

Implementación ambiente de desarrollo

Sistema La Casa Del Rayo

Documento de Implementación de ambiente del desarrollo

Tabla de contenido

■ Datos del documento	pág. 3
1.1 Histórico de revisiones	pág. 3 1.2
Información del proyecto	pág. 3 1.3
Integrantes	pág. 3 ■
Introducción	pág. 3 2.1

Alcance	pág. 3	2.2
Importancia de un entorno de desarrollo configurado	pág. 3	🎬
Requisitos del sistema	pág. 3	3.1
Hardware	pág. 4	3.2
Software	pág. 4	🎬
Instalación de herramientas	pág. 4	4.1
Instalación de Python y pip	pág.	4.2
Instalación de MySQL Workbench	pág. 4	🎬
Preparación del entorno de desarrollo	pág.	5.1
Creación de entorno virtual (venv)	pág.	5.2
Instalación de dependencias (requirements.txt)	pág.	5
Configuración de la base de datos	pág.	5.6.1
Parámetros de conexión MySQL (host, usuario, clave)	pág.	5.6.2
Migraciones y carga inicial de datos	pág.	5
Estructura del proyecto	pág.	5
Ejecución y desarrollo	pág.	6.8.1
Servidor de desarrollo local (runserver)	pág.	6.8.2
Acceso desde navegador y puertos	pág.	6.8.3
Entorno de pruebas recomendado (VS Code + navegador) ..	pág.	7
Vistas de la interfaz	pág.	7
Consideraciones técnicas	pág.	7
Buenas prácticas y seguridad de credenciales	pág.	7
Notas para despliegue futuro (Docker/CI-CD)	pág.	7
Conclusión	pág.	8

Datos del documento

Histórico de Revisiones

Versión	Fecha	Descripción/cambio	Autor
1.0	17/10/2025	Inicio documento de venv desarrollo	Benjamin Cordero

Información del Proyecto

Organización	Duoc UC. Escuela de Informática y Telecomunicaciones
Sección	001D
Proyecto (Nombre)	La Casa Del Rayo
Fecha de Inicio	22/08/2025
Fecha de Término	21/11/2025
Patrocinador principal	-
Docente	Arturo Vargas

Integrantes

Rut	Nombre	Correo
21.297.372-6	Benjamín Cordero	ben.cordero@duocuc.cl

20.114.384-5	Loreto Díaz	lo.diazb@duocuc.cl
19.524.578-9	Lukas Urrutia	lu.urrutia@duocuc.cl

Introducción

El presente documento tiene como propósito definir el flujo de desarrollo de la aplicación web del proyecto *La Casa Del Rayo*, estableciendo de manera clara los requisitos, pasos de instalación y configuración del entorno de trabajo. Esta guía busca asegurar una correcta implementación del entorno de desarrollo, minimizando errores y optimizando los tiempos de configuración para futuros desarrolladores.

Alcance

Este manual corresponde exclusivamente a la aplicación **web** desarrollada con **Django** y está enfocado en la preparación del entorno local para ejecución, pruebas y despliegue.

Importancia de un Entorno de Desarrollo Configurado

Contar con un entorno correctamente configurado permite asegurar:

- Estabilidad y consistencia en el desarrollo
- Reproducibilidad entre desarrolladores
- Colaboración eficiente en el equipo
- Reducción de errores por configuraciones inconsistentes

Requisitos del Sistema

Hardware

CPU	Intel i5 o Superior	Ryzen 5 o Superior
RAM	Mínimo: 8GB Recomendado 16GB	Mínimo: 8GB Recomendado 16GB
Espacio libre	SSD O HDD de 256GB (espacio libre mínimo 10GB)	SSD O HDD de 256GB (espacio libre mínimo 10GB)
GPU	Uso de emulador Android o dispositivo físico	Uso de emulador Android o dispositivo físico
Conectividad	Acceso estable a Internet (3 Mbps o superior)	Acceso estable a Internet (3 Mbps o superior)

Software

SO	Windows 10/11 (64bits) o Ubuntu 22.04 LTS
----	-------------------------------------------------

Python	v3.11 o superior
pip	v23.x o superior
MySQL Workbench	V8.x o superior
Django	v5.2 o superior
Docker Desktop	4.48.0 o superior
Docker Compose	2.20.0 o superior
Git	Last Release
IDE	VsC o Equivalente

Instalación de Herramientas

Paso a paso, por secciones:

Instalación de Python y pip

Descargar Python desde su pagina oficial [Python](#). Asegurarse de marcar la opción “Add Python to PATH” durante la instalación. Luego de la instalación, verificar la correcta instalación con:

