

Manual de Implementación

APLICACIÓN WEB: EFECTO DEL SALARIO MÍNIMO EN LA
ECONOMÍA DEL PAÍS DURANTE LOS ÚLTIMOS 5 AÑOS.

MATERIA: HERRAMIENTAS DE PRODUCTIVIDAD

UNIVERSIDAD DE EL SALVADOR FACULTAD DE INGENIERIA Y ARQUITECTURA

Jonathan David Larin Alvarenga

Carlos Alexander De León Gutiérrez

María de los Ángeles Acosta Méndez

Tabla de contenido

INTRODUCCIÓN	1
OBJETIVOS.	2
General:	2
Específico:.....	2
Requisitos del Sistema	3
Requerimientos de Software	3
Instalación y configuración	4
Arquitectura de la solución	39
Mantenimiento y actualizaciones.....	44
Contacto para Soporte	48

INTRODUCCIÓN

El presente manual está diseñado con el propósito de tener una guía completa sobre el uso, configuración y mantenimiento y la resolución de problemas de la aplicación web creada sobre el tema EFECTO DEL SALARIO MÍNIMO EN LA ECONOMÍA DEL PAÍS DURANTE LOS ULTIMOS 5 AÑOS. En el manual se detallan ciertos aspectos de interés para diferente tipo de personas como lo son los Desarrolladores, Administradores del sistema y Usuario final.

La presente aplicación web es una solución web en tiempo real desarrollada para conocer los efectos del salario mínimo en la economía de los salvadoreños durante los últimos 5 años con el propósito de conocer la realidad del país por medio de La Oficina Nacional de Estadística y Censos (ONEC) del Banco Central de Reserva de El Salvador. La aplicación tiene una interfaz intuitiva que permite a los usuarios poder acceder a la información de una forma fácil filtrando la información puntual requerida y mostrándola en graficas para que sea más comprensible.

OBJETIVOS.

General:

Proporcionar una guía completa y detallada sobre la implementación, mantenimiento, uso y solución de posibles problemas futuros de la aplicación web sobre el tema **“EFECTO DEL SALARIO MÍNIMO EN LA ECONOMÍA DEL PAÍS DURANTE LOS ULTIMOS 5 AÑOS”**.

Específico:

- Asegurar la correcta implementación y configuración.
- Proveer instrucción claras y detalladas para así poder guiar a resolución de problemas comunes.
- Facilitar la comprensión de la arquitectura del sistema.

Requisitos del Sistema

Requerimientos de Software

Para llevar a cabo la selección de las tecnologías de software a utilizar, se tomaron en

cuenta los siguientes criterios:

Estándares tecnológicos de la ONEC: El uso de herramientas de software en la construcción de los sistemas de información se limita al uso de tecnología Python.

Naturaleza del proyecto: el sistema informático a desarrollar está orientado a un ambiente Web. A continuación, se detallan las tecnologías y herramientas de software a utilizar en el proyecto:

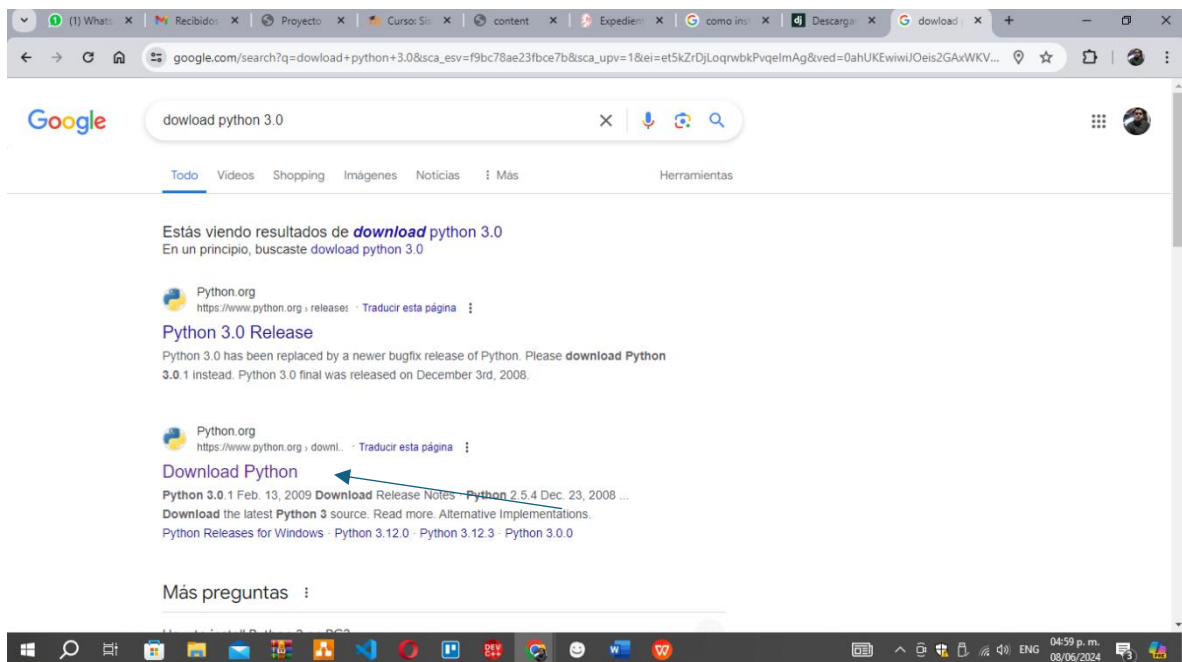
Tecnologías y Herramientas	Software seleccionado	Versión
Lenguaje de Programación	Python	3.0
Framework	Django Framework	5.0
Sistema Gestor de Base de Datos	SQL Server	2022
Entorno de Desarrollo	VS Studio	1.8
Gestión de Versiones	Git - GitHub	2.34
Administración de Proyecto	GitHub Project	2.34
Paquete de software libre	XAMPP	3.2.2
Pruebas Unitarias	Postman	10.2

