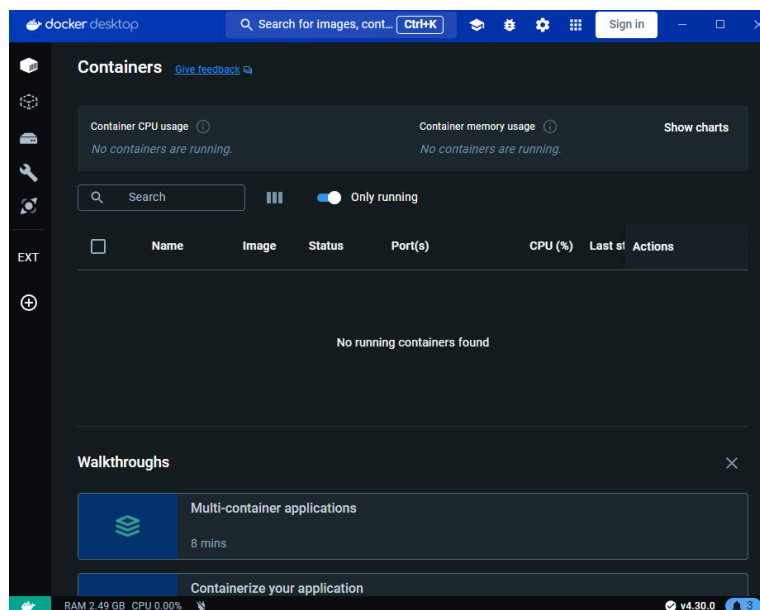


Figura D.1: Pantalla principal de Docker Desktop.

Podemos cerrar la aplicación sin problema porque se queda en ejecución en segundo plano, pero para confirmarlo, podemos abrir *el símbolo del sistema* - *cmd* o una consola de powershell y ejecutar el comando **docker version**, donde deberíamos obtener una imagen similar a la siguiente:

```
Simbolo del sistema
C:\Users\Daniel>docker version

Client:
 Cloud integration: v1.0.35+desktop.13
 Version: 26.1.1
 API version: 1.45
 Go version: go1.21.9
 Git commit: 4c78afa
 Built: Tue Apr 30 11:48:43 2024
 OS/Arch: windows/amd64
 Context: default

Server: Docker Desktop 4.30.0 (149282)
 Engine:
  Version: 26.1.1
  API version: 1.45 (minimum version 1.24)
  Go version: go1.21.9
  Git commit: ac2de55
  Built: Tue Apr 30 11:48:28 2024
  OS/Arch: linux/amd64
  Experimental: false
 containerd:
  Version: 1.6.31
  GitCommit: e377cd56a71523148ca6ae87e30244719194a521
 runc:
  Version: 1.1.12
  GitCommit: v1.1.12-0-g51d5e94
 docker-init:
  Version: 0.19.0
  GitCommit: de48ad9
```

Figura D.2: Versión de docker en ejecución desde CMD de Windows.

Lo importante de la imagen anterior es que aparezca la información de **Server: Docker Desktop**, si no apareciese, abrir la aplicación de Docker Desktop y comprobar que ahora si tenemos la información del servidor.

Git

Git es la herramienta que nos permite recuperar todo el proyecto desde el repositorio web [5]. Nos descargamos Git desde el siguiente enlace [2]. Seleccionamos la versión que se corresponda con nuestro sistema operativo y seguimos los pasos del asistente de instalación.