Para abrir la consola deberá hacer Win+R escribir cmd y darle al “Return”

```
python --version
pip --version
```

Instalación de [MySQL WorkBench](#)

Descargar desde la pagina oficial de [MySQL WorkBench](#), y descargar la versión compatible con su sistema operativo. Durante la instalación, se recomienda configurar:

- Base de datos: gimnasio_db
- Usuario: admin
- Contraseña: 57171580
- Host: db-gimnasio.c7gww4go6jj5.us-east-2.rds.amazonaws.com

Instalación de Docker

1. Descarga Docker Desktop desde <https://www.docker.com/products/docker-desktop>.
2. Instálalo y asegúrate de que el **Engine esté en “Running”**.
3. Verifica en terminal:
 docker --version
 docker compose version

Creación de entorno virtual

Se recomienda usar un entorno virtual para aislar las dependencias del proyecto:

```
C:\Users\PC>cd LaCasaDelRayo
C:\Users\PC\LaCasaDelRayo>python -m venv venv
C:\Users\PC\LaCasaDelRayo>cd venv
C:\Users\PC\LaCasaDelRayo\venv>cd Scripts
C:\Users\PC\LaCasaDelRayo\venv\Scripts>activar
(venv) C:\Users\PC\LaCasaDelRayo\venv\Scripts>cd..
(venv) C:\Users\PC\LaCasaDelRayo\venv>cd..
```

Instalación de dependencias

Instalar todas las dependencias necesarias con el siguiente comando:

```
pip install -r requirements.txt
```

Estructura del Proyecto

Descripción general de las carpetas y archivos principales del proyecto:

```
LACASADELRAYO/ # Raíz del proyecto
|   core/ # Proyecto Django (configuración global) |   |
core/
|   |   __init__.py
|   |   asgi.py # Config ASGI
|   |   settings.py # Ajustes del proyecto (apps, DB, static, etc.) |   |
|   |   urls.py # Enrutador raíz: incluye rutas de las apps |   |   wsgi.py
# Config WSGI
|   |   manage.py # Script de administración de Django (runserver,
migrate...)
|
|   sistemaPT/ # App principal del sistema (gimnasio) |   |
|   init_.py
|   |   admin.py # Registro en Django Admin
|   |   apps.py # Configuración de la app
|   |   forms.py # Formularios (Cliente, Empleado, Insumo, etc.) |   |
|   |   models.py # Modelos (Cliente, Empleado, Insumo...) |   |   urls.py # Rutas
de la app (clientes, empleados, insumos...) |   |   views.py # Vistas (listas,
CRUD, PDF, login personalizado...) |   |   migrations/ # Migraciones de los
modelos
|   |   static/ # Archivos estáticos propios de la app |   |   css/ ... #
Ej: clientes.css, header.css, responsive.css |   |   templates/ #
Plantillas HTML de la app
```

Ejecución y desarrollo

Servidor de desarrollo local

Para iniciar el servidor local y probar el funcionamiento de la app:

```
python manage.py runserver
```

Acceder desde: <http://127.0.0.1:8000>

Levantar n8n y docker

1. En una terminal de visual studio ejecutar estas lineas de comando:

```
docker compose up -d
```

```
docker compose -f docker-compose.n8n.yml up -d
```