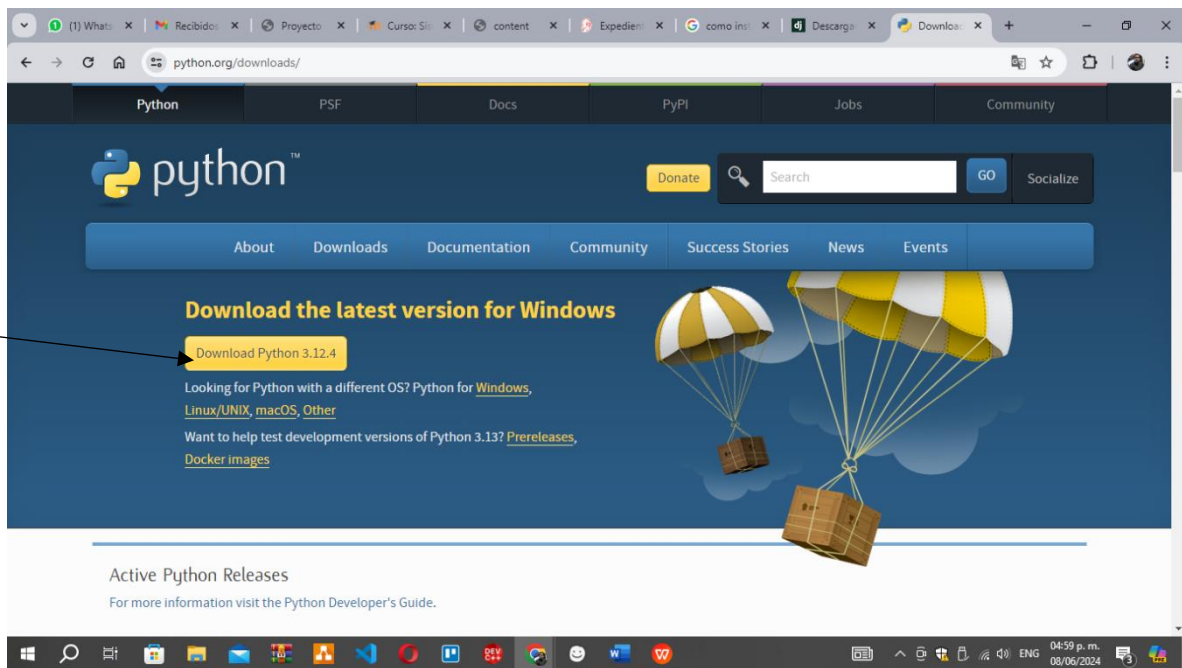
Instalación y configuración

🌐 Instalar python

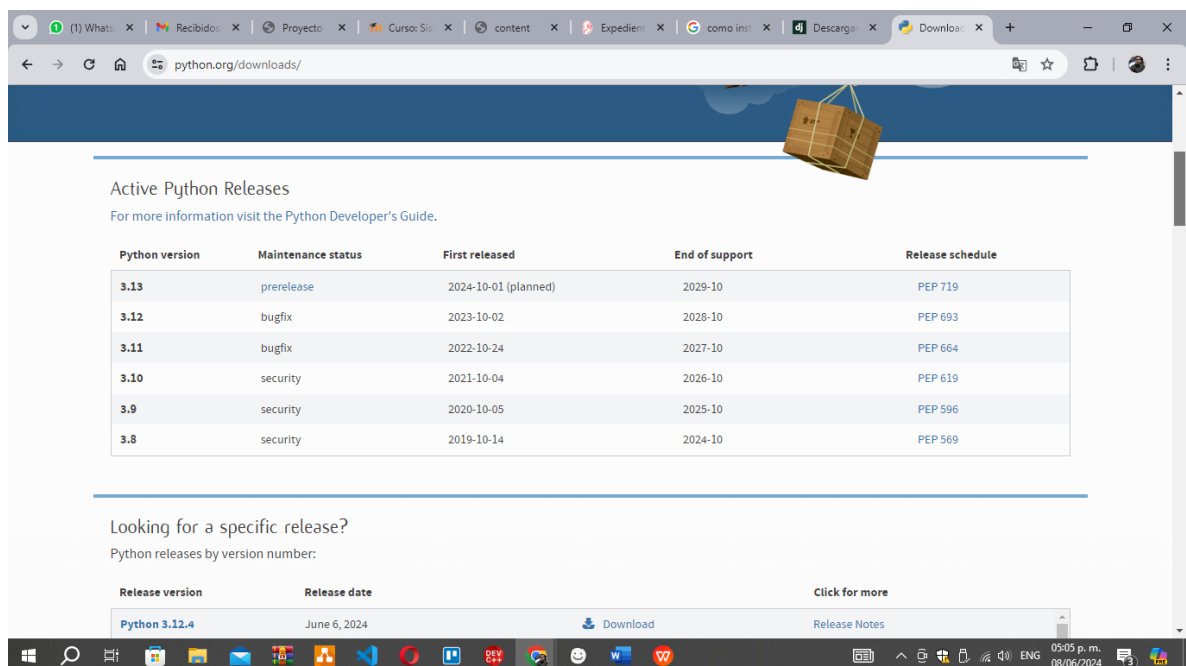
<https://www.python.org/downloads/>

Iniciamos colocando download python 3.0 o simplemente python para descargarlo



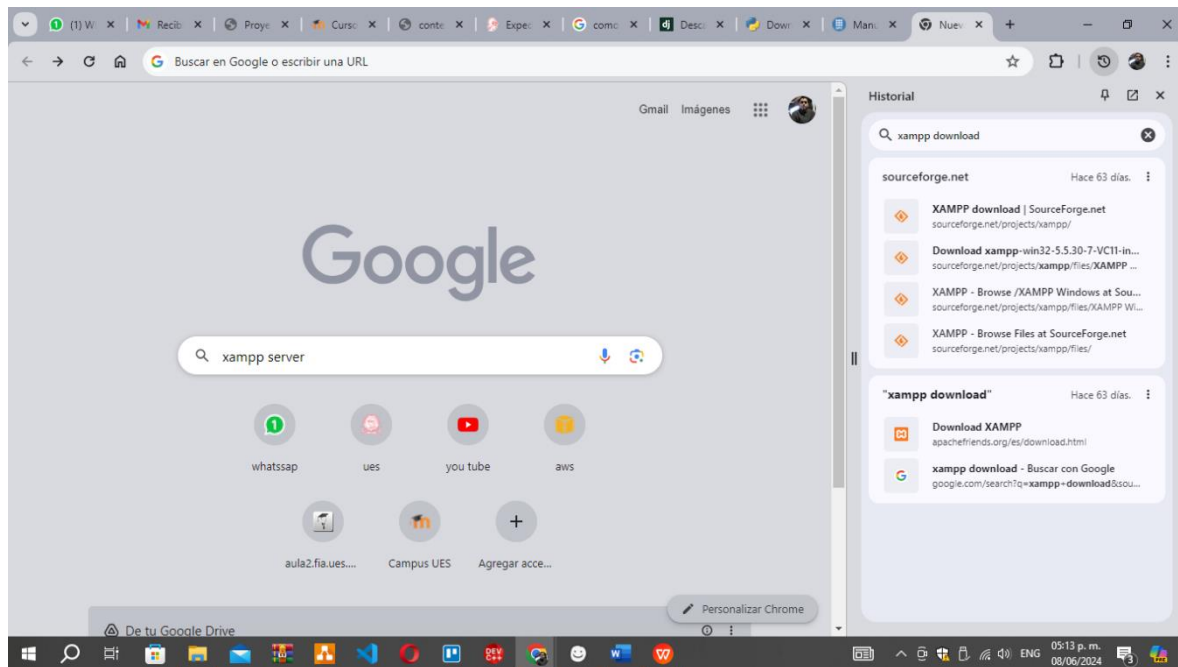


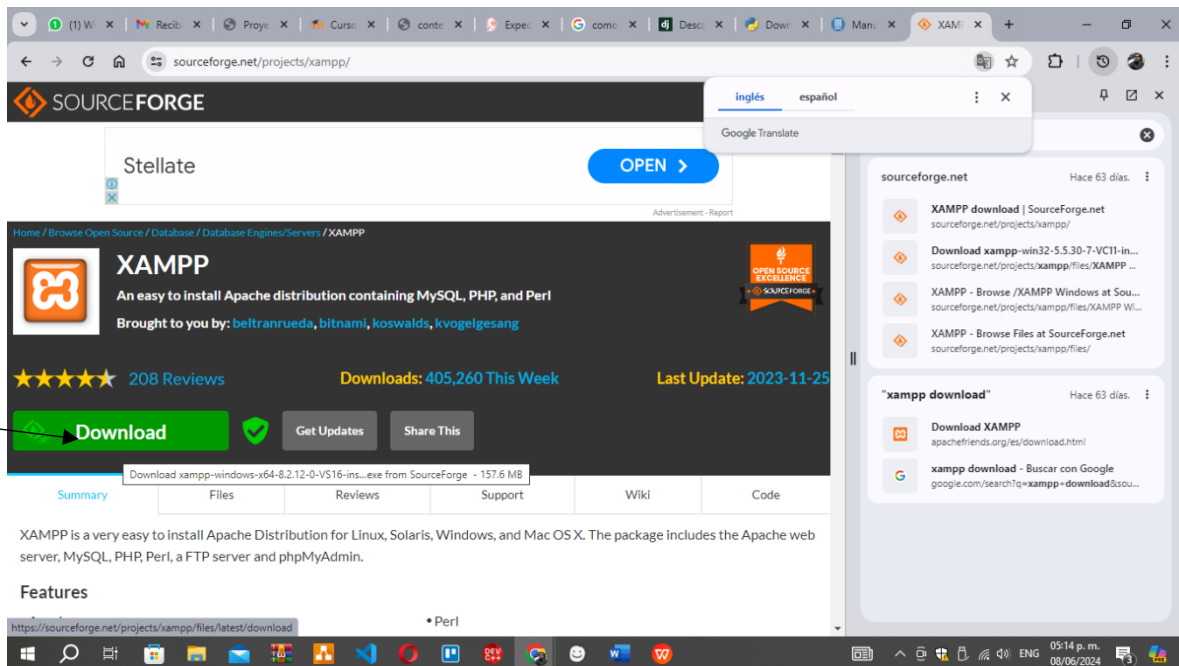
Al ingresar a la web podemos seleccionar la versión que mejor se acomode a nuestras necesidades.



