

Módulo 9

Desarrollo de portafolio de un producto digital

El Portafolio de Productos



Módulo 9

AE 1

OBJETIVOS

Comprender la importancia de un portafolio de productos y herramientas para su creación. Aplicar buenas prácticas en el diseño e implementación de un producto digital funcional. Desarrollar un producto tecnológico que resuelva un problema real con enfoque profesional.



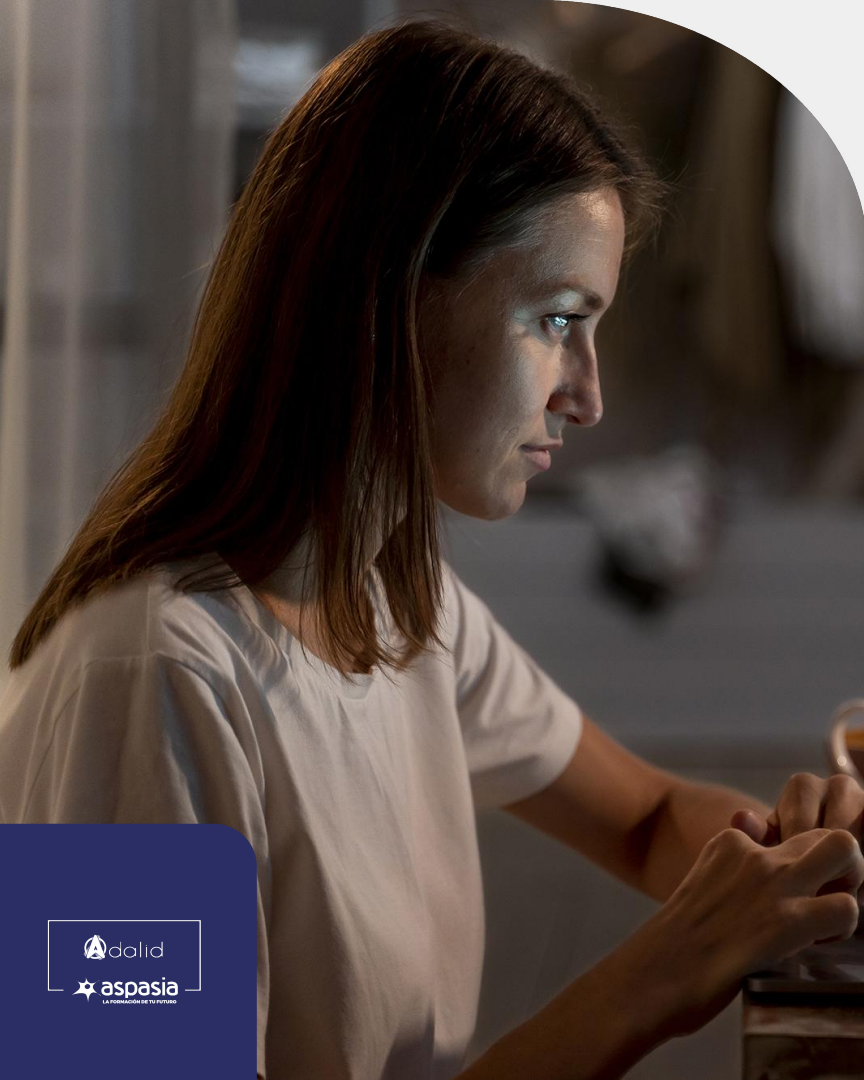
¿QUÉ VAMOS A VER?

- El Portafolio de Productos.
- Qué es un portafolio de productos.
- Importancia de contar con un portafolio.
- Buenas prácticas para la creación de un portafolio de productos.
- Herramientas que se pueden utilizar para la creación del portafolio de productos:
 - GitHub.
 - Hosting.
 - Página web personal.
 - YouTube.
 - Otros.



¿QUÉ VAMOS A VER?

