GRUPO DE TRABAJO G2-1

aSIGNATURA DISEÑO Y PRUEBAS 2, TERCER CURSO DEL GRADO INGENIERÍA INFORMÁTICA, INGENIERÍA DEL SOFTWARE  UNIVERSIDAD DE SEVILLA

ACME-VOLAR

INFORME RELATIVO A PRUEBAS Y SEGUIMIENTO DEL PROYECTO

# ÍNDICE

[ÍNDICE 1](#_Toc36493895)

[INTRODUCCIÓN 2](#_Toc36493896)

[ADVERSIDADES E INCONVENIENTES 3](#_Toc36493897)

[ESTRATEGIA DE PRUEBAS Y RESULTADOS 5](#_Toc36493898)

# INTRODUCCIÓN

El presente documento pertenece al proyecto de la asignatura desarrollado a cabo por el equipo de desarrollo compuesto por el grupo G2-1 de la asignatura, en el marco de desarrollo del proyecto “Acme-Volar” perteneciente a la parte práctica de la asignatura Diseño y Pruebas 2 perteneciente al grado de Ingeniería Informática, Ingeniería del Software cursado en la Universidad de Sevilla.

Con este documento, desde el grupo de trabajo, pretendemos dejar constancia, en primera instancia, de los diferentes contratiempos y adversidades a los que se ha enfrentado el equipo a la hora de llevar a cabo la implementación de la aplicación y sus pruebas.

De manera adicional, pretendemos mostrar y dejar constancia de la estrategia a la hora de implementar y realizar las pruebas y resultados de estas, para que sea considerada a la hora de analizar sus resultados a posteriori.

# ADVERSIDADES E INCONVENIENTES

Durante la elaboración de las tareas asociadas al proyecto, los diferentes miembros del grupo se enfrentaron a problemas o adversidades durante la implementación de la aplicación y sus pruebas, entre las que destacamos las que mostramos a continuación.

En lo relativo a los problemas que aparecieron durante las pruebas de integración, primero nos encontramos con problemas de conexión con la base de datos. Tuvimos problemas tanto de conexión, de privilegios (no lo hacía desde MySQL80 con permisos de administrador), primary keys de algunas tablas eran demasiado largas y no funcionaba correctamente, los varchar venían limitados a 255 en formato distinto del que se usaba en el eclipse (que era latin), el formateo de la fecha era incorrecto… Para solucionar esto, empezamos creando la base de datos sobre MySQL80 (posteriormente haciendo lo mismo que antes) e introdujimos en la URL “spring.datasource.url” en “application-mysql.properties” las distintas propiedades para modificarlo, quedando de la forma:

spring.datasource.url=jdbc:mysql://localhost:3306/acmevolardb?useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC

En lo relativo a los Mocks y Stubs en servicios, aparecieron diversos problemas como que apenas existe documentación en internet sobre ello, en principio solamente funcionan algunos métodos.

En lo relativo a la organización, se ha organizado repartiendo de manera que cada pareja realiza una serie de pruebas distintas los unos a los otros, evitando así que todo el mundo toque todos los tipos de prueba. Para solucionar parcialmente esto, las tareas no terminadas se juntaron y se repartieron en función de las funcionalidades y el tipo de prueba, permitiendo así que muchos miembros del equipo de desarrollo toquen casi totalmente todos los tipos de pruebas que existen.

En lo relativo a la funcionalidad Book, hubo problema en la creación de la vista en la que se crean y actualizan los Books, este error consistía en que no sabíamos cómo crear el atributo del precio total de dicho Book, que dependía del número de reservas/plazas se vaya a crear. Para solventarlo, implementamos una función que se agregó en el botón de creación, generando una alerta o “pop-up” indicando que se va a comprar y preguntando si quiere confirmar la compra, siendo este código:

function submitFunction(){ var quantity =$("#quantity")[0].value; var price = ${flight.price} var totalPrice = quantity \* price; var submit = confirm("The price would be "+totalPrice+". Confirm?"); if(submit == true){ $("#submit").submit(); } };

Este pop-up generaba problemas a la hora de ejecutar las pruebas de la entidad Book, por lo que se decidió descartar ese código, de forma que el Book ahora es creado sin necesidad de confirmar la reserva.

En cuanto al validador de Flight, se detectó un fallo debido a que al rellenar flightStatus lanzaba una excepción, ya que no estaba contemplada en el validador. Al actualizar un Plane, si se deseaba dejar el mismo nombre no se podía porque ya existía en la base de datos, por lo que es mejor no cambiarlo. Este problema también ocurría al actualizar aeropuertos.

En cuanto a Cucumber, uno de los principales inconvenientes fue el navegar a través de las distintas etiquetas HTML, produciéndose en algunos casos errores a la hora de acceder a distintos botones. En este último caso, el error era que un elemento bloqueaba el botón, solucionándose con un clic dentro del formulario.