🌐 Instalar XAMPP

Primero se descarga el XAMPP seleccionando el sistema operativo

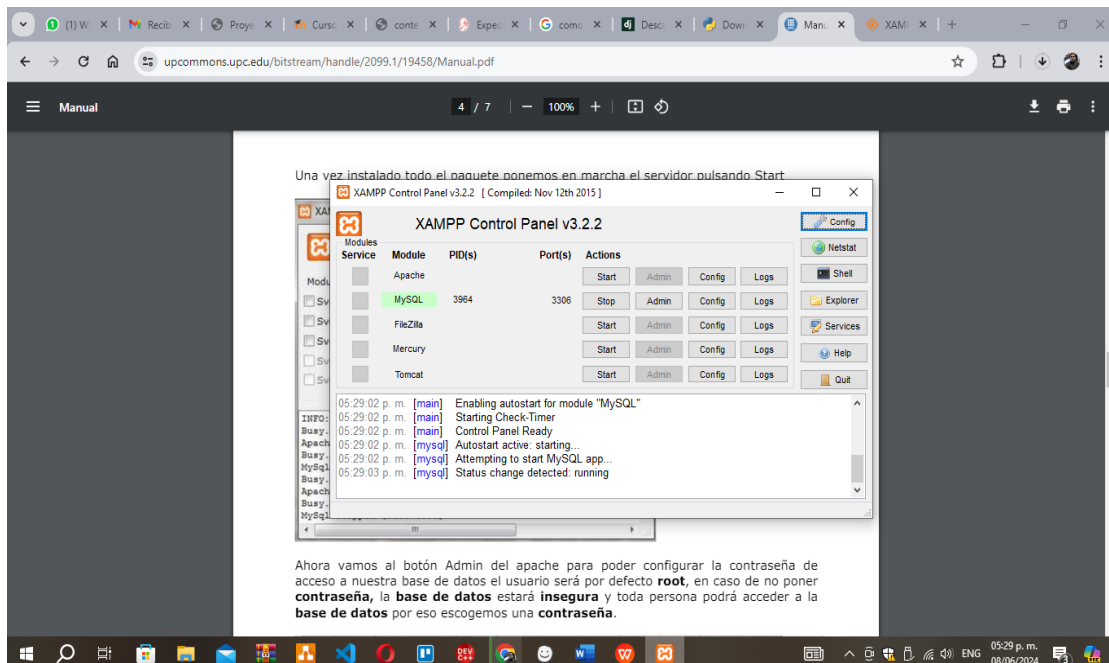
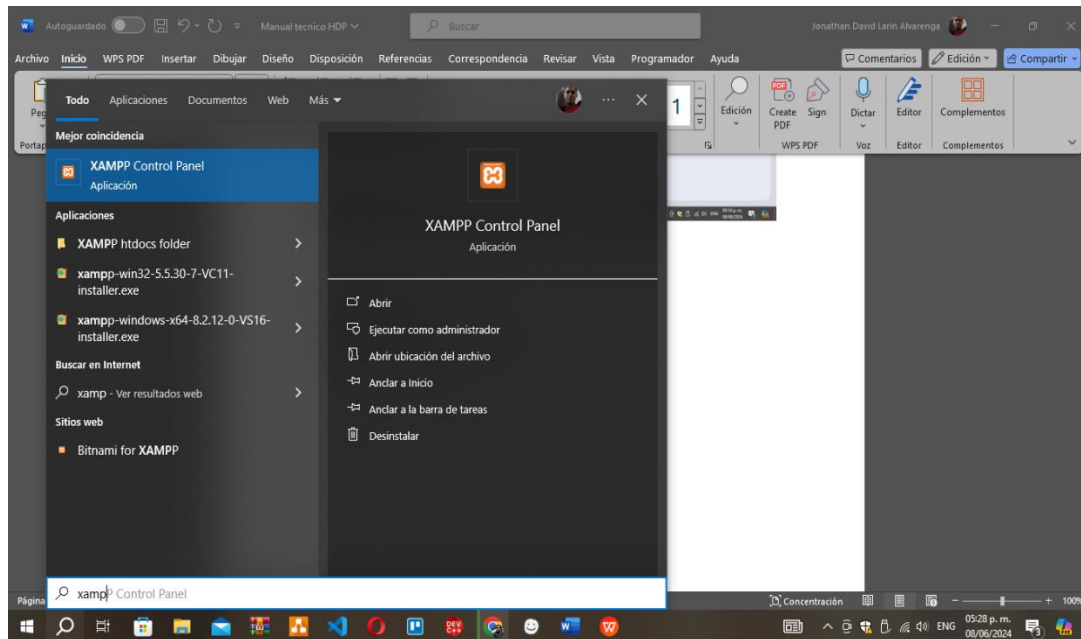




Después de descargar el archivo seguir los siguientes pasos:

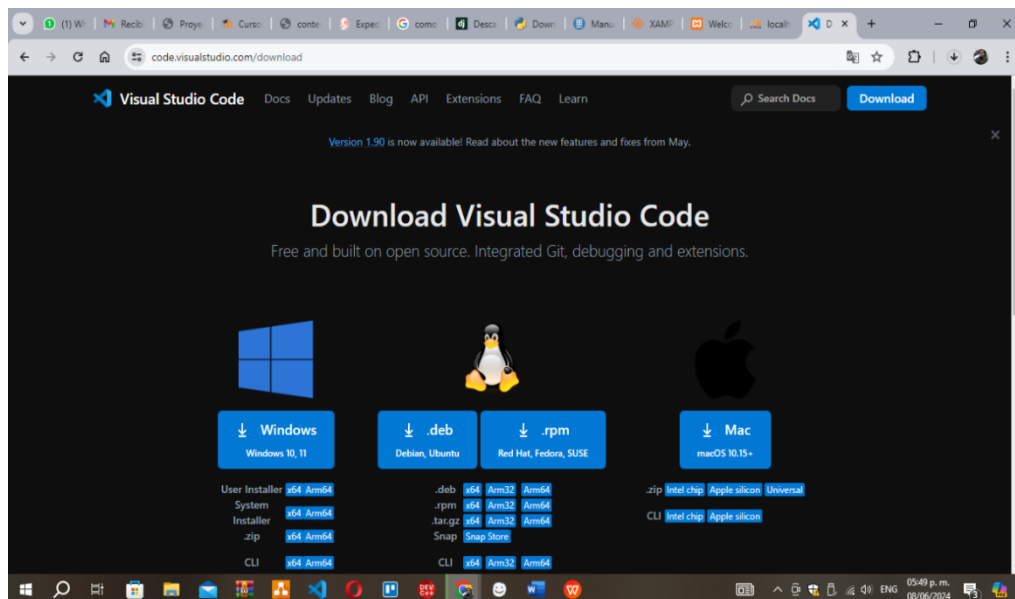
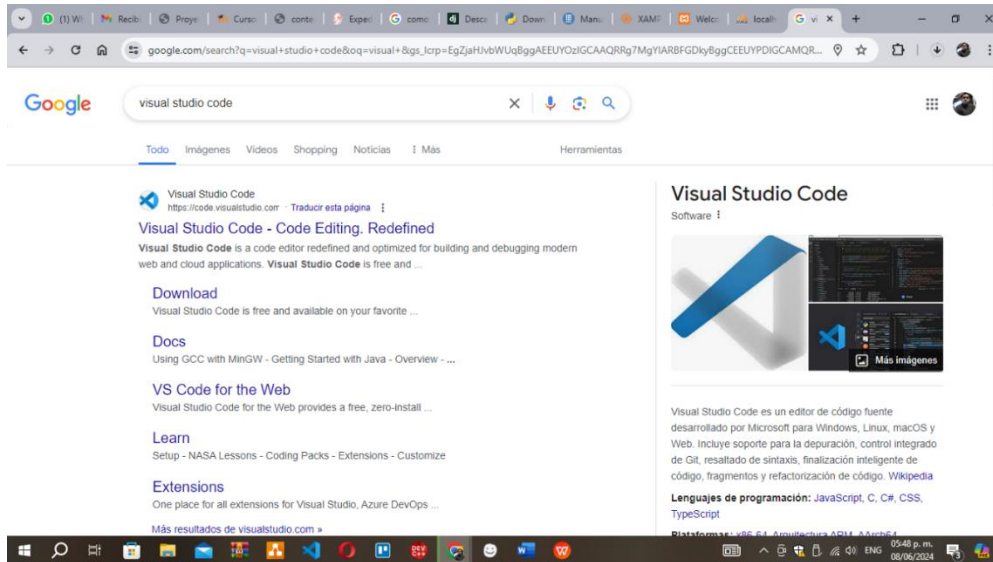
- ❖ Ejecutar el archivo de instalación
- ❖ Seleccionar idioma
- ❖ Aceptar
- ❖ Inicio de instalación
- ❖ Seleccionar directorio por defecto
- ❖ Terminar la instalación

Una vez instalado podemos encontrarlo de la siguiente manera



Instalar visual studio code

Buscamos en el buscador visual studio code y se descarga para
Proceder a instalar.



Crear un Entorno Virtual

Es recomendable crear un entorno virtual para tu proyecto Django para mantener las dependencias aisladas.

✓ Abrir la Línea de Comandos

Primero, abre la línea de comandos (cmd) en tu sistema.

✓ Navegar al Directorio del Proyecto

Usa el comando `cd` para navegar al directorio donde quieres crear el entorno virtual. Por ejemplo:

cmd

Copiar código

```
cd C:\ruta\al\tu\proyecto
```

✓ Crear el Entorno Virtual

Una vez en el directorio correcto, usa `virtualenv` o `venv` para crear el entorno virtual. `venv` es una herramienta incluida en las versiones de Python 3.3 y posteriores, mientras que `virtualenv` debe ser instalado a través de `pip`.

Usando `venv` (recomendado si usas Python 3.3+):

cmd

Copiar código

```
python -m venv nombre_del_entorno
```

Reemplaza nombre_del_entorno con el nombre que deseas dar a tu entorno virtual.