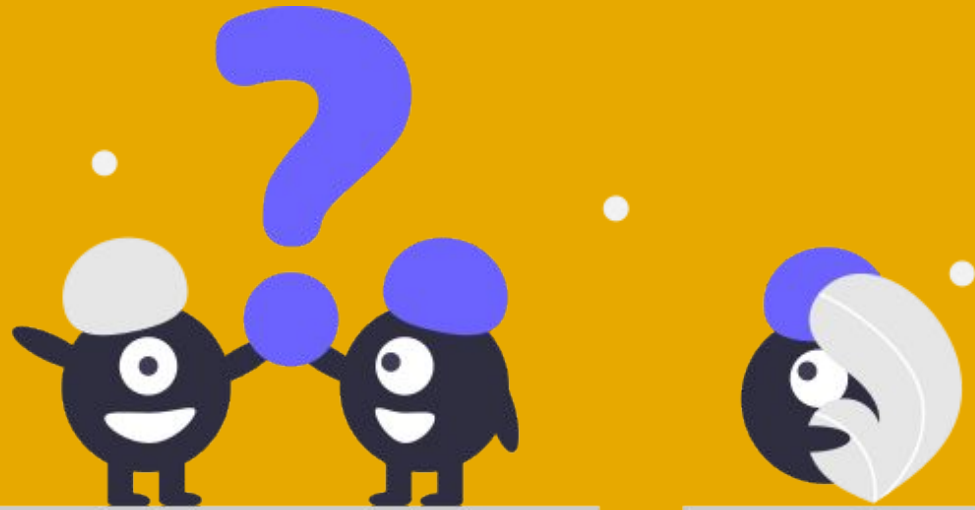
- Elaborar un Producto Tecnológico.
 - Distingue conceptos y buenas prácticas para el diseño de un producto digital que resuelve un problema.
 - Utiliza técnicas y herramientas de la disciplina para la implementación de un producto digital.
 - Ajustes finales y cierre de entregable.
 - Implementa un producto funcional que resuelve un problema real utilizando las buenas prácticas de la disciplina.



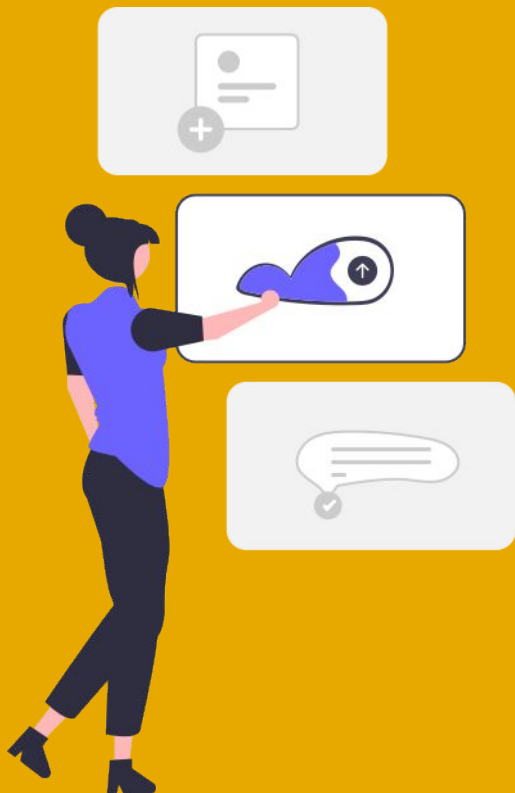
¿QUÉ VAMOS A VER?

- Finalización del Proyecto.
 - Revisión del producto construido a lo largo del curso.
 - Depuración y mejora del producto.
 - Feedback y retroalimentación.
 - Ajustes finales y cierre de entregable.

¿Que es un portafolio?



Introducción al Portafolio de Productos



¿Qué es un Portafolio de Productos?

Un portafolio de productos es una **colección de proyectos, trabajos y logros** que demuestran habilidades y experiencia en un área específica, como el desarrollo de software, diseño UX/UI, o marketing digital.

Importancia de Contar con un Portafolio

Beneficio	Descripción
Visibilidad profesional	Permite a reclutadores y clientes evaluar tu trabajo.
Construcción de marca personal	Refuerza tu identidad como profesional en el sector.
Evidencia de habilidades	Muestra proyectos reales y resultados obtenidos.
Diferenciación	Destaca entre otros candidatos con experiencia similar.

Elementos Claves de un Portafolio Exitoso

¿Qué debe incluir un portafolio?

- **Página de presentación personal** con una breve descripción y contacto.
- **Lista de proyectos destacados** con descripciones y capturas de pantalla.
- **Enlaces a repositorios** de código en GitHub/GitLab.
- **Pruebas de habilidades** como certificaciones o artículos técnicos.
- **Un diseño atractivo y funcional.**

Buenas Prácticas para la Creación del Portafolio

Práctica	Descripción
Mostrar proyectos relevantes	Seleccionar los mejores trabajos, no sobrecargar con contenido irrelevante.
Incluir capturas y descripciones	Explicar cada proyecto de manera clara.
Optimizar para dispositivos móviles	Asegurar que el portafolio sea accesible en cualquier pantalla.
Facilitar el contacto	Agregar links a redes sociales y correo.

