



UNIVERSIDAD PRIVADA DE TACNA

FACULTAD DE INGENIERÍA

Escuela Profesional de Ingeniería de Sistemas

“Desarrollo de un Sistema de Gestion de Usuario usando Framework DJANGO en el lenguaje PYTHON”

Curso: Base de Datos II

Docente: Ing. Patrick Jose Cuadros Quiroga

Integrantes:

<i>Chambe Torres, Edgard Reynaldo</i>	<i>2019064917</i>
<i>Chata Choque, Brant Antony</i>	<i>2020067577</i>
<i>Nina Vargas, Luigui Augusto</i>	<i>2019065166</i>
<i>Arce Bracamonte, Sebastian</i>	<i>2019062986</i>
<i>Condori Vargas, Tomas Yoel</i>	<i>2018000487</i>

**Tacna – Perú
2023**

Resumen

Este proyecto presenta el desarrollo de un sistema de gestión de usuarios utilizando el framework Django y nuestro API creado en el lenguaje Python, donde un administrador puede crear roles y permisos para proceder a crear los usuarios y tanto el usuario creado y el admin podrán hacer uso del actualizar Perfil y ver su lista de usuarios ,estos permisos desempeñarán un papel fundamental en la asignación de las capacidades que los usuarios podrán llevar a cabo en el sistema. Esto proporcionará un control eficiente y seguro sobre los recursos y la funcionalidad del sistema que emplee este API.

Abstract

This project involves the development of a user management system using the Django framework and our API created in the Python language. In this system, an administrator can create roles and permissions to proceed with user creation. Both the created user and the admin can update their profiles and view the list of users. These permissions will play a fundamental role in assigning the capabilities that users can perform within the system. This will provide efficient and secure control over the resources and functionality of any system that utilizes this API.

1. introducción

El presente proyecto tiene por objetivo a la gestión efectiva de los usuarios la cual es esencial para desarrollar aplicaciones web seguras y eficientes. Este proyecto se centra en la creación de un Sistema de Gestión de Usuarios utilizando Django y Python. Permitirá a los administradores definir roles y permisos, facilitando la administración de cuentas de usuario y mejorando la experiencia del usuario final. Los permisos desempeñan un papel clave al asignar capacidades, garantizando un control eficiente y seguro de los recursos del sistema.

2. Título "Desarrollo de un Sistema de gestión de Usuarios usando framework DJANGO en el lenguaje PYTHON"

3. Autores

- *Chambe Torres, Edgard Reynaldo*
- *Chata Choque, Brant Antony*
- *Nina Vargas, Luigui Augusto*
- *Condori Vargas, Tomas Yoel*
- *Sebastian Arce Bracamonte*

4. Planteamiento del problema

4.1. Problema

El problema central que enfrenta el desarrollo de un Sistema de Gestión de Usuarios es garantizar la seguridad, eficiencia, satisfacción del usuario y cumplimiento normativo en aplicaciones web, así como la escalabilidad a medida que evoluciona un proyecto.

4.2. Justificación

El desarrollo de un Sistema de Gestión de Usuarios es esencial para abordar cuestiones críticas como seguridad, eficiencia, satisfacción del usuario y cumplimiento normativo en aplicaciones web. Al priorizar estos aspectos, podemos construir aplicaciones web más sólidas y confiables que satisfagan las expectativas de los usuarios y cumplan con las normativas aplicables.

4.3. Alcance

El proyecto aborda las necesidades y objetivos típicos de los sistemas de gestión de usuarios desarrollados con Django y Python, que son aspectos comunes en este tipo de sistemas.

5. Objetivos

5.1. General

- Desarrollo de un sistema de Gestión de Usuarios enfocado en la Administración con el objetivo principal de simplificar las tareas administrativas y mejorar la colaboración en el entorno que se utilice.

5.2. Específicos

- Implementar funcionalidades específicas para la gestión de usuarios de roles, permisos y creación de usuarios.
- Analizar las necesidades y requerimientos en relación con la gestión de usuarios.
- Permitir a los administradores asignar roles específicos.
- Simplificar las tareas administrativas mediante procesos automatizados.