Usando virtualenv (si ya lo tienes instalado):

Si no tienes virtualenv instalado, primero instálalo con:

cmd

Copiar código

```
pip install virtualenv
```

Luego crea el entorno virtual:

cmd

Copiar código

```
virtualenv nombre_del_entorno
```

Activar el Entorno Virtual

Después de crear el entorno virtual, necesitas activarlo. La activación depende del sistema operativo que estés utilizando:

En Windows:

cmd

Copiar código

nombre_del_entorno\Scripts\activate

En macOS y Linux:

bash

Copiar código

source nombre_del_entorno/bin/activate

5. Verificar que el Entorno Virtual está Activado

Cuando el entorno virtual esté activado, deberías ver el nombre del entorno en el prompt de la línea de comandos. Por ejemplo, debería parecerse a esto en Windows:

cmd

Copiar código

(nombre_del_entorno) C:\ruta\al\tu\proyecto>

✓ Instalar Paquetes en el Entorno Virtual

Ahora puedes instalar paquetes dentro de tu entorno virtual utilizando pip.

Por ejemplo, para instalar Django, ejecuta:

cmd

Copiar código

pip install django

✓ Desactivar el Entorno Virtual

Para desactivar el entorno virtual y volver al entorno global de Python, simplemente usa el comando:

cmd

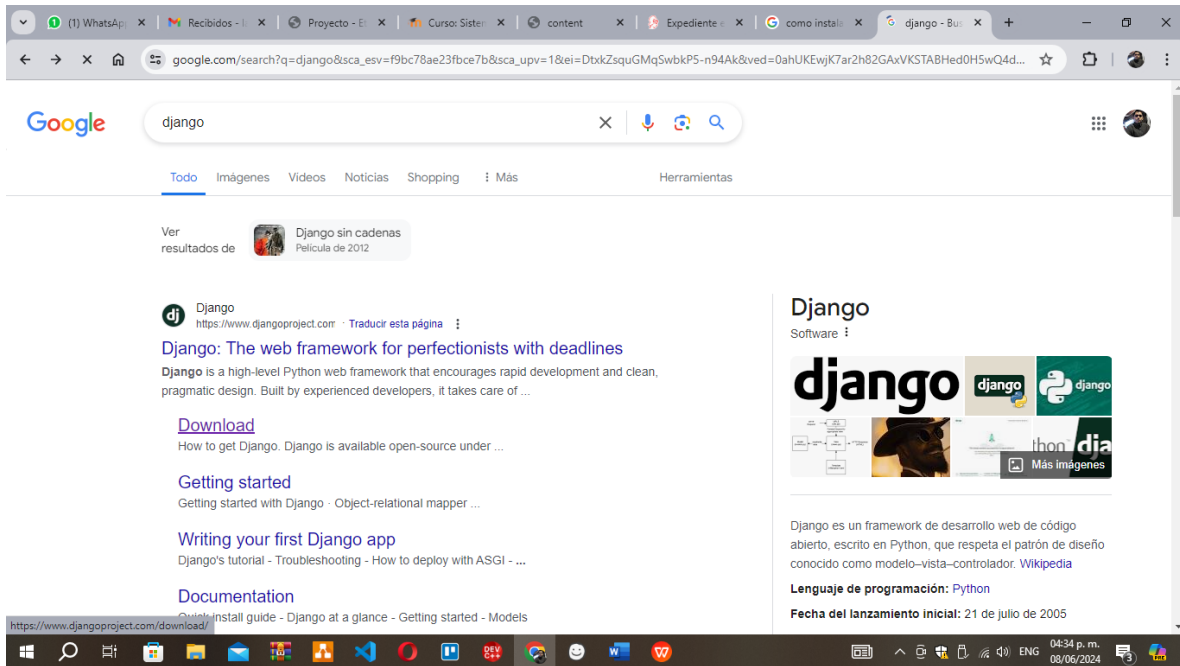
Copiar código

deactivate

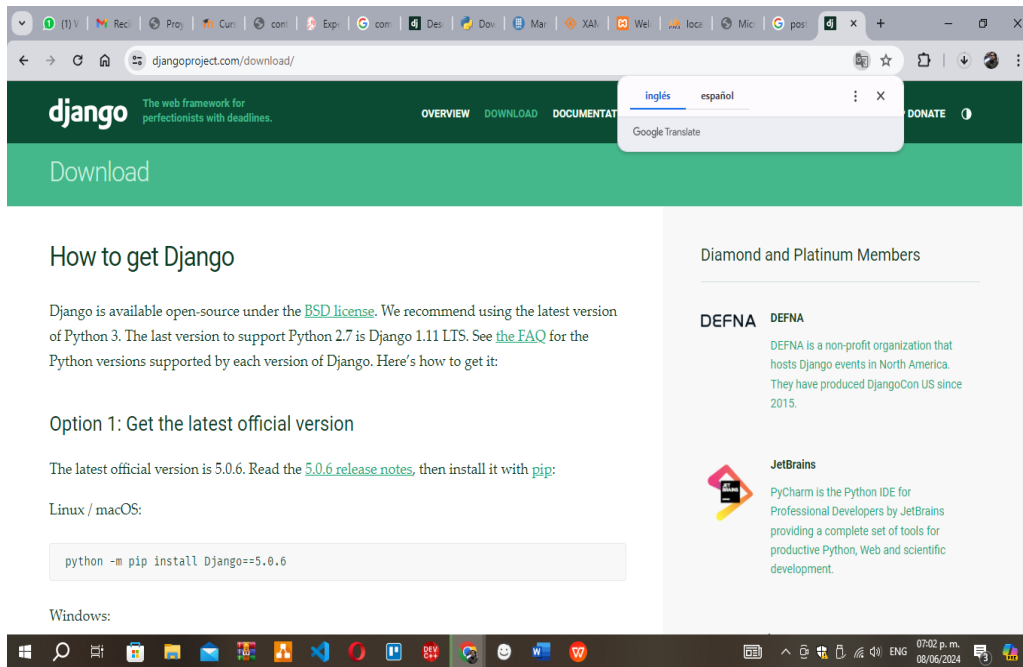
Instalar Django

Paso 1 colocar en el buscador Django para ver las instrucciones de su instalación

<https://www.djangoproject.com/download/>



En la página oficial aparecen los comandos como instalar Django para cada sistema operativo



Resumen de Comandos

1. Verificar Python:

cmd

Copiar código

python --version

2. **Actualizar pip:**

cmd

Copiar código

```
python -m pip install --upgrade pip
```

3. **Crear Entorno Virtual:**

cmd

Copiar código

```
python -m venv nombre_del_entorno
```

4. **Activar Entorno Virtual:**

cmd

Copiar código

```
nombre_del_entorno\Scripts\activate
```

5. **Instalar Django:**

cmd

Copiar código

```
pip install django
```

6. **Verificar Instalación de Django:**

cmd

Copiar código

```
django-admin --version
```

7. Crear Proyecto Django:

```
cmd
```

Copiar código

```
django-admin startproject nombre_del_proyecto
```

8. Navegar al Directorio del Proyecto:

```
cmd
```

Copiar código

```
cd nombre_del_proyecto
```

9. Ejecutar Servidor de Desarrollo:

```
cmd
```

Copiar código

```
python manage.py runserver
```

10. Desactivar Entorno Virtual:

```
cmd
```

Copiar código

```
deactivate
```

Siguiendo estos pasos, tendrás Django instalado y un proyecto básico en funcionamiento.

Instalar conector Mysql

Resumen de Pasos

1. **Abrir la Línea de Comandos.**
2. **Activar el Entorno Virtual:**

cmd

Copiar código

nombre_del_entorno\Scripts\activate

3. **Instalar mysqlclient:**

cmd

Copiar código

pip install mysqlclient

4. **Configurar la Base de Datos en settings.py:**

python

Copiar código

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.mysql',  
        'NAME': 'nombre_de_tu_base_de_datos',  
        'USER': 'tu_usuario',  
        'PASSWORD': 'tu_contraseña',  
        'HOST': '127.0.0.1',  
        'PORT': '3306',  
    }  
}
```

5. Crear la Base de Datos en MySQL:

sql

Copiar código

```
CREATE DATABASE nombre_de_tu_base_de_datos;
```

6. Aplicar las Migraciones:

cmd

Copiar código

```
python manage.py makemigrations
```

```
python manage.py migrate
```

7. Crear un Superusuario (Opcional):

cmd

Copiar código

```
python manage.py createsuperuser
```


8. Ejecutar el Servidor de Desarrollo:

cmd

Copiar código

```
python manage.py runserver
```

Siguiendo estos pasos, habrás configurado Django para trabajar con una base de datos MySQL utilizando el conector mysqlclient.

 Se crea el proyecto django

Resumen de Comandos

1. Verificar Python:

cmd

Copiar código