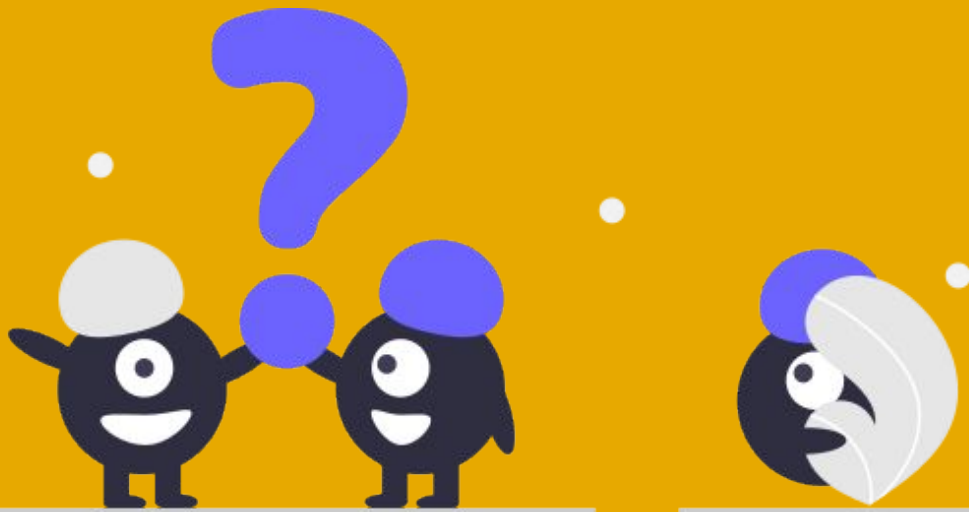
Plataformas para Alojar un Portafolio

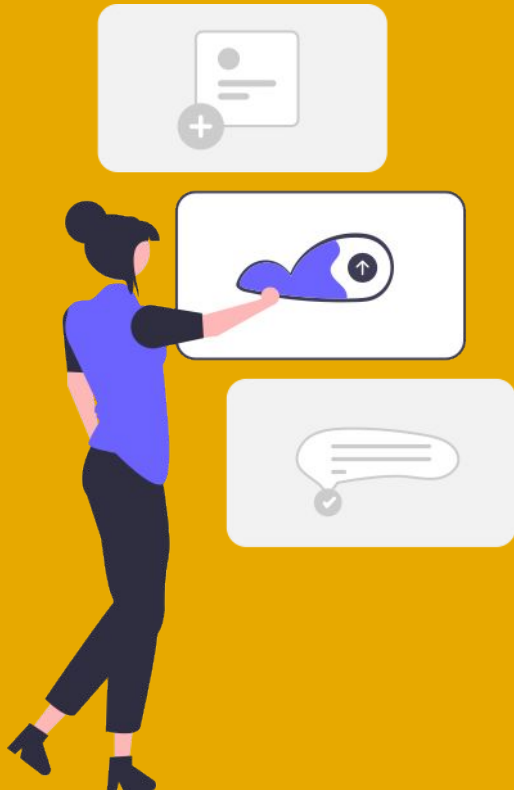
Herramienta	Descripción
GitHub Pages	Permite alojar portafolios estáticos directamente desde GitHub.
Netlify	Plataforma de hosting con integración CI/CD.
Vercel	Especializado en proyectos React y Next.js.
WordPress	Ideal para blogs y portafolios con CMS.

Otras Herramientas para Crear un Portafolio

Herramienta	Uso
GitHub	Repositorios de código, README detallados.
YouTube	Mostrar proyectos en video, tutoriales y demostraciones.
LinkedIn	Crear una red profesional y compartir proyectos.
Behance/Dribbble	Para diseñadores que buscan mostrar su trabajo visualmente.

¿Como crearas tu portafolio?





Diseño de un Producto Digital

¿Qué es un Producto Tecnológico?

Un producto tecnológico es una solución digital que resuelve **un problema o necesidad específica** mediante el uso de software, hardware o servicios en la nube.

