

**APTC106**

**Semana 6**

**Sumativa 2:**

**Propuesta de Aplicación**

Nombres integrantes:

FELIPE NICOLAS IGNACIO ARGOMEDO EMPARANZA  
ASTRID CORDERO ROJO  
GIOVANI DÍAZ MANRÍQUEZ  
KEVIN PAOLO ALESSANDRO GUARDA ALTAMIRANO

## Contenido

Introducción .....	3
Análisis Crítico .....	4
Implementación y Desarrollo .....	6
PASO 1: EJECUTAR PROYECTO .....	6
PASO 2: CONFIGURACIONES .....	8
PASO 3: PRUEBAS CRUD .....	10
PASO 4: OTRAS VISTAS.....	13
PASO 5: REVISION DE CÓDIGO .....	15
Conclusión .....	17

# Introducción

Hoy en día la salud mental ha cobrado mucha importancia en la vida de las personas, la necesidad de facilitar el acceso a servicios terapéuticos es más crítica que nunca. La pandemia global de COVID-19 ha incrementado los problemas de salud mental en todo el mundo, haciendo evidente la falta de recursos accesibles y personalizados para el bienestar emocional y psicológico. Las soluciones digitales han demostrado ser herramientas valiosas en la superación de barreras geográficas y económicas que muchas veces impiden a las personas buscar ayuda. Este panorama ofrece una oportunidad única para desarrollar plataformas que no solo brinden apoyo terapéutico, sino que también creen comunidades de cuidado y soporte mutuo.

El objetivo principal de este proyecto es implementar un CRUD basado en la propuesta del proyecto de salud mental utilizando los recursos de Django entregado en el ramo

Buscamos proporcionar un espacio seguro y privado para la terapia, garantizar que todos los intercambios y datos compartidos en la plataforma estén protegidos con los más altos estándares de seguridad para preservar la confidencialidad y la integridad de la información personal y de las sesiones. Facilitar la accesibilidad a servicios terapéuticos y diseñar una interfaz amigable y fácil de usar que permita a los usuarios registrarse, agendar sesiones, y participar en foros de discusión sin barreras técnicas complicadas.

Fomentar una comunidad de apoyo para crear espacios dentro de la aplicación donde los usuarios puedan compartir experiencias, recursos y obtener apoyo de pares, facilitando así un entorno de soporte continuo.

Integrar herramientas interactivas y recursos educativos desarrollando funcionalidades que incluyan la posibilidad de acceder a recursos multimedia y herramientas interactivas que pueden ser utilizadas de manera independiente o como complemento a las sesiones de terapia. Ofrecer seguimiento y personalización en el tratamiento para implementar un sistema que permita a los terapeutas y pacientes revisar el progreso, ajustar tratamientos y personalizar la experiencia de terapia basada en necesidades individuales.

A través de estos objetivos, el proyecto busca no solo mitigar los efectos de la escasez de servicios de salud mental, sino también proporcionar una herramienta poderosa y transformadora que mejore el bienestar y la calidad de vida de los usuarios, haciendo uso de las capacidades que la tecnología digital ofrece.

# Análisis Crítico

Dado el diseño y las configuraciones del proyecto Django para la aplicación de terapia diseñada, incluyendo la estructura de URLs, la configuración del settings.py, y los fragmentos de código de las vistas, es posible indicar la justificación de objetivos clave y justificar las decisiones de diseño que se han tomado en el desarrollo de esta aplicación. Aquí se abordan estos puntos:

## 1. Seguridad y Privacidad

Uso de login\_required: Dada la naturaleza sensible de una aplicación de terapia, es fundamental que el acceso a las vistas esté restringido a usuarios autenticados. Esto ayuda a proteger la privacidad y seguridad de los datos de los usuarios.

SECRET\_KEY y DEBUG: La generación de una clave secreta y la indicación para mantenerla protegida en producción es esencial para la seguridad de la aplicación, evitando problemas de falsificación de sesiones y otros ataques.