6. Desarrollo de la propuesta

Caso de Uso API:

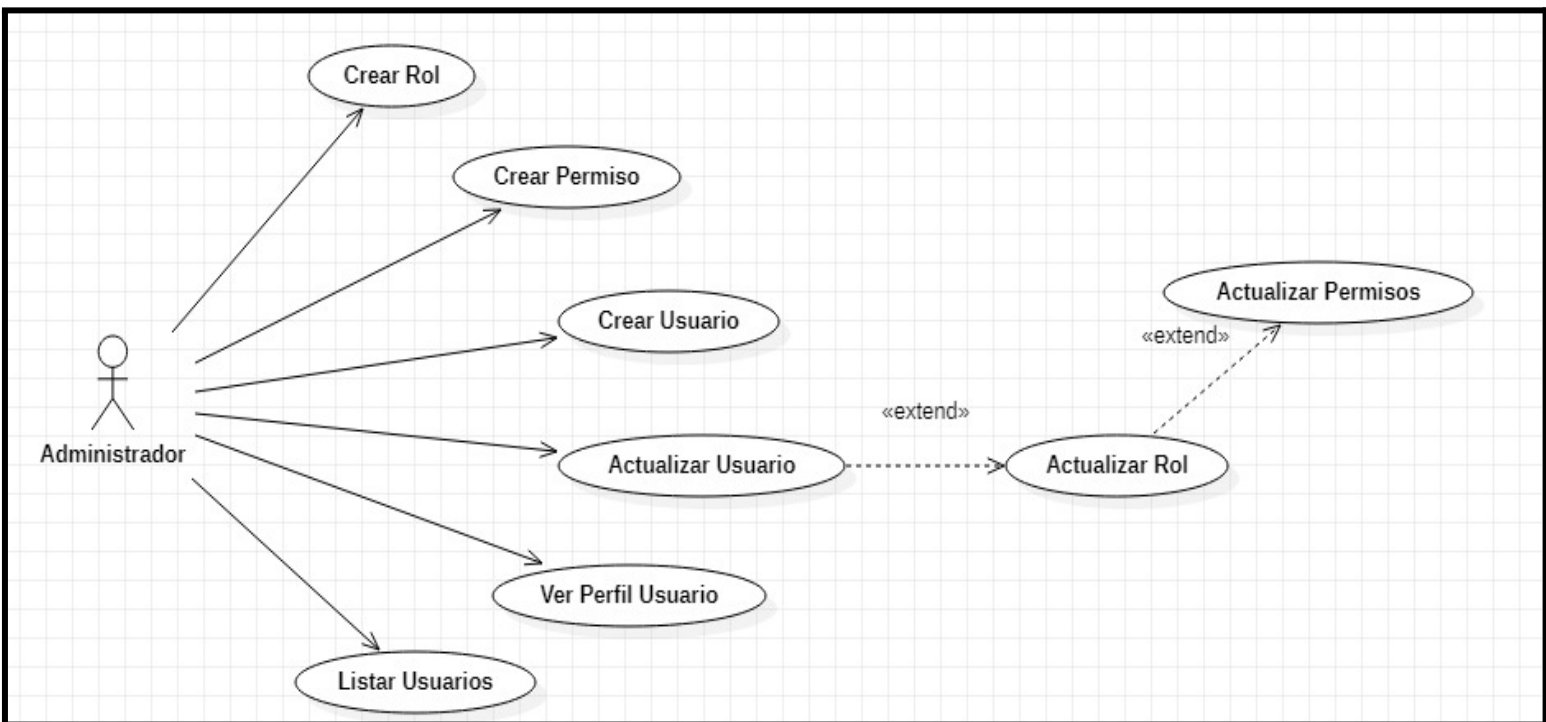