Ejemplos de Productos Tecnológicos:

- Aplicaciones móviles.
- Plataformas web.
- Servicios en la nube (SaaS).
- Dispositivos IoT (Internet of Things).

Buenas Prácticas para el Diseño de un Producto Digital

Etapas	Descripción
Identificación del Problema	Analizar y definir la necesidad del usuario.
Investigación y Benchmarking	Explorar soluciones existentes y definir la diferenciación.
Definición de Objetivos	Establecer funcionalidades y metas del producto.
Diseño de la Experiencia de Usuario (UX/UI)	Crear prototipos y pruebas de usabilidad.
Desarrollo Tecnológico	Construcción del producto utilizando herramientas adecuadas.
Pruebas y Validación	Detectar errores y mejorar funcionalidades antes del lanzamiento.

Técnicas y Herramientas para Implementar un Producto

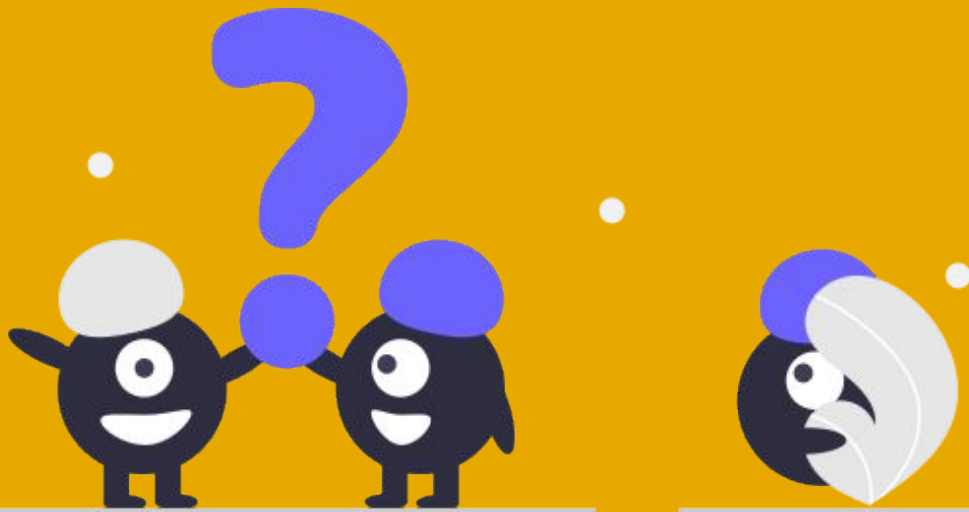
Área	Herramientas Recomendadas
Diseño UX/UI	Figma, Adobe XD, Sketch
Desarrollo Frontend	React, Vue.js, Angular
Desarrollo Backend	Node.js, Django, Flask, Rails
Bases de Datos	Firebase, PostgreSQL, MongoDB
Infraestructura y DevOps	Docker, Kubernetes, AWS, Netlify
Pruebas y Validación	Jest, Cypress, Postman

Validación y Ajustes Antes del Lanzamiento

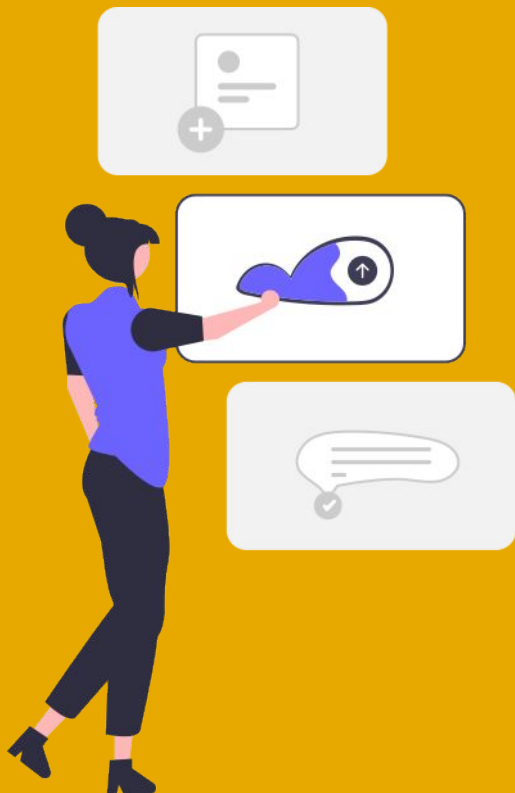
Factores Clave a Evaluar Antes del Cierre:

- **Funcionalidad Completa** → ¿El producto cumple su propósito?
- **Experiencia de Usuario (UX)** → ¿Es fácil de usar?
- **Rendimiento y Escalabilidad** → ¿Funciona bien en diferentes dispositivos?
- **Pruebas de Seguridad** → ¿Protege la información del usuario?

