



códigofacilito



Proyecto Final

Bootcamp DevOps



📋 Proyecto Final

>_ ¡Llegó el momento del proyecto final!



Objetivos

>_ Objetivos

>_ Lineamientos

>_ Entregables

>_ Plazos



📋 Proyecto Final

>_ Los Objetivos del proyecto es todo lo que queremos lograr



Objetivos

- >_ Aplicar los conocimientos adquiridos durante el Bootcamp
- >_ Inicien o adicionen este proyecto a su portafolio
- >_ Desarrollar un proyecto que reflejen situaciones reales, nos enfrentamos a desafíos auténticos, lo que nos prepara mejor para el entorno laboral.
- >_ Podernos graduar del Bootcamp



Proyecto Final

>_ ¡Este proyecto es una oportunidad para demostrar nuestro potencial y habilidades únicas!

Es una oportunidad para desafiarnos, crecer y brillar como verdaderos profesionales de DevOps.

Cada desafío que superamos nos acerca más a nuestro objetivo final y nos permite mostrar al mundo lo que somos capaces de lograr. ¡Vamos a aprovechar al máximo esta oportunidad y crear algo que no solo nos haga sentir orgullosos, sino que también inspire a otros a alcanzar nuevas alturas en su propio camino hacia el éxito!



☰ Proyecto Final [1]

>_ El proyecto será un aplicativo Web

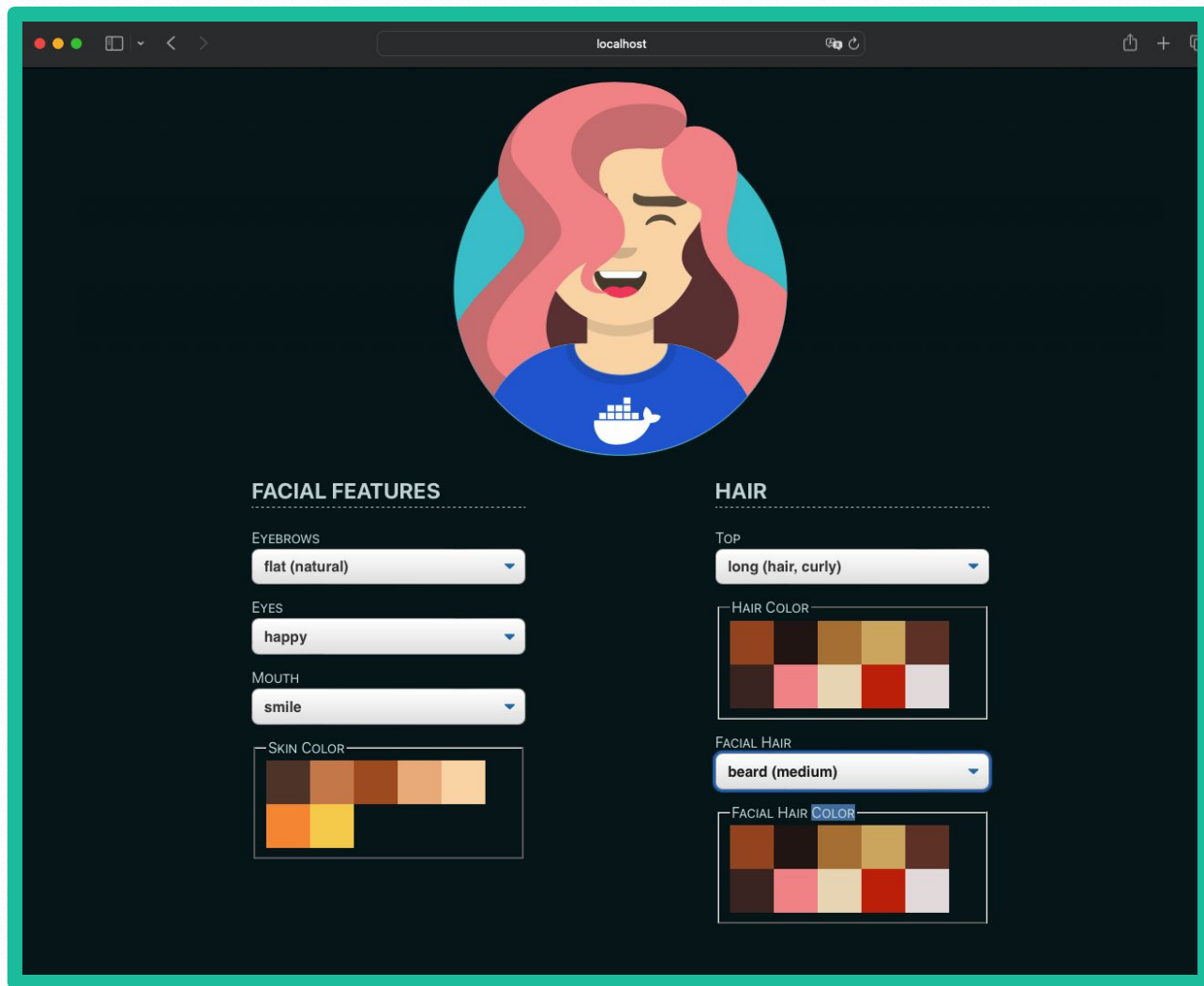
- BackEnd, API [Python]
- FrontEnd [React+VITE]