Diagrama de Clases

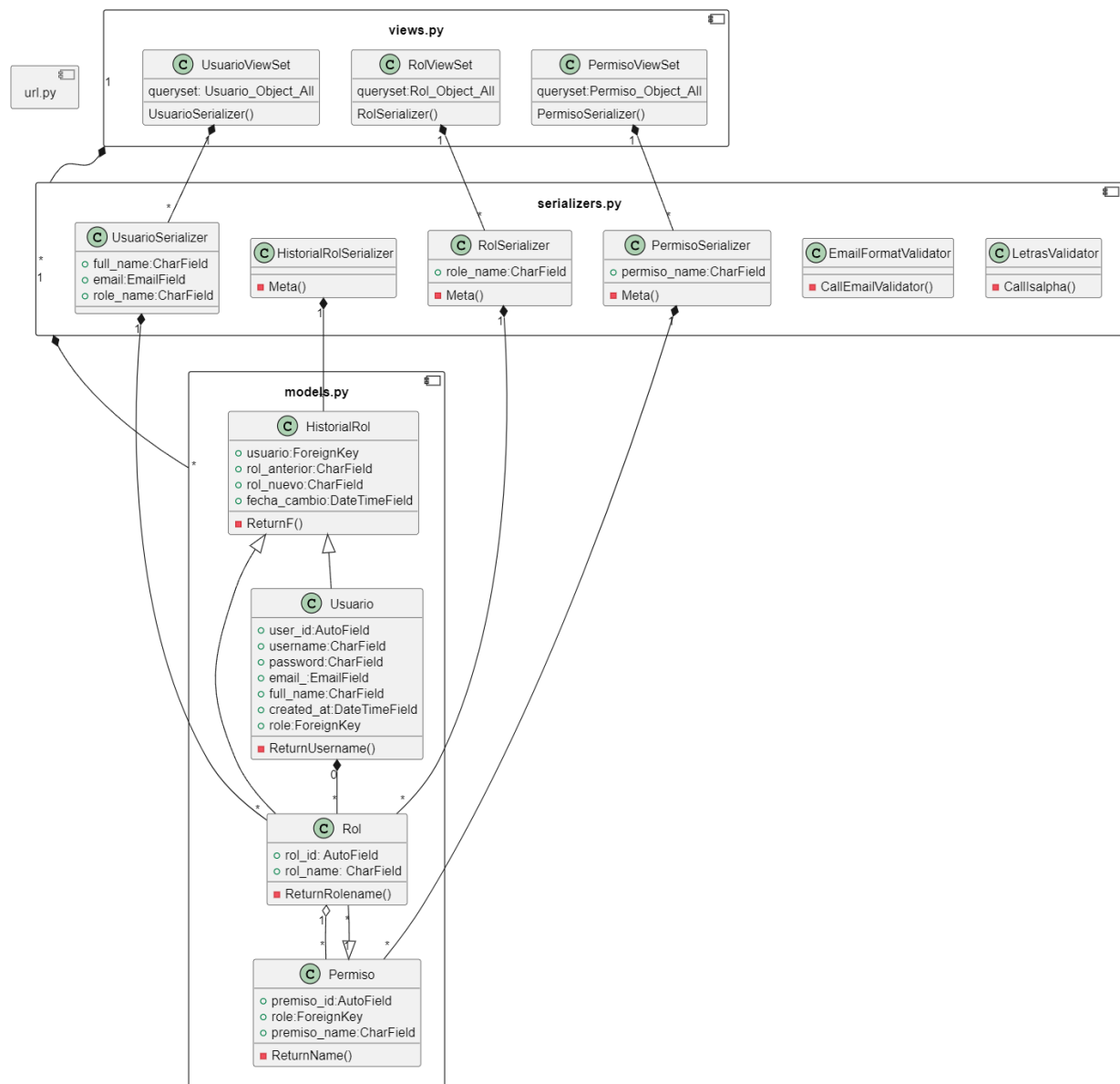