Una vez instalado, vamos a utilizar la herramienta **Git Bash**

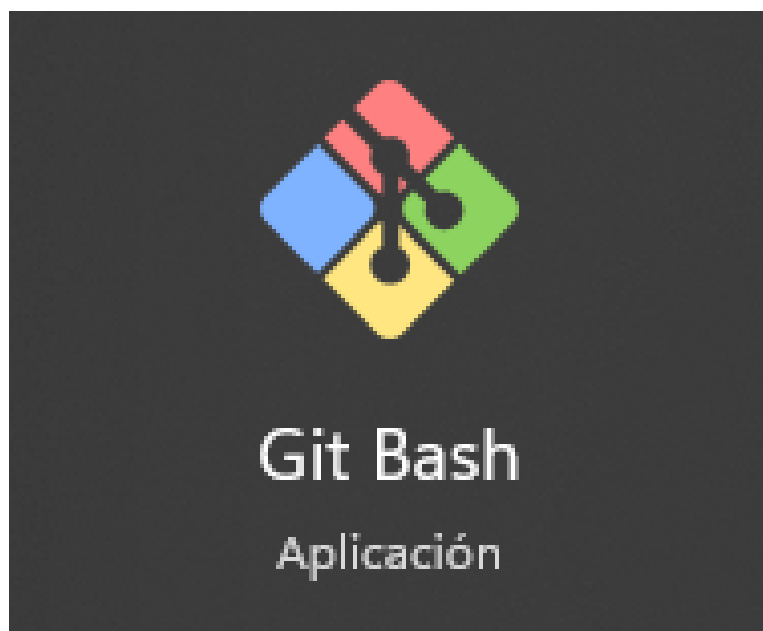
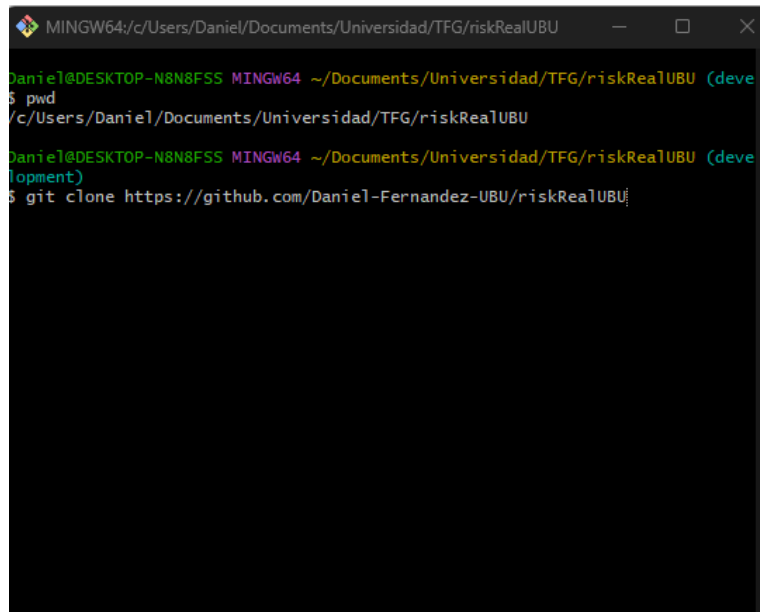


Figura D.3: Aplicación Git Bash.

Navegamos por los directorios hasta el directorio en el que queramos replicar el proyecto y a continuación, escribimos `git clone https://github.com/Daniel-Fernandez-UBU/riskRealUBU`



```
MINGW64:/c:/Users/Daniel/Documents/Universidad/TFG/riskRealUBU
Daniel@DESKTOP-N8N8F55 MINGW64 ~/Documents/Universidad/TFG/riskRealUBU (development)
$ pwd
/c:/Users/Daniel/Documents/Universidad/TFG/riskRealUBU
Daniel@DESKTOP-N8N8F55 MINGW64 ~/Documents/Universidad/TFG/riskRealUBU (development)
$ git clone https://github.com/Daniel-Fernandez-UBU/riskRealUBU
```

Figura D.4: Comando Git Clone.

Spring Tools 4 for Eclipse

Spring Tools 4 for Eclipse es la herramienta desde la que se ha construido el proyecto de Spring al completo.

Se puede obtener desde el siguiente enlace [\[4\]](#)

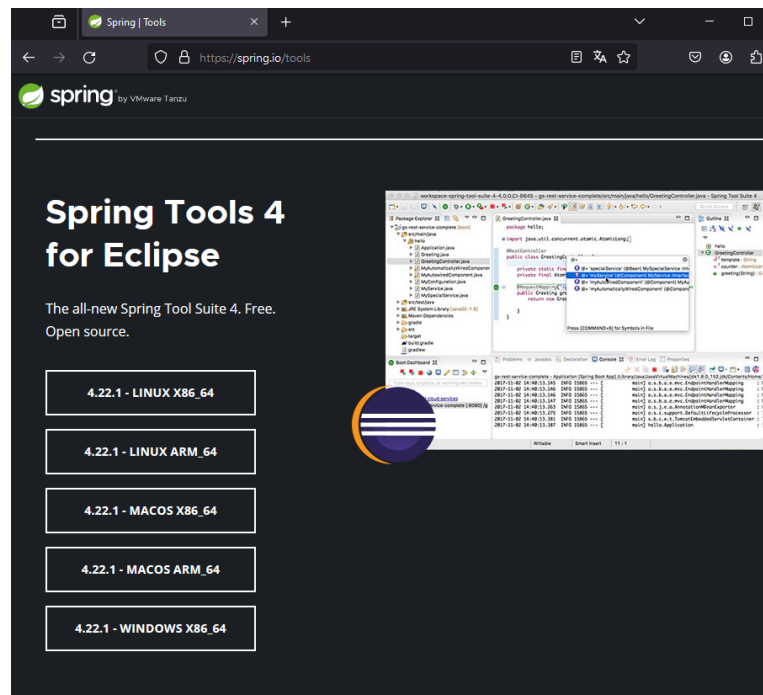


Figura D.5: Web de descarga de Spring Tools 4.

