



Lista de Actividades

PGPI 2024-2025 - Grupo L3.1

LISTA DE ACTIVIDADES

Nombre del Proyecto	Safeport - 1	Código del Proyecto	S317	Fecha de Elaboración	18/10/2024
---------------------	--------------	---------------------	------	----------------------	------------

Historial de versiones		
Fecha	Versión	Cambios
18/10/2024	v1.0	Realización inicial del documento
03/11/2024	v2.0	Adición de actividades

ID	DENOMINACIÓN	DESCRIPCIÓN	ID PAQUETE DE TRABAJO
001	Funcionalidad de registro (backend)	Se debe de crear un componente en Django que permita a los usuarios registrarse en la aplicación.	001
002	Funcionalidad de registro (frontend)	Se debe de crear un formulario con html que permita a los usuarios registrarse en la aplicación.	001
003	Funcionalidad de login (backend)	Se debe de crear un componente en Django que permita a un usuario iniciar sesión en su cuenta mediante su usuario y contraseña.	001
004	Funcionalidad de login (frontend)	Se debe de crear un formulario con html que permita a los usuarios iniciar sesión en la aplicación.	001
005	Cargar productos (backend)	Cuando el usuario cargue la aplicación, se debe extraer todos los productos almacenados en la base de datos.	002
006	Mostrar productos (frontend)	La aplicación debe mostrar al usuario todos los productos almacenados en la base de datos.	002
007	Crear perfil administrador (backend)	La aplicación debe de tener un tipo especial de usuario, este será el administrador.	003
008	Crear vista administrador (frontend)	La aplicación debe de diferenciar entre perfil usuario y perfil administrador al iniciar sesión en la aplicación.	003

LISTA DE ACTIVIDADES

009	Registrar productos con rol de administrador (backend)	La aplicación debe de tener implementada la funcionalidad de creación de un nuevo producto.	004
010	Crear vista de registro de productos con rol de administrador (front)	La aplicación debe de mostrar un formulario al administrador el cual debe permitirle la creación de un nuevo producto.	004
011	Crear cesta (backend)	La aplicación debe de tener implementada la funcionalidad de cesta, esta debe permitir a los usuarios almacenar, ver y eliminar los productos de esta.	005
012	Crear vista de la cesta (frontend)	La aplicación debe mostrar al usuario los elementos que ha seleccionado en una cesta, estos podrán ser añadidos o eliminados de esta antes de confirmar el pago.	005
013	Crear catálogo (backend)	La aplicación debe de tener implementada la funcionalidad de catálogo, este estará estructurado según los tipos de embarcaciones y el puerto.	006
014	Crear vista catálogo (frontend)	La aplicación debe de tener implementada la vista de catálogo, esta mostrará al usuario los productos según los tipos de embarcaciones y el puerto.	006
015	Crear componente de búsqueda en catálogo (backend)	La aplicación debe de tener implementada la funcionalidad componente de búsqueda de catálogo, esta debe permitir la búsqueda de embarcaciones por fecha, y filtrar según el tipo y su puerto.	007
016	Crear vista componente de búsqueda en catálogo (frontend)	La aplicación debe de tener implementada la vista componente de búsqueda de catálogo, esta debe permitir al usuario hacer búsqueda de embarcaciones por fecha, y filtrar según el tipo y su puerto.	007
017	Modificar productos con rol de administrador (backend)	La aplicación debe de tener implementada la funcionalidad de modificar y eliminar los productos registrados por el administrador.	008
018	Crear vista de registro de productos con rol de administrador (front)	La aplicación debe de mostrar unos botones en el listado de productos que permita al administrador la modificación o el borrado del producto.	008
019	Proceso de compra (backend)	La aplicación debe de integrar una API externa que permita al usuario el pagar los productos que hay en el carrito.	009
020	Proceso de compra (fronted)	La aplicación debe de tener implementada una vista que permita al usuario ver un formulario con los datos que ha introducir para realizar el pago de los productos que hay en el carrito.	009

LISTA DE ACTIVIDADES

021	Envío de correo automático(backend)	La aplicación debe de integrar una API externa que envíe correos automatizados que confirmen a los usuarios que la compra se ha realizado correctamente.	010
022	Panel administrador del sistema(backend)	El sistema debe de tener una funcionalidad que permita a los administradores tener un listado de los usuarios registrados en el sistema	011
023	Visualizar datos de los usuarios(backend)	El sistema debe de tener una funcionalidad que permita a los administradores visualizar los datos de las cuentas de usuarios almacenadas, accesible desde el panel de administrador.	011
024	Panel administrador del sistema(frontend)	El sistema debe de mostrar una vista de panel de administrador que permita a los administradores tener un listado de los usuarios registrados en el sistema	011
025	Visualizar datos de los usuarios(frontend)	El sistema debe de mostrar una vista que permita a los administradores visualizar los datos de las cuentas de usuarios almacenadas, accesible desde el panel de administrador.	011
026	Crear contenedor de la aplicación	La aplicación debe de entregarse como contenedor de Docker.	012
027	Desplegar la aplicación	La aplicación debe de ser desplegada en la web, para ello se usará render.	013
028, 029	Creación de pruebas unitarias	Los módulos deben de ser testeados mediante el uso de test unitarios .	014
030, 031	Creación de pruebas automáticas	Se deben de implementar test automáticos con selenium para comprobar que la aplicación funciona de forma correcta.	014
032	Configuración del proyecto	Instalación inicial del proyecto y su configuración para poder comenzar el desarrollo	015
033	Aprendizaje de la tecnología	Familiarizarse con el framework a usar y las demás tecnologías que serán necesarias	015
034	Crear informes	Redacción de los informes de seguimiento del proyecto.	016
035	Creación del Manual de Usuario	Se creará un manual donde se detallarán las funcionalidades y su uso de la aplicación	017