2. Luego pueden ingresar a ver el flujo mediante esta url:

<http://localhost:5678/>

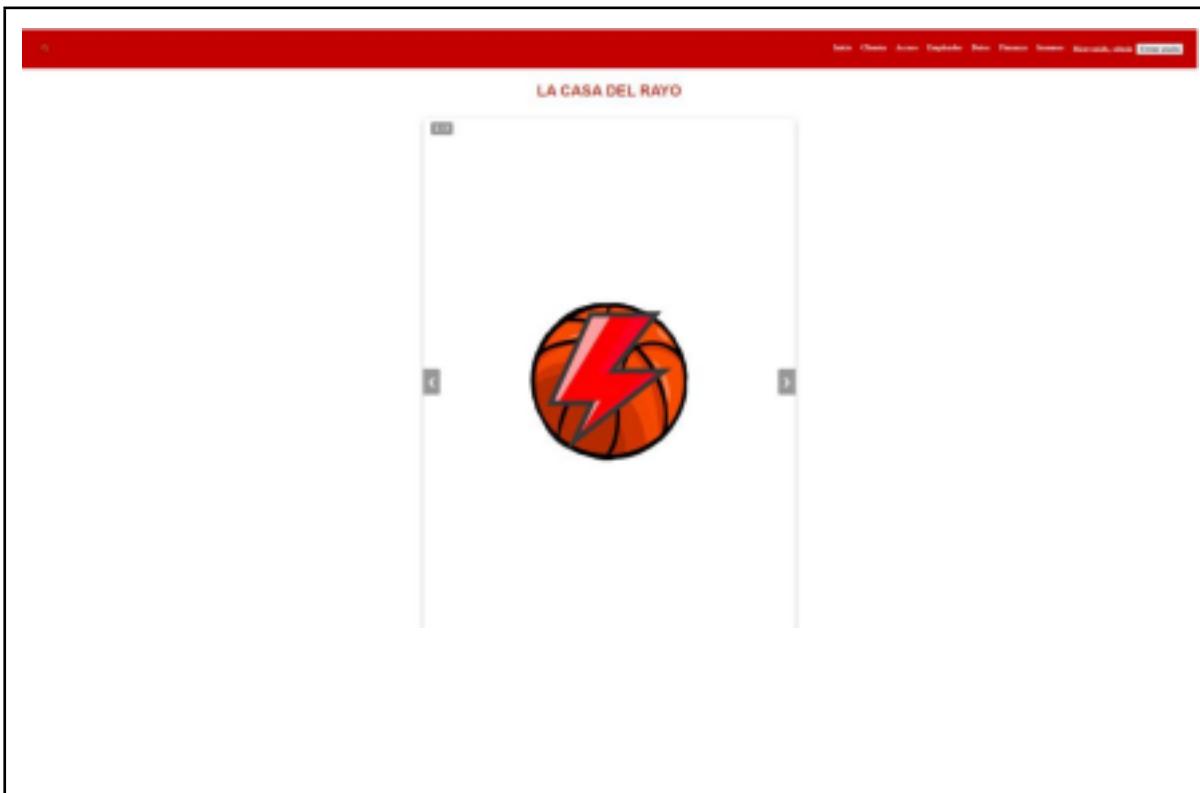
Consideraciones Técnicas

Levantar el servidor de desarrollo

- **Base de datos:** MySQL configurado en la nube
- **Requisitos de puertos:** Usar puerto 8000 en desarrollo

- **Entorno de pruebas:** Recomendado usar VS Code + Chrome/Brave

Vistas del interfaz



CLIENTES											
Nuevo Cliente		Lista de Clientes									
Nombre	Apellido	RUT	FECHA NACIMIENTO	PLAZO	TIPICO	RETIRO DE	INCIDENCIA	ALERTAS	ESTADO	FICHA	ACCIONES
Casta	Huato	18.410.700-2	10 de Mayo de 1990	Busto	Manual	10	Sí				
Talma	Gonzalo	18.510.446-1	10 de Agosto de 1990	Busto	Manual	5	Sí				
Huato	Aime	18.500.446-4	15 de Julio de 1990	Busto	Manual	15	Sí				
Camino	Lopez	18.800.446-2	18 de Mayo de 1992	Presentación	Presentación	10	Sí				
Spinoza	Euge	18.300.446-2	13 de Agosto de 1992	Busto	Manual	10	Sí				
Durán	Pedro	18.110.446-2	10 de Junio de 1992	Busto	Manual	5	Sí				
Urquiaga	Alex	18.110.446-2	27 de Septiembre de 1992	Busto	Presentación	10	Sí				
Alfaro	Martina	18.800.446-2	17 de Mayo de 1992	Presentación	Manual	5	Sí				
Urdiales	Eugenio	18.800.446-2	17 de Septiembre de 1992	Busto	Manual	10	Sí				
Casta	Huato	18.410.700-2	10 de Mayo de 1990	Busto	Manual	10	Sí				

ID	NOMBRE	APELLIDO	FECHA_NACIMIENTO	CARGO	ACCIONES
1	Pedro	Santos	21/04/1990	Desarrollador	
2	Ricardo	Perez	15/02/1985	Analista	
3	Cristina	Soto	17/09/1978	Supervisora	
4	Juan	Lopez	05/08/1992	Administrador	
5	Patricia	Diaz	10/05/1988	Asistente	
6	Eduardo	Perez	09/06/1980	Desarrollador	
7	Maria	Santos	25/07/1995	Oficina	

Conclusión

La correcta implementación del entorno web permite mantener la estabilidad del backend, asegurar la persistencia de datos y facilitar la integración con otros servicios.

Este entorno basado en Django y MySQL está preparado para despliegue en plataformas modernas y puede ser extendido para entornos productivos mediante Docker o CI/CD.