En el caso de Windows, el fichero descargado es un *.jar*.

1. Hacemos doble click sobre el fichero *.jar* descargado, para que se inicie la extracción automática del contenido.
2. En la carpeta en la que esté el fichero *.jar* se generará una nueva carpeta con nombre **sts-4.22.1.RELEASE**, o similar, dependiendo de la versión de la aplicación en el momento de la descarga.
3. Movemos esa carpeta a la ubicación que nos interese, pues el programa de Spring Tools es *portable*.
4. Accedemos a la carpeta y ejecutamos **SpringToolSuite4.exe**

Tras unos instantes en los que aparece la siguiente ventana:



Figura D.6: Spring Tools Suite 4 inicio.

Nos pedirá que indiquemos la ubicación de nuestro espacio de trabajo, donde se guardarán por defecto todos los proyectos que creemos desde la aplicación.

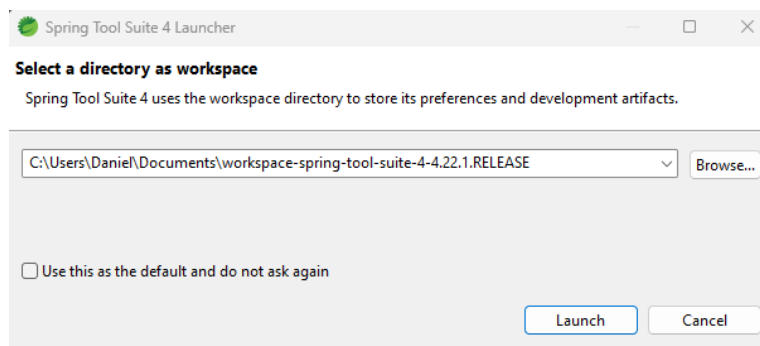


Figura D.7: Spring Tools Suite 4 espacio de trabajo.

Tras indicar nuestra ubicación preferida, pulsamos en *Launch*.

Tras unos instantes en los que carga la aplicación, se nos abre la aplicación:

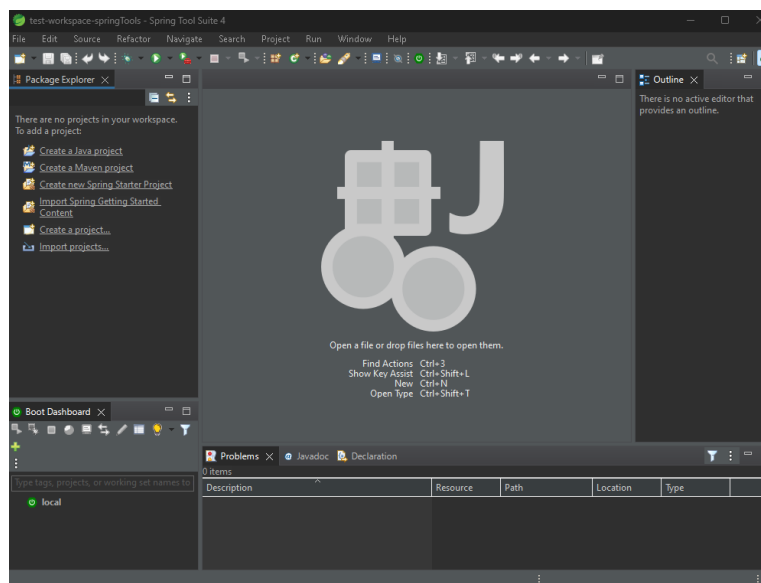


Figura D.8: Spring Tools Suite 4 - IDE.

Antes de importar el proyecto, vamos a instalar algunos plugins que pueden resultar de utilidad.

El proceso para la instalación de los plugins es el mismo, solo cambia el plugin a buscar.

Pasos a seguir para la instalación de cualquier plugin:

1. Pulsamos en *Help* → *Eclipse Marketplace...*
2. Introducimos el nombre del plugin que queremos buscar.
3. Pulsamos en *Go* o le damos al *Intro* en el teclado.
4. Pulsamos en *Install*.
5. Seguimos los pasos, aceptando las diferentes ventanas de confirmación que nos aparecen.
6. Tras terminarse la instalación del plugin, nos pide reiniciar Spring Tools Suite 4. **Consejo: Instalar todos los plugins y al final reiniciar la aplicación**

Thymeleaf

Siguiendo los pasos anteriores, obtendríamos la siguiente ventana, donde ya solo quedaría darle a *Install* y seguir los pasos del asistente de instalación.