Nota: Es un Proyecto Open Source



<https://gitlab.com/training-devops-cf/avatares-devops.git>

Avatares



☰ Descripción (Proyecto Backend)

- El backend de la API está desarrollado en Python 3.10 utilizando un framework web como Flask.

- Requiere dos variables de entorno

`FLASK_APP=app.py`

`FLASK_ENV=development`

- La API proporciona endpoints para generar avatares basados en diferentes parámetros de entrada.

`/api/avatar`

`/api/avatar/spec`

`/ready`

☰ Descripción (Proyecto Frontend)

- El frontend es una aplicación de página única (SPA) desarrollada en Node.js 18 utilizando un framework como React+vite.
- Requiere dos variables de entorno
VITE_HOST=0.0.0.0
VITE_PORT=5173
- Usar el gestor de paquetes npm
- El frontend se comunica con el backend a través de solicitudes HTTP para generar y mostrar avatares.

☰ Proyecto Final [2]

>_ El proyecto será un aplicativo Web

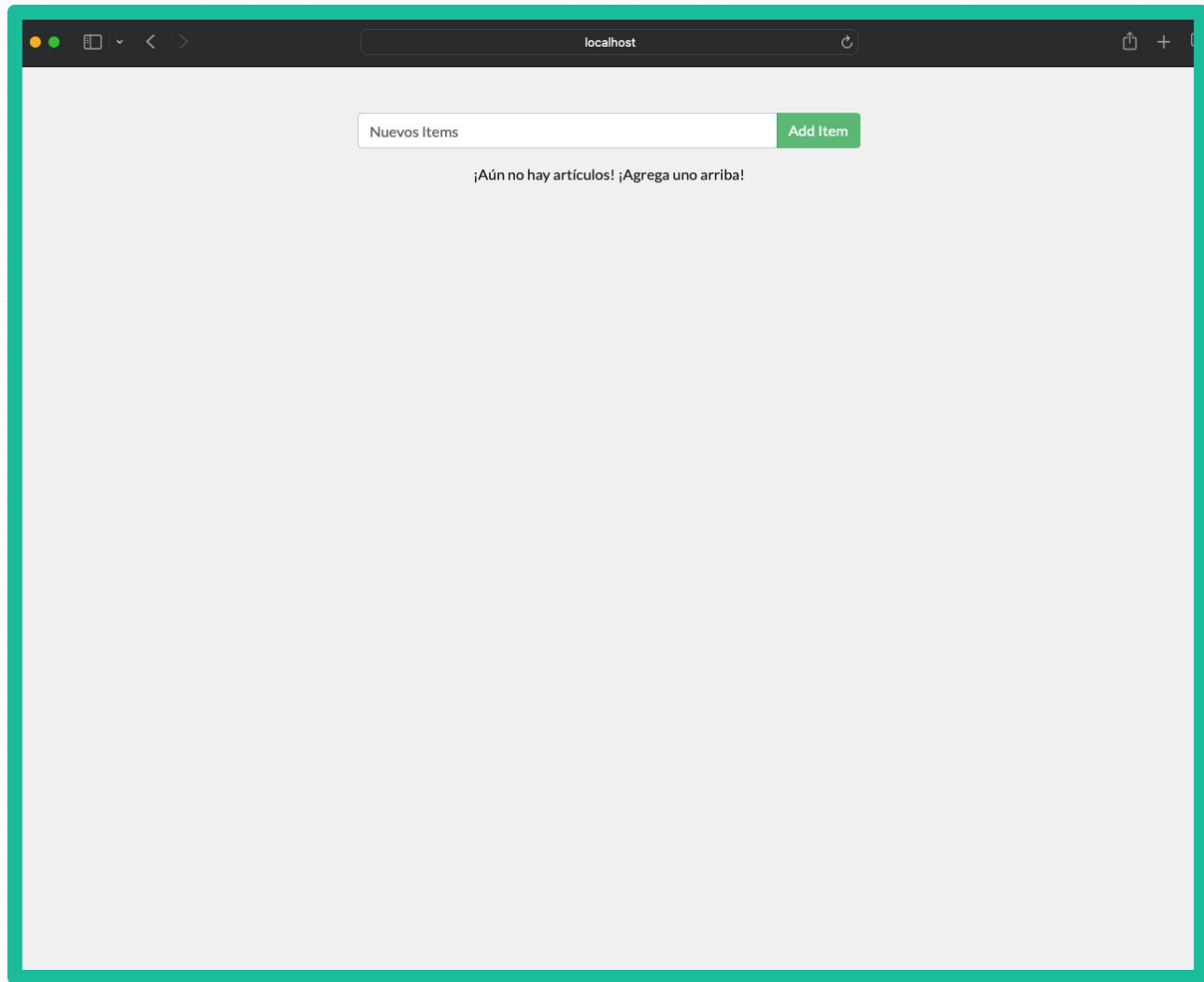
- Backend Node.js con Express
- Base de datos Mysql