```
python --version
```

2. Actualizar pip:

cmd

Copiar código

```
python -m pip install --upgrade pip
```

3. Crear Entorno Virtual:

cmd

Copiar código

```
python -m venv nombre_del_entorno
```

4. Activar Entorno Virtual:

cmd

Copiar código

```
nombre_del_entorno\Scripts\activate
```

5. Instalar Django:

cmd

Copiar código

```
pip install django
```

6. Crear Proyecto Django:

cmd

Copiar código

```
django-admin startproject nombre_del_proyecto
```

7. Navegar al Directorio del Proyecto:

cmd

Copiar código

```
cd nombre_del_proyecto
```

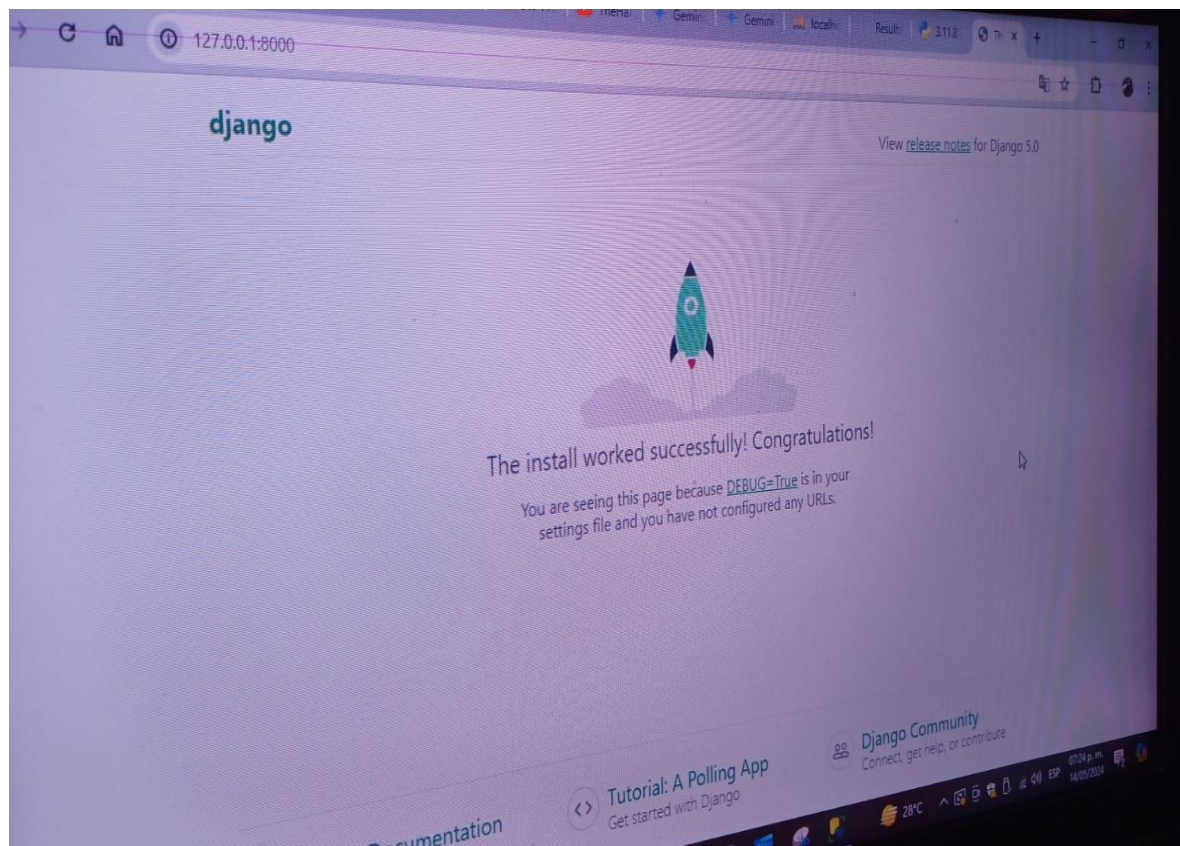
8. Ejecutar Servidor de Desarrollo:

cmd

Copiar código

```
python manage.py runserver
```

Deberías ver un mensaje indicando que el servidor está corriendo y puedes visitar `http://127.0.0.1:8000/` en tu navegador para ver la página de bienvenida de Django



9. Crear Aplicación Django (Opcional):

cmd

Copiar código

```
python manage.py startapp nombre_de_la_aplicacion
```

Siguiendo estos pasos, habrás creado un proyecto Django y podrás comenzar a desarrollar tu aplicación web.

Configurar base de datos

1. Instalar SQL Server Management Studio (SSMS)

Si aún no tienes instalado SSMS, descárgalo e instálalo desde el sitio web oficial de Microsoft.

2. Conectarse a un Servidor

- Abre SQL Server Management Studio.
- En la ventana de "Conectar con el servidor", ingresa la información necesaria para conectarte al servidor de bases de datos. Esto puede incluir el nombre del servidor, el tipo de autenticación (Windows o SQL Server), y tus credenciales de inicio de sesión.

3. Crear una Nueva Base de Datos

Una vez que te hayas conectado al servidor, puedes crear una nueva base de datos:

- Haz clic derecho en la carpeta "Bases de datos" en el Explorador de objetos.
- Selecciona "Nueva base de datos".
- Ingresa el nombre de la nueva base de datos y configura cualquier otra opción necesaria, como el tamaño del archivo y el crecimiento automático.

4. Crear Tablas y Definir Esquemas

Dentro de tu nueva base de datos, puedes crear tablas y definir los esquemas de tu base de datos:

- Haz clic derecho en la carpeta "Tablas" en el Explorador de objetos.
- Selecciona "Nueva tabla".
- Define las columnas de tu tabla, especificando el nombre de la columna, el tipo de datos y cualquier restricción adicional necesaria.
- Guarda la tabla.

5. Insertar Datos

Una vez que hayas creado tus tablas, puedes insertar datos en ellas:

- Haz clic derecho en la tabla en el Explorador de objetos.
- Selecciona "Editar 200 primeras filas" o "Seleccionar las 1000 primeras filas" para abrir la ventana de edición de datos.
- Ingresa los datos en las filas disponibles.

6. Consultar y Manipular Datos

Utiliza SQL Server Management Studio para escribir consultas SQL y manipular datos en tu base de datos:

- Abre una nueva consulta en el Editor de consultas.
- Escribe tus consultas SQL para seleccionar, insertar, actualizar o eliminar datos en tu base de datos.
- Ejecuta las consultas y revisa los resultados.

7. Realizar Copias de Seguridad y Restauración

Es importante realizar copias de seguridad de tu base de datos regularmente para proteger tus datos:

- Haz clic derecho en la base de datos en el Explorador de objetos.
- Selecciona "Tareas" > "Realizar copia de seguridad".
- Sigue el asistente para realizar la copia de seguridad de la base de datos.

8. Configurar Seguridad y Permisos

Puedes configurar la seguridad y los permisos de tu base de datos para controlar quién puede acceder y realizar operaciones en ella:

- Utiliza el Administrador de seguridad para agregar usuarios y roles.
- Asigna permisos específicos a usuarios y roles en tu base de datos.

Con estos pasos, habrás configurado y comenzado a trabajar con una base de datos en SQL Server Management Studio.

Instalar el Conector MySQL (mysqlclient):

cmd

Copiar código

nombre_del_entorno\Scripts\activate

pip install mysqlclient

Configurar Django para Usar MySQL:

- a. Abrir settings.py
- b. Modificar la sección DATABASES:

python

Copiar código

```
DATABASES = {  
    'default': {  
        'ENGINE': 'django.db.backends.mysql',  
        'NAME': 'nombre_de_tu_base_de_datos',  
        'USER': 'tu_usuario',  
        'PASSWORD': 'tu_contraseña',  
        'HOST': '127.0.0.1',  
        'PORT': '3306',  
    }  
}
```

Crear la Base de Datos en MySQL:

- a. Iniciar MySQL en XAMPP
- b. Crear la base de datos usando phpMyAdmin

Aplicar Migraciones en Django:

cmd

Copiar código

```
python manage.py makemigrations
```

```
python manage.py migrate
```

Crear un Superusuario (Opcional):

cmd

Copiar código

```
python manage.py createsuperuser
```


Ejecutar el Servidor de Desarrollo:

cmd

Copiar código

```
python manage.py runserver
```

Siguiendo estos pasos, habrás configurado tu proyecto Django para usar una base de datos MySQL.

 Iniciar el servidor de desarrollo Django

Resumen de Comandos

1. Navegar al Directorio del Proyecto:

cmd

Copiar código

```
cd C:\ruta\al\tu\proyecto
```

2. Activar el Entorno Virtual:

- En Windows:

cmd

Copiar código