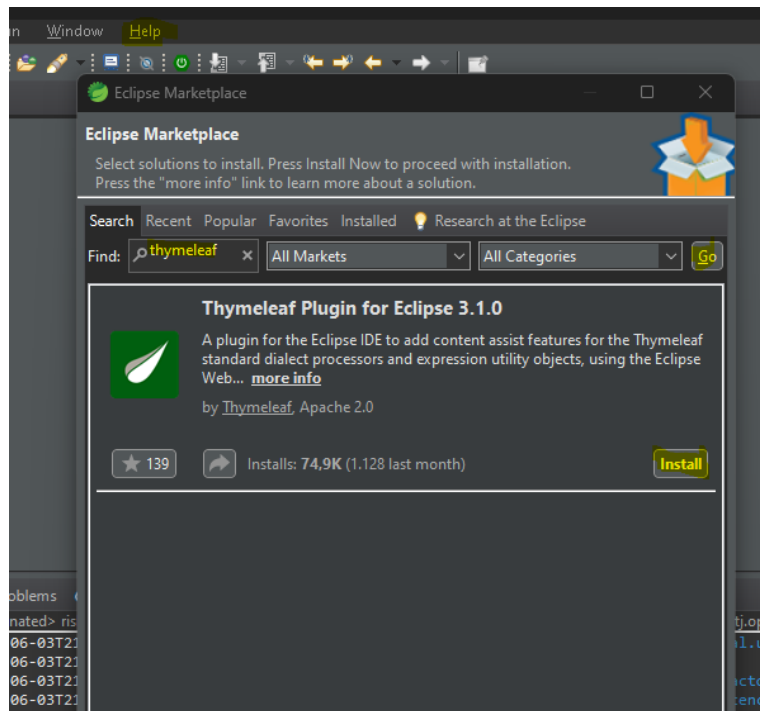


Figura D.9: Instalación del plugin de Thymeleaf en Spring Tools Suite.

Eclipse Web Developer

Instalamos la versión *Eclipse Enterprise Java and Web Developer Tools 3.33* que incluye los editores de html, json y css entre otros.

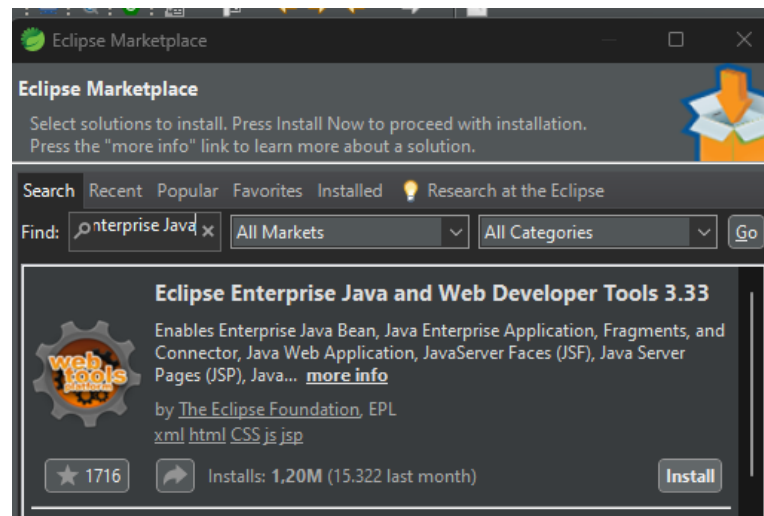


Figura D.10: Instalación del plugin de Eclipse Enterprise Java and Web en Spring Tools Suite.

Seleccionamos las características que vamos a instalar:

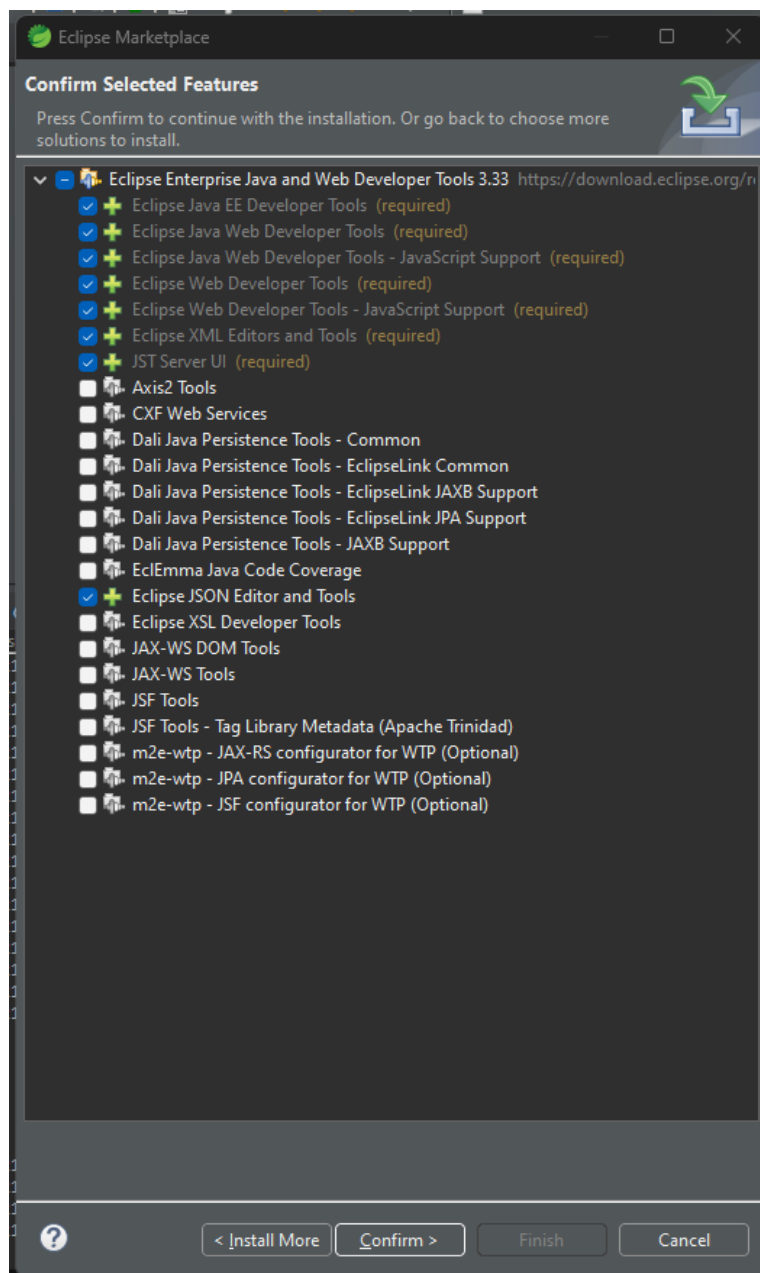


Figura D.11: Selección de características de Eclipse Enterprise Java and Web en Spring Tools Suite.

Eclipse Docker Tooling

Este plugin viene instalado por defecto en la última versión disponible de Spring Tools 4 for Eclipse, pero si no apareciese en la sección de *Installed* en *Eclipse Marketplace*, también sería necesario instalarlo.

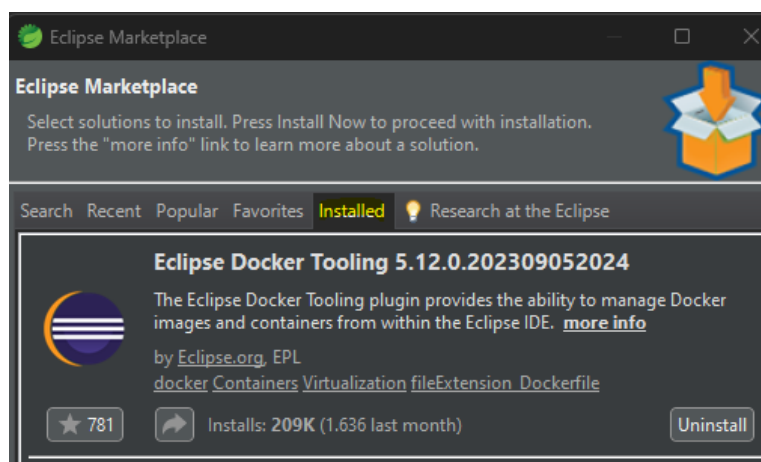


Figura D.12: Plugin de Docker en Spring Tools Suite.

Importación de nuestro proyecto

Una vez hayamos finalizado con la instalación de los plugins anteriores, procederemos a importar nuestro proyecto.