Nota: Es un Proyecto Open Source



<https://gitlab.com/training-devops-cf/todo-list-devops.git>

Todo-List



📋 Descripción (Proyecto Backend)

Desarrollar una aplicación web simple utilizando Node.js y Express para el backend.

- Requiere Node 18
- La aplicación permitirá a los usuarios crear, leer, actualizar y eliminar tareas de una lista.
- Se integrará con una base de datos MySQL para almacenar las tareas.

☰ Proyecto Final [3]

>_ Proyecto Libre.

- Si estás realizando otro curso en Código Facilito y cuentas con un proyecto [Aplicación] puedes tomarlo como base para realizar el proyecto final.



☰ Descripción (Proyecto General)

- Contenedores
 - a. Utilizar contenedores Docker para empaquetar y distribuir la aplicación backend y frontend, junto con todas sus dependencias.
 - b. Definir Dockerfiles para construir imágenes de contenedor optimizadas y seguras.
 - c. Utilizar Docker Compose para orquestar contenedores en entornos locales de desarrollo y pruebas.
 - d. Subir las imágenes al Registry de Docker-Hub
- Orquestación de Contenedores con Kubernetes
 - a. Implementar Kubernetes para orquestar y administrar los contenedores en un entorno de producción.
 - b. Definir y desplegar manifestos de Kubernetes (archivos YAML) para desplegar y escalar la aplicación en clústeres de Kubernetes.
 - c. Pueden usar Minikube/k3s/K0s/Killercoda/AWS-EKS

☰ Descripción (Proyecto General)

- Observabilidad
 - a. Implementar herramientas de observabilidad como Prometheus para la recopilación de métricas, Grafana para la visualización de métricas.
 - b. Configurar paneles de control en Grafana para monitorear el rendimiento de la aplicación, la utilización de recursos y la salud del clúster de Kubernetes
- Buenas Prácticas
 - a. Aplicar prácticas de seguridad.
 - b. Implementar estrategias de gestión de configuración para mantener la configuración de la aplicación separada del código fuente.
 - c. Utilizar GitFlow para gestionar el flujo de código y versionado de la aplicación.

☰ Descripción (Proyecto General)

- Preferiblemente en alguna cloud provider
- Puede ser individual o grupal de hasta 3 integrantes
- Calidad en el Código, Código Limpio
- Documentación en el código (En donde sea necesario)

Entregables

- >_ Repositorio de código en un SVC como GitHub, GitLab o Bitbucket.
- >_ Documentación que describa la arquitectura del proyecto, la configuración de la infraestructura, los pasos de implementación de DevOps y las prácticas seguidas.
- >_ Pipeline de CI/CD configurado y funcionando para el despliegues.
- >_ Aplicación desplegada y ejecutándose correctamente en un clúster de Kubernetes
- >_ Herramientas de observabilidad configuradas y proporcionando información útil sobre el estado y rendimiento de la aplicación.



Entregables

- >_ Especificación Integrantes
- >_ Cómo funciona, cómo se conecta
- >_ Puntos clave que quieran destacar



Fechas

>_ **Fecha de entrega:** 3 de Junio

>_ **Medio de entrega:** <https://forms.gle/QfBTaCGB5RbtUZEPA>



Tips

- >_ ¡Diviértete!
- >_ Apóyate de los profes y de tus compañeros
- >_ Hazlo Tuyo (que demuestre quién eres)
- >_ Elige un proyecto Retador
- >_ Desafiate!



¡IMPORTANTE!

>_ Asegúrate de lo siguiente:

- Que tu repositorio es público o que agregues al team de instructores como colaboradores de tu repositorio
- Que si agregas un link directo a tu presentación (en el form o repositorio), el acceso no debe ser solicitado (es decir, que cualquier persona tenga acceso)





¡Gracias!

¿Preguntas?

 @codigofacilito

 contacto@codigofacilito.com