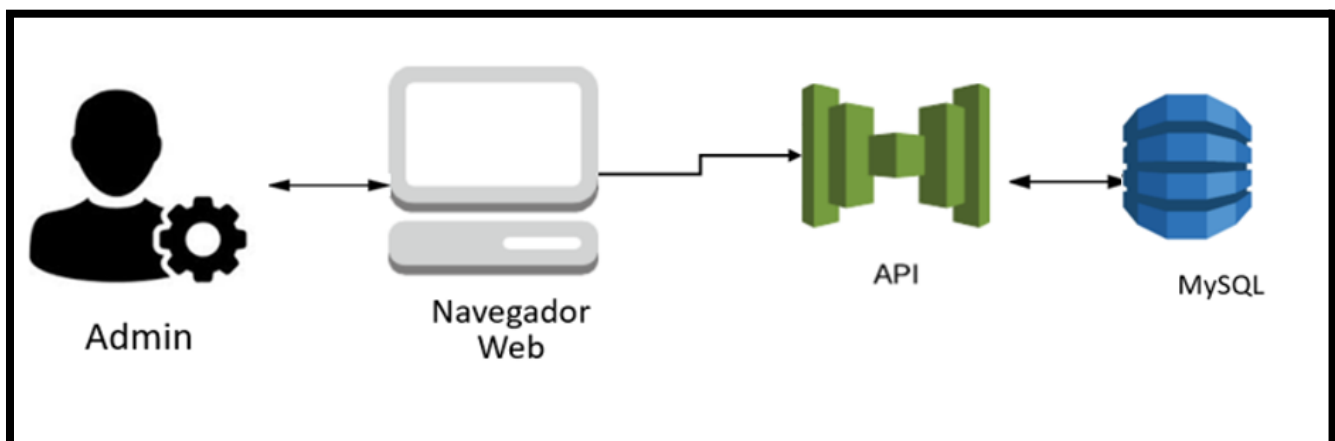
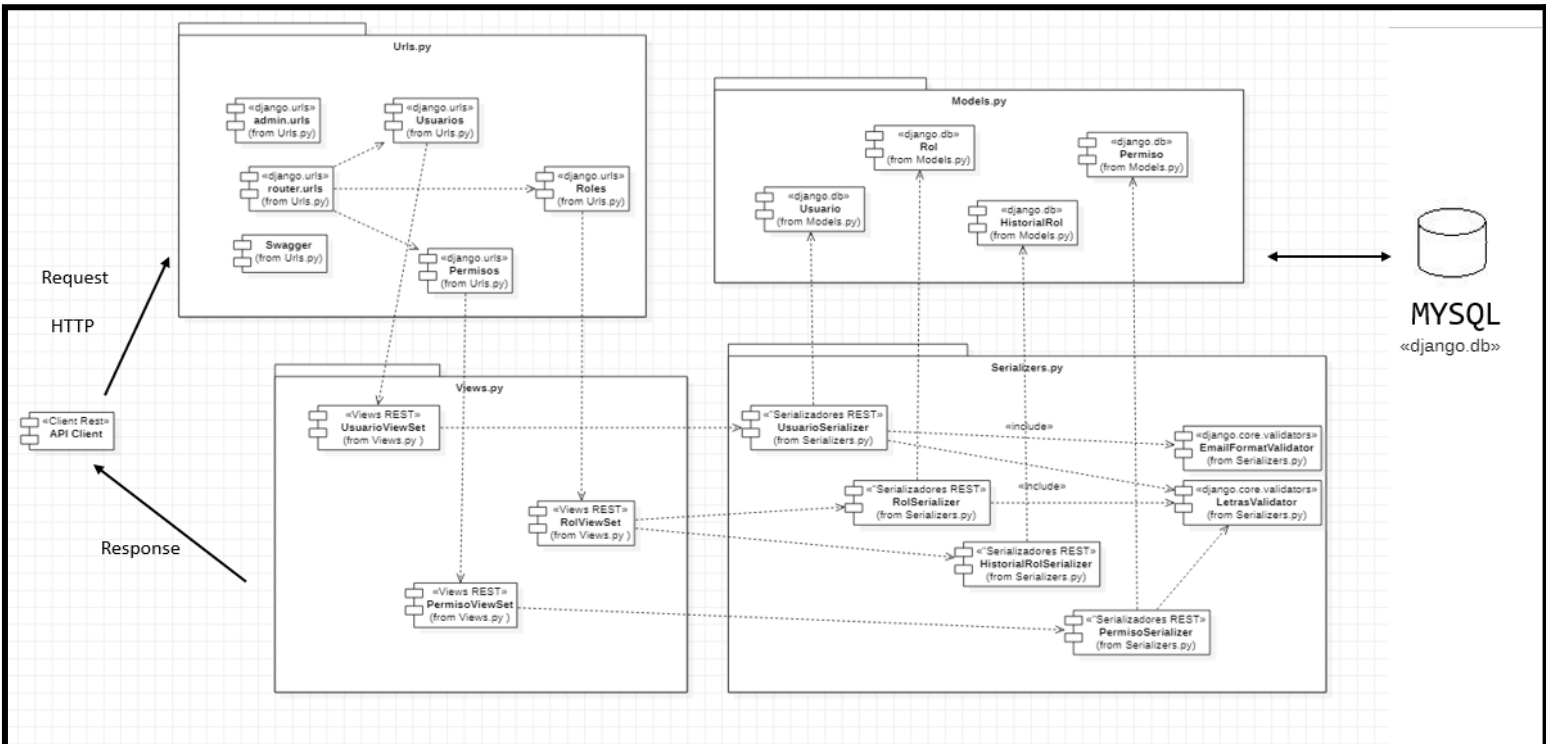