```
nombre_del_entorno\Scripts\activate
```

- En macOS y Linux:

bash

Copiar código

```
source nombre_del_entorno/bin/activate
```

3. Iniciar el Servidor de Desarrollo:

cmd

Copiar código

```
python manage.py runserver
```

4. Opciones Adicionales:

- Cambiar el puerto:

cmd

Copiar código

```
python manage.py runserver 8080
```

- Hacer el servidor accesible externamente:

cmd

Copiar código

```
python manage.py runserver 0.0.0.0:8000
```

5. Detener el Servidor de Desarrollo:

Presiona CTRL+C en la línea de comandos.

Siguiendo estos pasos, podrás iniciar y detener el servidor de desarrollo en tu proyecto Django fácilmente.

 Configurar visual estudio code

1. Instalar Visual Studio Code

Si aún no tienes Visual Studio Code instalado, descárgalo e instálalo desde [su sitio web oficial](#).

2. Instalar Extensiones de VS Code

Para trabajar con Django de manera eficiente en VS Code, puedes instalar las siguientes extensiones desde el mercado de extensiones de VS Code:

- **Django Template:** Proporciona resaltado de sintaxis y snippets para archivos de plantilla de Django.
- **Python:** Proporciona soporte para Python, incluyendo resaltado de sintaxis, autocompletado y depuración.
- **Django:** Ofrece herramientas para trabajar con proyectos Django, incluyendo la capacidad de crear nuevas aplicaciones, ejecutar comandos de gestión y acceder al panel de administración de Django.
- **MySQL Syntax:** Proporciona resaltado de sintaxis para archivos SQL, que puede ser útil cuando estás escribiendo consultas SQL para tu base de datos MySQL.

Puedes instalar estas extensiones desde la pestaña "Extensions" en VS Code.

3. Configurar el Proyecto Django

Asegúrate de tener un proyecto Django configurado correctamente. Si no lo tienes, sigue los pasos para crear un proyecto Django como se mencionó en respuestas anteriores.

4. Configurar XAMPP

Si aún no has configurado XAMPP, sigue los pasos para instalarlo como se mencionó en respuestas anteriores.

5. Configurar la Base de Datos en Django

Asegúrate de que la configuración de la base de datos en tu proyecto Django (`settings.py`) esté correctamente establecida para usar MySQL con XAMPP, como se explicó anteriormente.

6. Ejecutar el Servidor de Desarrollo de Django

Abre la terminal en VS Code e inicia el servidor de desarrollo de Django con el comando:

```
bash
```

Copiar código

```
python manage.py runserver
```

7. Iniciar XAMPP

Abre XAMPP Control Panel y asegúrate de que Apache y MySQL estén ejecutándose.

8. Crear y Acceder a la Base de Datos MySQL

Usa phpMyAdmin para crear y gestionar la base de datos MySQL. Puedes acceder a phpMyAdmin visitando <http://localhost/phpmyadmin> en tu navegador web.

9. Desarrollar tu Aplicación Django

Ahora puedes desarrollar tu aplicación Django en VS Code, con el servidor de desarrollo de Django en funcionamiento y XAMPP proporcionando la base de datos MySQL.

10. Depurar y Ejecutar tu Aplicación

VS Code ofrece herramientas para depurar tu aplicación Django y ejecutar comandos de gestión desde la interfaz. Asegúrate de aprovechar estas características para un flujo de trabajo más eficiente.

Con estos pasos, deberías estar listo para configurar Visual Studio Code para trabajar con Django y XAMPP de manera efectiva.

Configurar xamp y SQL server

Configurar SQL y XAMPP implica preparar el entorno de desarrollo para ejecutar y administrar bases de datos MySQL utilizando el servidor local XAMPP. Aquí tienes los pasos detallados:

Iniciar los Servicios de MySQL en XAMPP

- Abre XAMPP Control Panel.
- Haz clic en los botones "Start" junto a "Apache" y "MySQL". Esto iniciará los servicios de servidor web y base de datos MySQL.

Acceder a phpMyAdmin

- Abre tu navegador web y visita <http://localhost/phpmyadmin>.
- Esto abrirá la interfaz web de phpMyAdmin, que te permite administrar tu base de datos MySQL.

Configurar Contraseña de Root para MySQL (Opcional)

- Por razones de seguridad, es recomendable configurar una contraseña para el usuario root de MySQL.
- En phpMyAdmin, ve a la pestaña "Usuarios" y haz clic en "Editar privilegios" para el usuario "root".
- Cambia la contraseña o configúrala si aún no está configurada.

Crear una Nueva Base de Datos (Opcional)

- Si necesitas crear una nueva base de datos, puedes hacerlo desde phpMyAdmin.
- En la pestaña "Bases de datos", haz clic en "Crear base de datos", ingresa el nombre de la base de datos y haz clic en "Crear".

Acceder a la Consola de MySQL (Opcional)

- Si prefieres trabajar con la línea de comandos, puedes acceder a la consola de MySQL.
- Abre la terminal o el símbolo del sistema y navega al directorio donde está instalado XAMPP.

- Ejecuta el comando `mysql -u root -p` y luego ingresa la contraseña de root cuando se te solicite.

Configurar SQL en tu Aplicación

- Si estás desarrollando una aplicación que utiliza una base de datos MySQL, asegúrate de configurar correctamente la conexión a la base de datos en tu aplicación.
- Utiliza las credenciales y la información de conexión (nombre de host, nombre de usuario, contraseña y nombre de la base de datos) proporcionadas por XAMPP o configuradas por ti mismo.

Con estos pasos, has configurado XAMPP y SQL en tu entorno local. Puedes empezar a desarrollar y trabajar con bases de datos MySQL utilizando XAMPP y administrarlas con phpMyAdmin o la consola de MySQL.

 Realizar migraciones de bases de datos

Resumen de los Pasos

1. Realizar cambios en los modelos en `models.py`.
2. Crear migraciones con `python manage.py makemigrations`.
3. Revisar las migraciones generadas.
4. Aplicar las migraciones con `python manage.py migrate`.
5. Verificar la base de datos para confirmar los cambios.

6. Realizar pruebas en tu aplicación para asegurarte de que todo funcione correctamente.

Siguiendo estos pasos, podrás realizar migraciones de bases de datos en tu proyecto Django de manera efectiva y segura.

Crear un super usuario

- ✓ Abre la línea de comandos (cmd).
- ✓ Navega hasta el directorio raíz de tu proyecto Django.
- ✓ Activa el entorno virtual de tu proyecto, si es que lo tienes configurado. Por ejemplo:

```
bash
```

Copiar código

```
nombre_del_entorno\Scripts\activate
```

Ejecuta el siguiente comando:

```
bash
```

Copiar código

```
python manage.py createsuperuser
```

Aparecerá un mensaje que te solicitará ingresar un nombre de usuario, una dirección de correo electrónico y una contraseña para el nuevo superusuario. Ingresa la información solicitada y presiona Enter después de cada campo.

plaintext

Copiar código

Username: tu_nombre_de_usuario

Email address: tu_correo_electronico@example.com

Password: *****

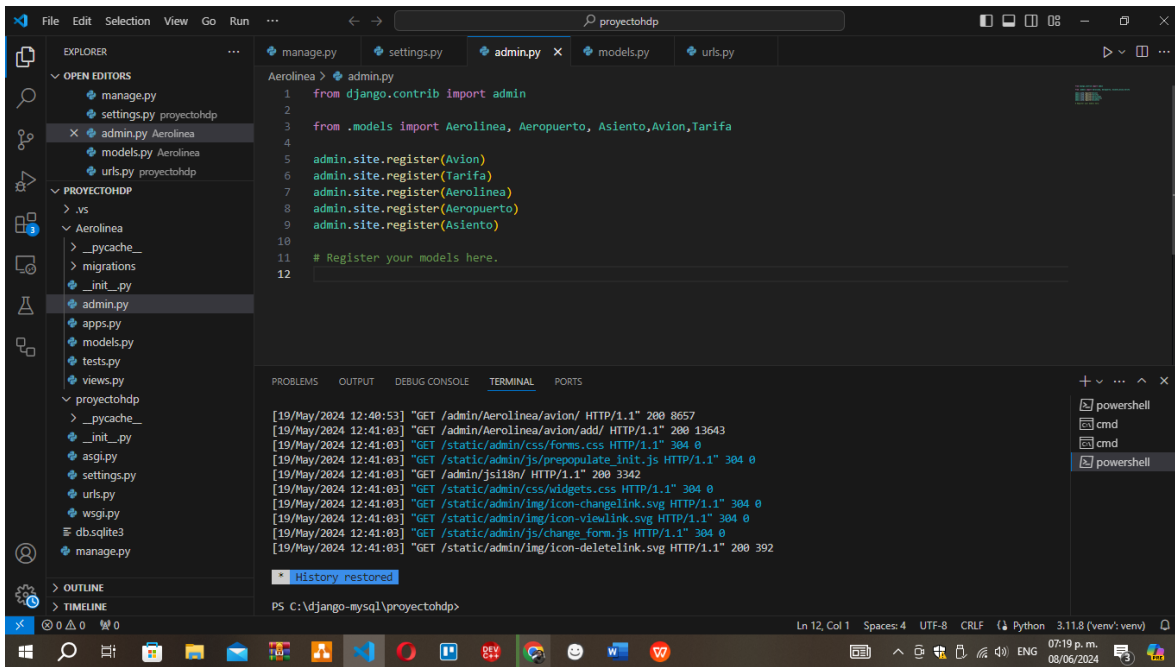
Password (again): *****

Después de ingresar la información solicitada, el superusuario se creará y recibirás un mensaje de confirmación.

Una vez creado el superusuario, puedes usar estas credenciales para acceder al panel de administración de Django y realizar tareas administrativas, como agregar, modificar o eliminar datos en la base de datos.

Siguiendo estos pasos, habrás creado un superusuario en tu proyecto Django y podrás comenzar a utilizarlo para administrar tu aplicación a través del panel de administración de Django.

Ejecutar el proyecto en visual studio code



Acceder al proyecto



- 🔧 **instalar Git:** Descarga e instala Git desde git-scm.com.
- 🔧 **Crear una Cuenta en GitHub:** Si aún no tienes una, regístrate en [GitHub](https://github.com).
- 🔧 **Configurar Git:** Configura tu nombre de usuario y dirección de correo electrónico en Git, que se utilizarán para identificar tus contribuciones en GitHub.

bash

Copiar código