## 2. Usabilidad

URLs claras: Las URLs están diseñadas para ser intuitivas y fáciles de entender, utilizando patrones y nombres descriptivos que facilitan a los usuarios y desarrolladores identificar los recursos y acciones correspondientes.

Estructura de templates y archivos estáticos: Organizar los archivos estáticos y los templates de manera clara y lógica ayuda a mantener el proyecto ordenado y mejora la eficiencia del desarrollo.

## 3. Escalabilidad y Mantenimiento

Estructura de la aplicación en módulos: La división del proyecto en aplicaciones Django separadas (apps.therapy) permite un mejor manejo del crecimiento del proyecto y facilita el mantenimiento al modularizar los componentes.

4. INSTALLED\_APPS y middlewares: La configuración cuidadosa de las aplicaciones instaladas y los middleware habilitados asegura que la aplicación pueda escalar de manera efectiva, proporcionando funciones como autenticación, manejo de sesiones, y más.

## 5. Manejo de Contenido Dinámico

Uso de formularios y validación en las vistas: Esto permite que la aplicación maneje entrada de datos de manera segura y eficiente, asegurando la integridad de los datos y proporcionando una interfaz de usuario robusta.

Manejo de archivos multimedia: Configuraciones para MEDIA\_URL y MEDIA\_ROOT permiten a los usuarios subir y gestionar archivos, lo cual es crucial en un entorno terapéutico donde los recursos como documentos, imágenes o videos pueden ser parte del proceso terapéutico.

## 6. Experiencia del Usuario

Redirecciones inteligentes: Después de acciones como iniciar sesión o editar contenido, re

direccionar a los usuarios a páginas relevantes mejora la fluidez y coherencia de la experiencia del usuario.

Todas estas decisiones reflejan un compromiso con la creación de una aplicación segura, usable y personalizable que pueda escalar y adaptarse a las necesidades de los usuarios y desarrolladores a lo largo del tiempo. Las decisiones tomadas apuntan a garantizar que la aplicación no solo cumpla con los requisitos funcionales sino que también ofrezca una experiencia de usuario coherente y agradable, al mismo tiempo que mantiene altos estándares de seguridad y privacidad.

# Implementación y Desarrollo

Como parte del proceso de implementación en el contexto del ramo, se nos entrega un CRUD basado en Django el que contiene una estructura MVC completa. Y como alcance esperado para esta semana, el objetivo es poder presentar un CRUD adaptado a la realidad de nuestro proyecto. Con lo anterior hemos decidido además incorporar algunos elementos adicionales al proyecto a fin de complementar con aspectos formales como son:

- Incorporación de GIT al proyecto: Dejando el repositorio disponible en GITHUB.
- Habilitación de PORT FORWARD con soporte en GITHUB: Permitiendo acceder al interfaz de manera externa vía URL.

## PASO 1: EJECUTAR PROYECTO

Hemos tomado el CRUD entregado y lo hemos ejecutado, siguiendo los elementos definidos en el README.

The screenshot shows a Visual Studio Code interface. On the left, the Explorer sidebar displays a file tree for a project named 'PROYECTO\_S...'. It includes folders like 'apps' (containing 'therapy'), 'migrations', 'templates', and 'crud'. The main editor area shows the 'README.md' file for 'simple-crud-django'. The document contains a list of five steps to run the project: 1. Create virtual environment, 2. Install dependencies, 3. Prepare migrations, 4. Apply migrations, and 5. Run the server. The Terminal panel at the bottom shows the output of running 'python manage.py migrate' and starting the Django development server at http://127.0.0.1:8000/.

*Ilustración 1: La imagen anterior, muestra el proyecto corriendo localmente con las dependencias debidamente instaladas.*

Se ha habilitado el puerto 8000 y se ha utilizado el recurso de Forwarded Address para disponibilidad del proyecto a internet, utilizando devtunnels de Microsoft.