En primer lugar seleccionamos **File** → **Import...**

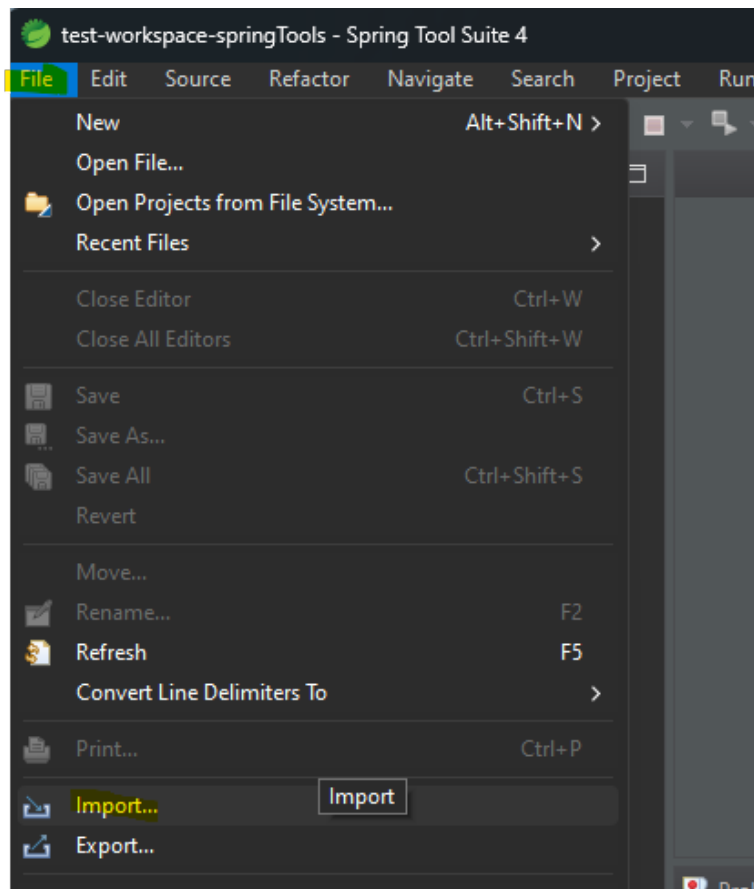


Figura D.13: Importar proyecto en Spring Tool Suite.

En la nueva ventana que se muestra, navegamos hasta **Maven** → **Existing Maven Projects** → **Next**.

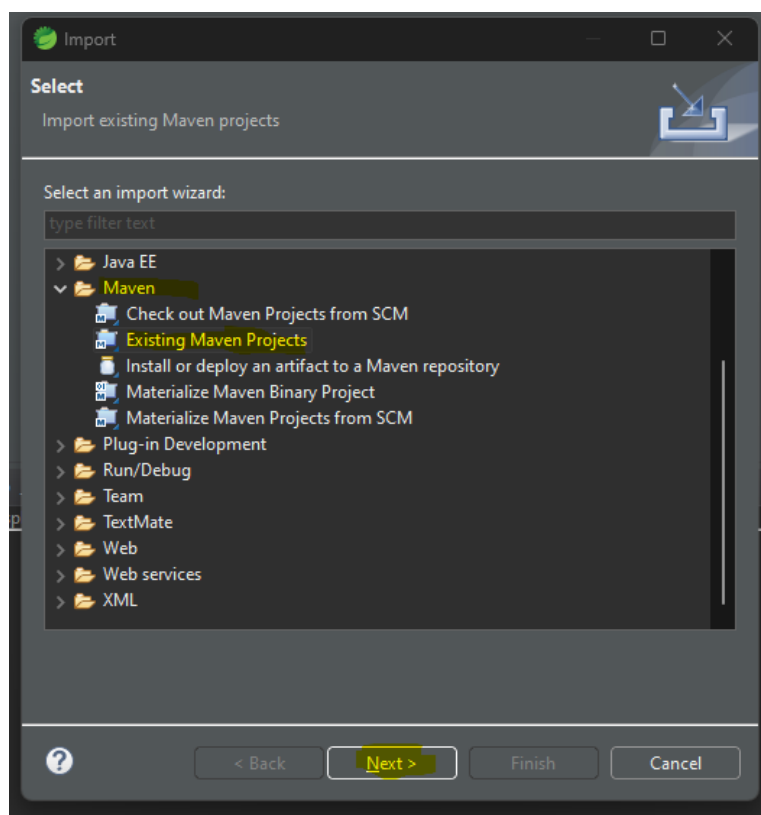


Figura D.14: Importación de proyecto Maven en Spring Tool Suite.

A continuación, nos pedirá que busquemos la ruta donde tenemos el proyecto que queremos importar. **Tenemos que ir a la ruta sobre la que hayamos ejecutado el *git clone* del paso previo.**

Accedemos a la ruta donde hemos hecho el clon del repositorio de Git, y dentro de la carpeta del proyecto, *riskRealUBU*, accedemos a *javaProject* → *riskRealApp*.

Si hemos seleccionado la ruta adecuada, deberá quedar de esta forma la importación, habiendo reconocido el fichero **pom.xml** de nuestro proyecto y a la espera de que pulsemos en *Finish*.