```
git config --global user.name "Tu Nombre"
```

```
git config --global user.email "tu_correo@example.com"
```


Arquitectura de la solución

ARQUITECTURA DE LA SOLUCIÓN

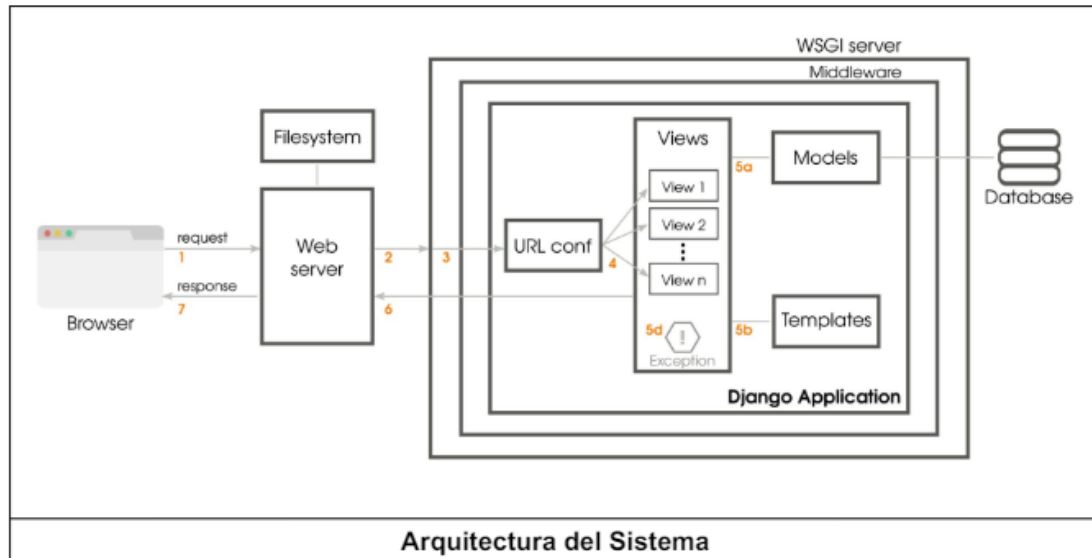
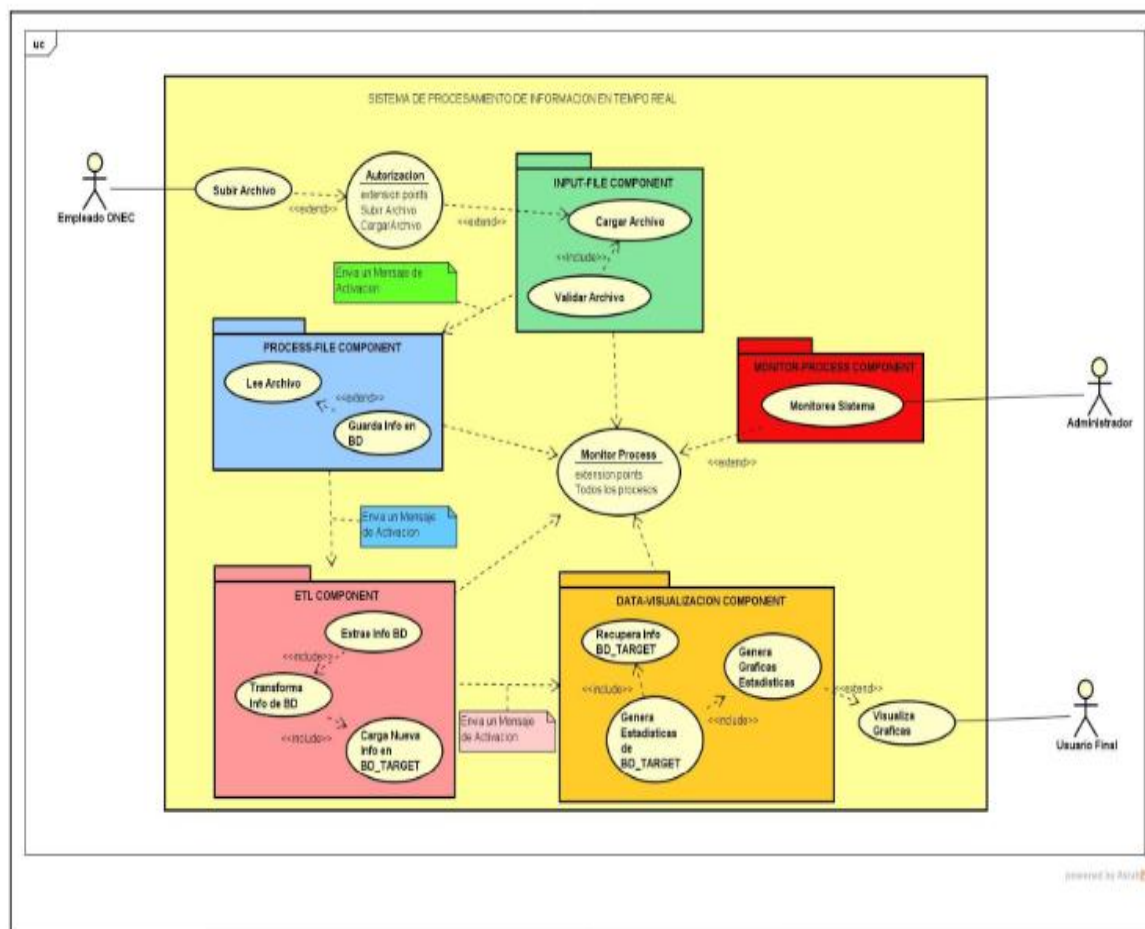


Diagrama de Caso de Uso General



Descripción del Caso de Uso

Nombre del Caso de Uso	Sistema de Procesamiento de Información en Tiempo Real		CU-000
Versión	1.0		
Dependencias	Input-File, Process-File, ETL, Data-Visualization		
Descripción	Empleado de ONEC sube un archivo		
Actores principales	Empleado de ONEC, Usuario Final		
Actores secundarios	Administrador		
Pre-condicion	El archivo subido con nueva información no debe estar vacio		
Secuencia Normal	Paso	Acción	
	1	Empleado sube un archivo al sistema	
	2	Input-File Component, carga el archivo	
	3	Input-File Component valida el archivo cargado	
	4	Input-File Component envía un mensaje de carga exitosa	
	5	Input-File Component envía un mensaje de activación a Process-File Component.	
	6	Process-File Component lee el archivo	
	7	Process-File Component guarda la información del archivo en Base de Datos	
	8	Process-File Component envía un mensaje de activación a ETL Component.	
	9	ETL Component extrae la información guarda en la Base de Datos	
	10	ETL Component transforma la información en datos numéricos	
	11	ETL Component carga la información en BD_TARGET	
	12	ETL Component envía un mensaje de activación a Data-Visualization Component.	
	13	Data-Visualization Component recupera la información de BD_TARGET	
	14	Data-Visualization Component genera estadísticas la información de BD_TARGET	
15	Data-Visualization Component genera graficas estadísticas para el usuario final.		
Post-Condicion	El Usuario Final visualiza graficas con información estadística		

Secuencia de Excepciones	Paso	Acción
	1	Empleado sube archivo incorrecto
	3	Input-File Component rechaza el archivo
	7	Process-File Component no se puede conectar a la Base de Datos
	9	ETL Component no puede extraer información
	13	Data-Visualization Component no puede recuperar información debido a que BD_TARGET esta vacía
Importancia	Alta	
Urgencia	Media	
Observaciones	N/A	
Comentarios	N/A	

DIAGRAMA GENERAL DE CLASE

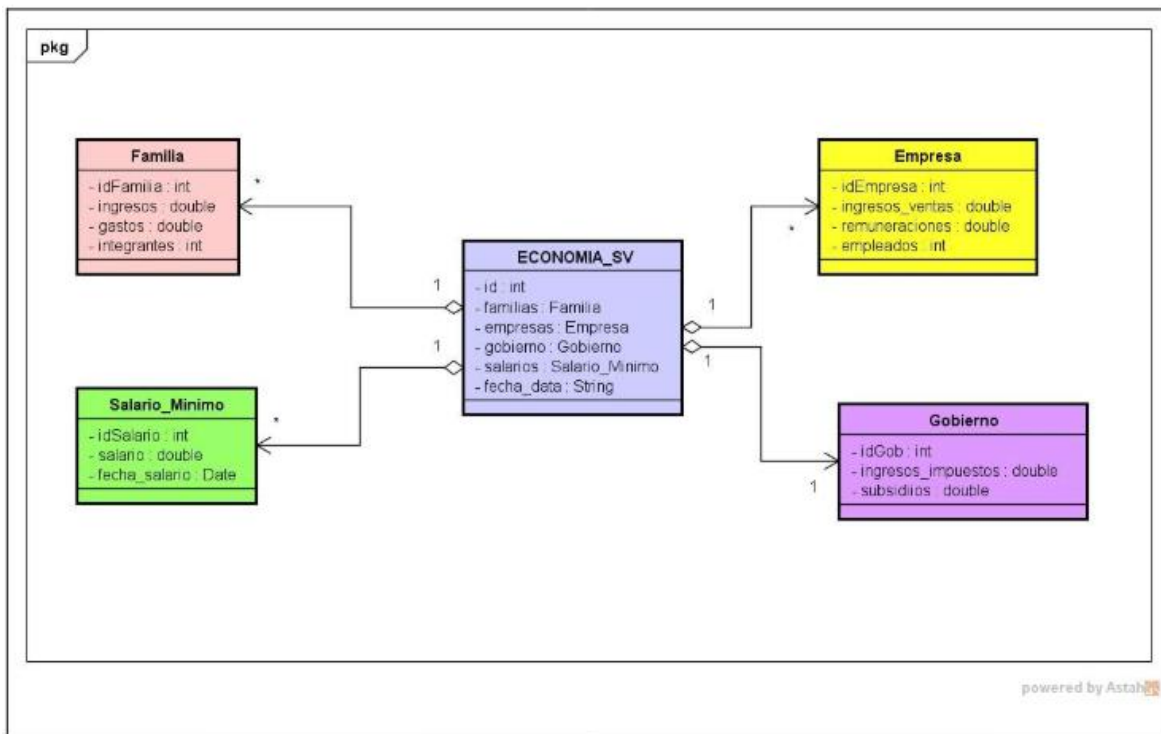
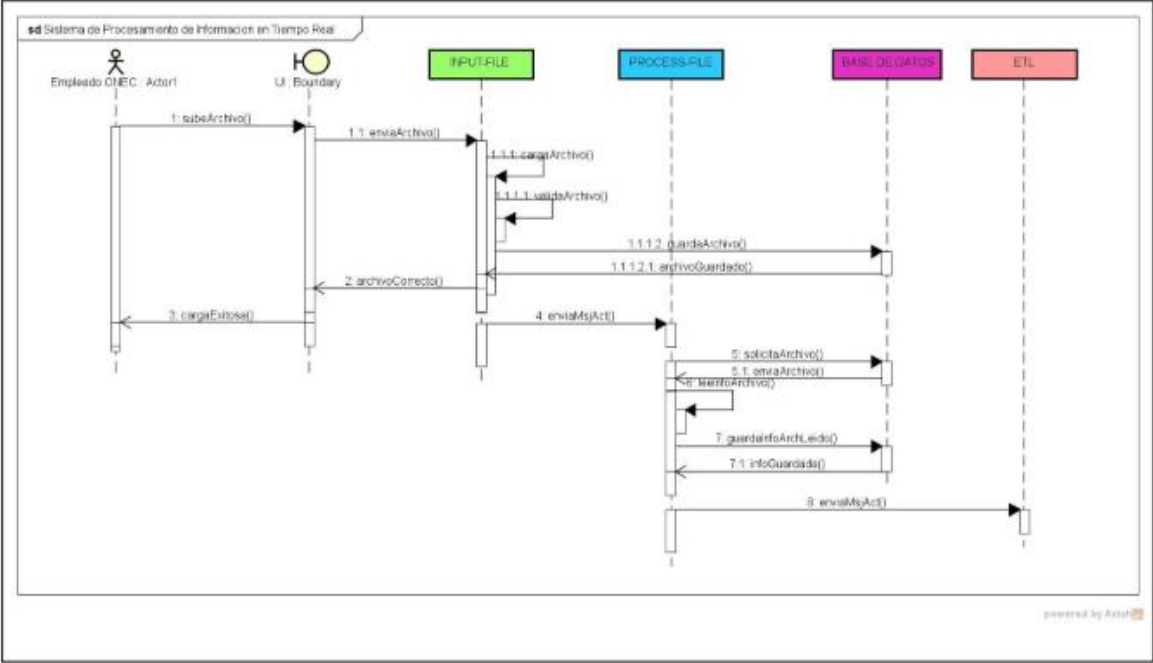


DIAGRAMA GENERAL DE SECUENCIA



Mantenimiento y actualizaciones.

Política de Mantenimiento:

- Realizamos mantenimiento preventivo semanalmente con el fin de garantizar un mejor rendimiento del sistema.
- El mantenimiento correctivo se lleva a cabo según sea necesario para abordar problemas críticos de seguridad o rendimiento.
- Se notificará a los usuarios sobre el mantenimiento programado mediante correos electrónicos y notificaciones en la aplicación.