¿Cuál será tu próximo producto digital?



Revisión del Producto Construido



Evaluación del Producto Construido

Antes de dar por finalizado el proyecto, es fundamental hacer una **revisión detallada de todo lo que se ha construido a lo largo del curso.**

Evaluación del Producto Construido

Puntos Clave a Evaluar:

- **Funcionalidad Completa** → ¿El producto cumple con los requerimientos iniciales?
- **Experiencia de Usuario (UX/UI)** → ¿Es intuitivo y fácil de usar?
- **Rendimiento y Optimización** → ¿Carga rápido y es eficiente?
- **Compatibilidad** → ¿Funciona correctamente en distintos dispositivos y navegadores?
- **Seguridad** → ¿Los datos están protegidos adecuadamente?

Identificación de Errores y Mejoras

Proceso de Depuración:

- Ejecutar pruebas unitarias e integración.
- Buscar errores de código y optimizar funciones ineficientes.
- Depurar logs de consola y errores en el navegador.

Ejemplo de Depuración con `console.log()` y `debugger` en JavaScript:

```
function calcularTotal(precio, cantidad) {  
  debugger; // Permite pausar la ejecución y analizar variables  
  console.log("Precio:", precio, "Cantidad:", cantidad);  
  return precio * cantidad;  
}
```


Identificación de Errores y Mejoras

Herramientas para Depuración:

- DevTools en el navegador (Chrome, Firefox).
- Extensiones como React Developer Tools.
- Pruebas con Jest y Cypress.

Importancia del Feedback en el Desarrollo de Software

¿Por qué es importante el feedback?

- Ayuda a detectar **problemas no previstos**.
- Mejora la **experiencia del usuario** con ajustes basados en su retroalimentación.
- Facilita la **colaboración en equipo** al compartir ideas y mejoras.

Fuentes de Feedback:

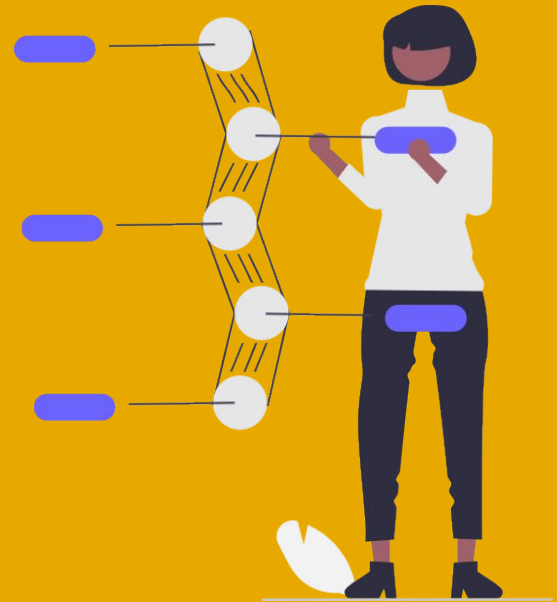
- **Usuarios finales:** Identifican problemas y sugerencias.
- **Equipo de desarrollo:** Discuten mejoras en código y arquitectura.
- **Pruebas automatizadas:** Detectan fallos en la lógica del sistema.

Ajustes Finales y Cierre del Entregable

Checklist Final antes de la Entrega:

- Corrección de errores identificados.
- Últimos ajustes de UX/UI para una mejor experiencia.
- Optimización de rendimiento para mayor eficiencia.
- Pruebas de compatibilidad en distintos dispositivos.
- Preparación de la documentación del proyecto.

Resumen de lo aprendido



Resumen de lo aprendido

- Un portafolio digital demuestra habilidades y experiencia con herramientas como GitHub o sitios web.
- Diseñar un producto tecnológico requiere identificar problemas y aplicar buenas prácticas de desarrollo.
- La implementación de un producto funcional implica planificación, iteración y ajustes finales.
- Feedback y mejoras continuas son clave para entregar un producto sólido y profesional.

GRACIAS POR TU ATENCIÓN

Nos vemos en la próxima clase