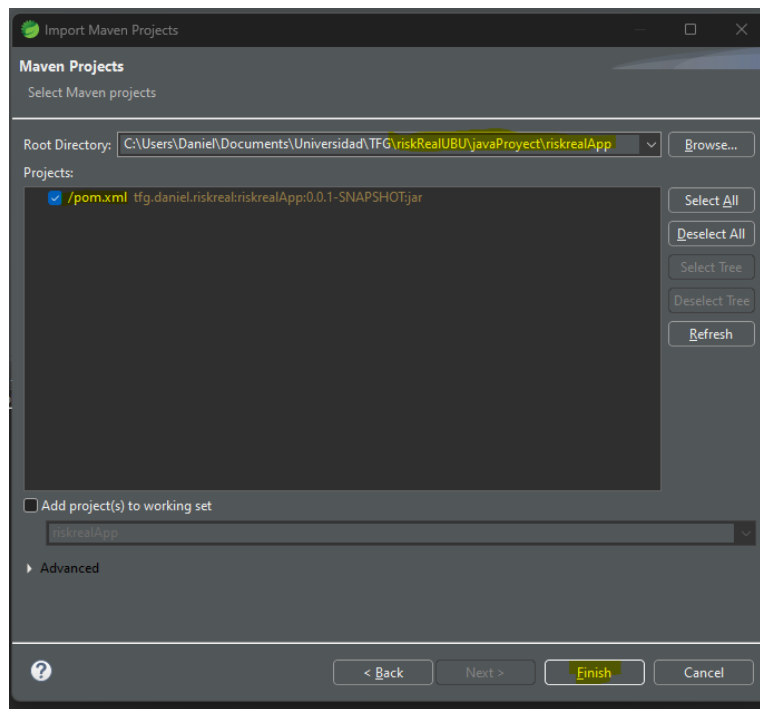


Figura D.15: Ruta del proyecto indicada en Spring Tool Suite.

Tras unos instantes, ya nos aparecería el proyecto correctamente cargado en nuestra aplicación.

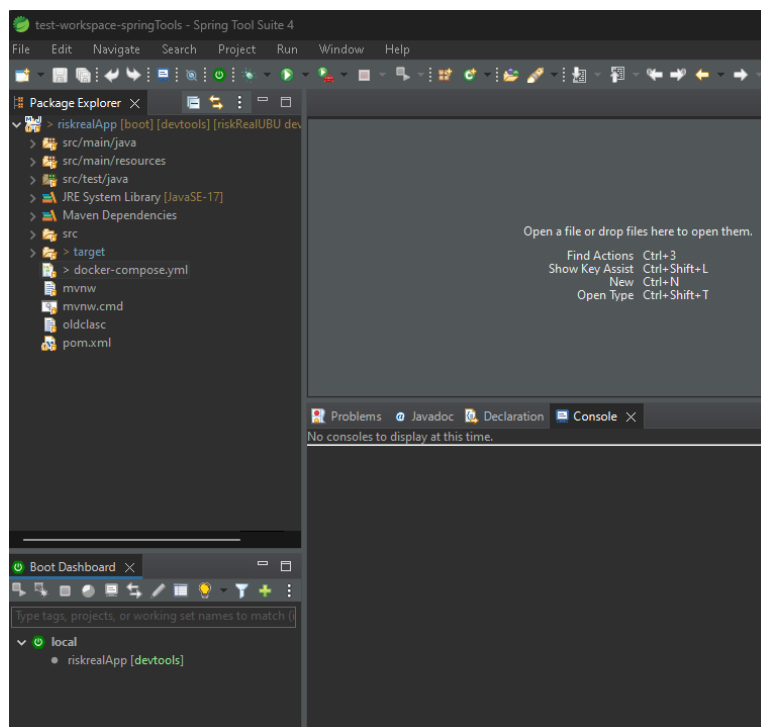


Figura D.16: Proyecto importado en Spring Tools Suite.

Iniciando la aplicación

Tras haber completado correctamente todos los pasos anteriores, y teniendo *Docker Server* funcionando, desde Spring Tools Suite podemos ejecutar la aplicación.

Los puertos por defecto que usa son: **3306 para mysql y 8088 para la aplicación web**, sería recomendable no tener ninguno de los 2 en uso, para el correcto despliegue de la aplicación sin necesidad de tener que cambiar opciones de configuración en *application.properties*.

En la parte inferior izquierda de la interfaz de Spring Tool, tendremos *Boot Dashboard* y si desplegamos *local* nos aparecerá nuestra aplicación **riskrealApp**

Seleccionamos nuestra aplicación y pulsamos en el icono que tenemos justo encima con un *cuadrado rojo y triángulo verde* para iniciarla.