Procedimientos de Actualización:

- ✓ Las actualizaciones se implementan durante los períodos de menor actividad para reducir al mínimo las interrupciones del servicio.
- ✓ Las actualizaciones se desarrollan en un entorno de prueba y se someten a pruebas rigurosas antes de su implementación.
- ✓ Realizamos copias de seguridad completas antes de aplicar cualquier actualización para minimizar el riesgo de pérdida de datos.

Gestión de Versiones:

- Utilizamos un sistema de versionado semántico para nuestras actualizaciones, utilizando herramientas como git y GitHub.
- Mantenemos un registro detallado de los cambios y nuevas características en cada versión para una transparencia completa.
- Las versiones antiguas serán retiradas después de un período de tiempo establecido, con soporte continuo para versiones más nuevas.

SOLUCION DE PROBLEMAS

Problemas de Configuración

1. Error en la configuración de la base de datos:

Solución: Asegúrate de Verificar las credenciales de la base de datos (nombre de usuario, contraseña, nombre de la base de datos) en el archivo de configuración coincidan con las configuraciones de tu servidor de base de datos. Asegúrate de que el servidor de base de datos esté funcionando.

2. Problemas con variables de entorno:

Solución: Asegúrate de que todas las variables de entorno necesarias estén definidas correctamente. Utiliza un archivo .env para definir las variables y asegúrate de que esté cargado correctamente en la aplicación.

Problemas de Rendimiento

3. Aplicación lenta:

Solución: Revisa el rendimiento de las consultas a la base de datos. Optimiza las consultas y considera el uso de índices en las tablas de la base de datos. Implementa técnicas de caché si es necesario.

Problemas de Datos

Datos incorrectos o incompletos:

Solución: Verifica la integridad de los datos que estás importando o ingresando a la base de datos. Asegúrate de que los datos históricos del salario mínimo y otros indicadores económicos estén actualizados y sean precisos.

Problemas de Seguridad

Inyecciones SQL:

Solución: Utiliza consultas preparadas o procedimientos almacenados en tu base de datos para prevenir inyecciones SQL.

Acceso no autorizado:

Solución: Implementa un sistema robusto de autenticación y autorización. Asegúrate de que las rutas y recursos sensibles estén protegidos adecuadamente.

Contacto para Soporte

Si encuentras un problema que no está cubierto en esta sección, por favor contacta a nuestro equipo de soporte:

- **Correo electrónico:** dg12003@ues.edu.sv , AM17040@ues.edu.sv,
la20005@ues.edu.sv
- **Teléfono:** +503 76872886
- **Foro Campus Ues:** grupo 4 del grupo teórico 1