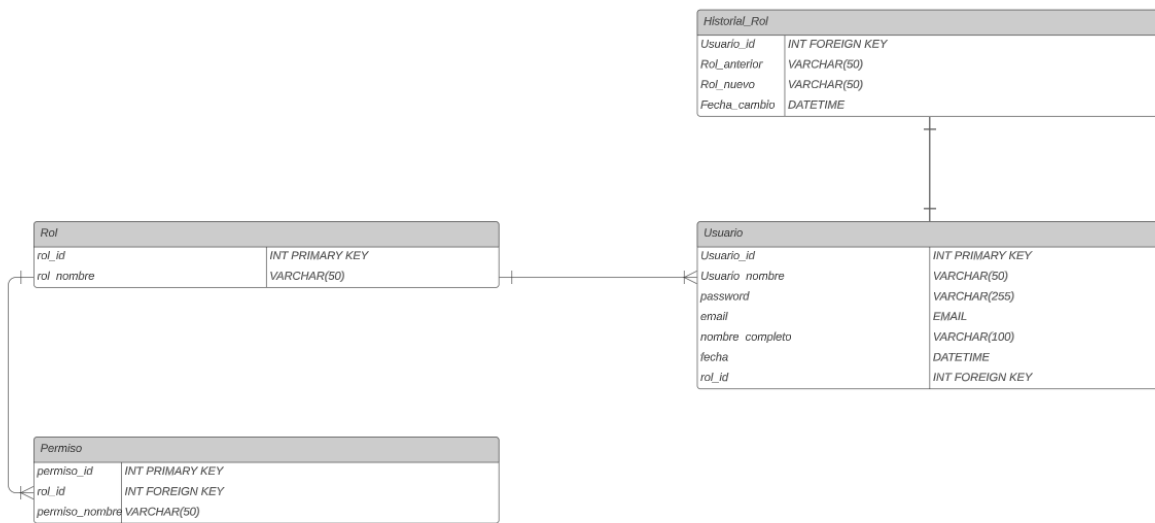
Diagrama de Arquitectura



6.2. Diagrama de Componentes



6.4. Diagrama de Base de Datos



7. Bibliografía

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

<https://www.freecodecamp.org/espanol/news/para-que-se-utiliza-django-de-python-5-razones-claves-por-las-que-uso-el-framework-django-para-proyectos/>

<https://www.django-rest-framework.org/>

<https://codigofacilito.com/articulos/orm-explicacion>

<https://pythondiario.com/tutorial-django-desde-cero>

8. Anexos

8.1 Diccionario de datos de su base de datos relacional

Tablas, con los siguientes campos:

Nombre de Columna | Descripción | Tipo de Dato | Permite nulos | Constraint

Nombre de la Tabla:			Rol			
Descripción de la Tabla:			Esta tabla tendrá los roles que se asignará a los usuarios.			
Objetivo:			Guardar datos de los roles.			
Relaciones con otras Tablas:			Permiso, Usuario			
Descripción de los campos						
Nro.	Nombre del campo	Tipo dato longitud	Permite nulos	Clave primaria	Clave foránea	Descripción del campo
1	role_id	Int	Not null	Si	No	Identifica el id único del rol
2	role_name	Char(50)	Not null	No	Si	Dato del nombre del rol

Nombre de la Tabla:			Permiso			
Descripción de la Tabla:			Esta tabla tendrá los datos de los permisos			
Objetivo:			Guardar datos de los permisos registrados			
Relaciones con otras Tablas:			Rol			
Descripción de los campos						
Nro.	Nombre del campo	Tipo dato longitud	Permite nulos	Clave primaria	Clave foránea	Descripción del campo
1	Permiso_id	Int	Not null	Si	No	Identifica el id único del permiso
2	role	Int	Not null	No	Si	Identificación del rol al que se le asigna el permiso
3	Permiso_name	Char(50)	Not null	No	No	Nombre asignado al permiso registrado

Nombre de la Tabla:			Usuario			
Descripción de la Tabla:			Esta tabla almacena los datos necesarios de los usuarios			
Objetivo:			Datos de los Usuarios			
Relaciones con otras Tablas:			Rol			

Descripción de los campos						
Nro.	Nombre del campo	Tipo dato longitud	Permite nulos	Clave primaria	Clave foránea	Descripción del campo
1	User_id	Int	Not null	Si	No	Identifica el id único de usuario
2	username	Char(50)	Not null	No	No	Nombre del usuario
3	Password	Char(255)	Not null	No	No	Contraseña de acceso del usuario
4	email	email	Not null	No	No	Correo del usuario
5	Full_name	Char(100)	Not null	No	No	Nombre completo del usuario
6	create_at	Datetime	Not null	No	No	Fecha de creación del usuario
7	role	Int	Not null	No	Si	Identificación del rol asignado al usuario