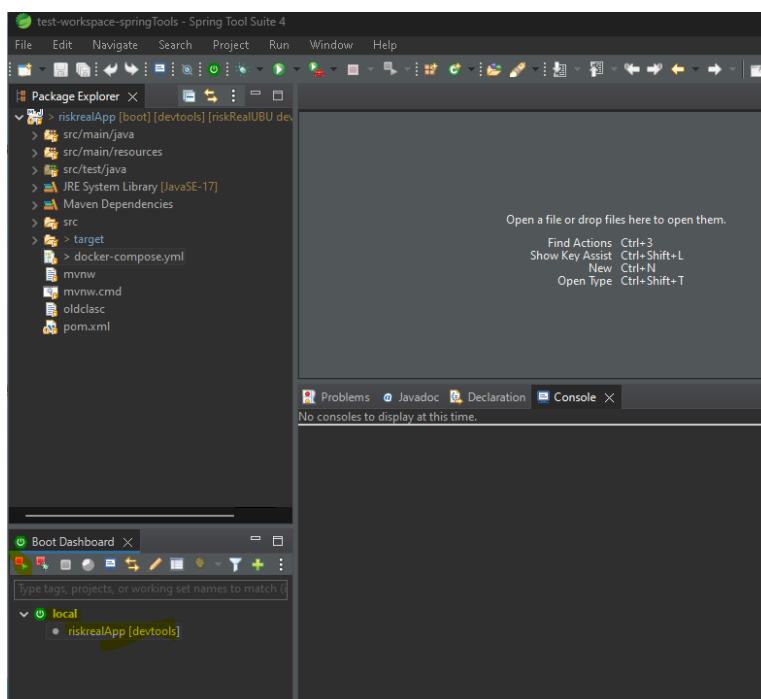


Figura D.17: Iniciar la aplicación en Spring Tools Suite.

Tras iniciar la aplicación, si es la primera vez tardará un poco más porque tiene que descargarse de docker la imagen de MySQL, debería quedarse en un estado similar a este:

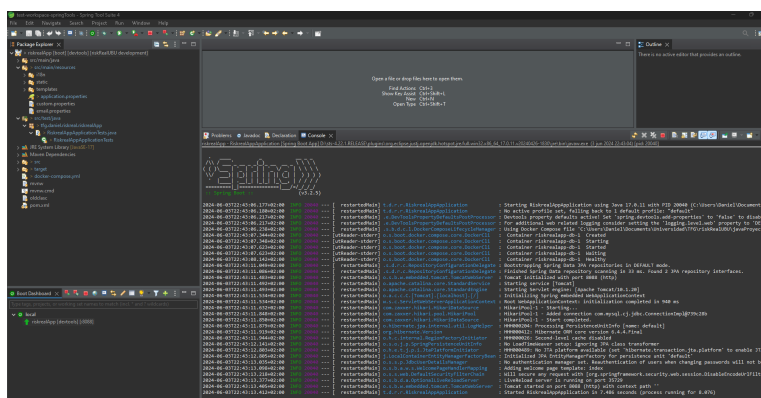


Figura D.18: Aplicación iniciada en Spring Tools Suite.

Ya tenemos nuestro entorno de desarrollo configurado, con nuestra aplicación iniciada y accesible para poder probarla desde <http://localhost:8088>.

D.4. Pruebas del sistema

Apéndice E

Documentación de usuario

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

Apéndice F

Anexo de sostenibilización curricular

F.1. Introducción

Este anexo incluirá una reflexión personal del alumnado sobre los aspectos de la sostenibilidad que se abordan en el trabajo. Se pueden incluir tantas subsecciones como sean necesarias con la intención de explicar las competencias de sostenibilidad adquiridas durante el alumnado y aplicadas al Trabajo de Fin de Grado.

Más información en el documento de la CRUE https://www.crue.org/wp-content/uploads/2020/02/Directrices_Sostenibilidad_Crue2012.pdf.
[7, 6]

Este anexo tendrá una extensión comprendida entre 600 y 800 palabras.

Bibliografía

- [1] Docker desktop. <https://www.docker.com/products/docker-desktop/>. [Internet; Accedido junio de 2024].
- [2] Git. <https://www.git-scm.com/downloads>. [Internet; Accedido junio de 2024].
- [3] Java se development kit 17.0.10. <https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>. [Internet; Accedido junio de 2024].
- [4] Spring tools suite for eclipse. <https://spring.io/tools>. [Internet; Accedido junio de 2024].
- [5] Daniel Fernández Barrientos. Proyecto riskrealubu — github. <https://github.com/Daniel-Fernandez-UBU/riskRealUBU>, 2024. [Internet; actualizado 2024].
- [6] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [7] Wikipedia. Latex — wikipedia, la enciclopedia libre, 2015. [Internet; descargado 30-septiembre-2015].