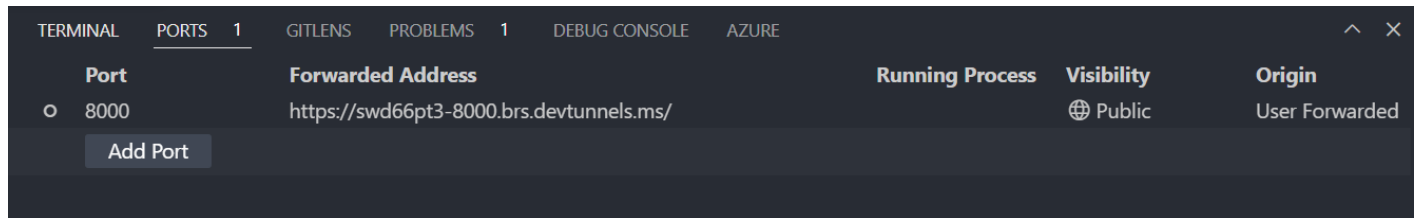


Ilustración 2: Configuración en VSC

El proyecto queda disponible y visible en <https://swd66pt3-8000.brs.devtunnels.ms/>.

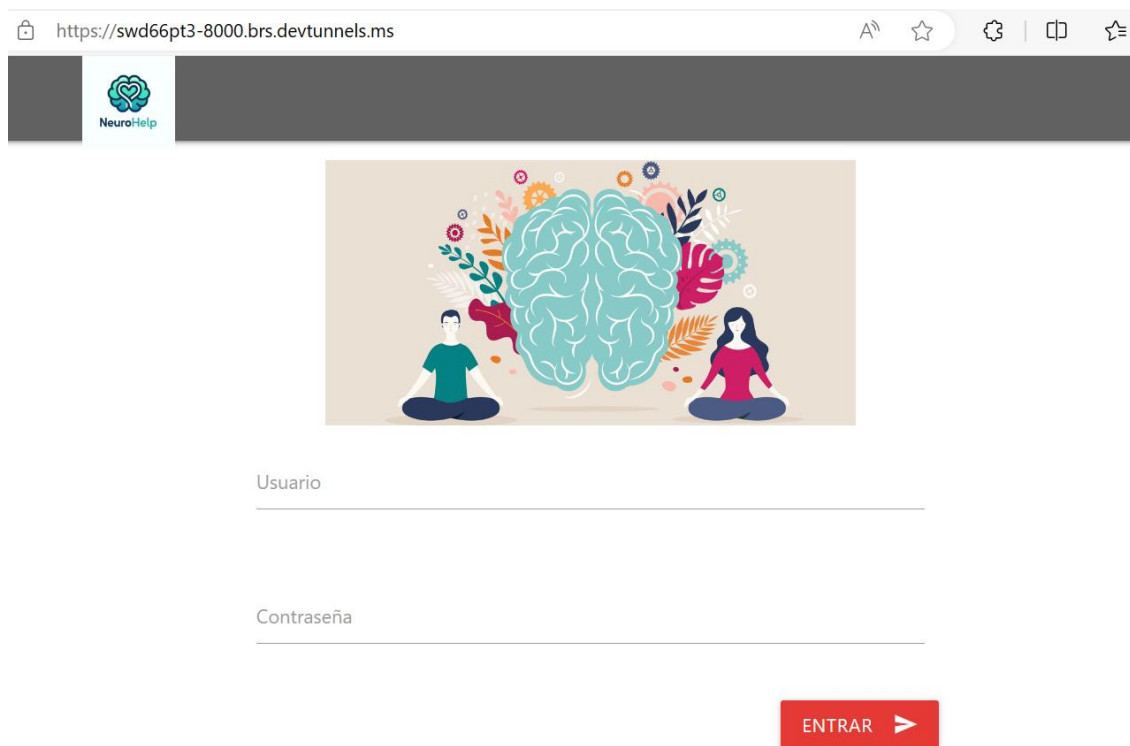


Ilustración 3: Home desplegado en URL abierta, con modificaciones estéticas.

## PASO 2: CONFIGURACIONES

Creamos un superuser a fin de poder acceder remotamente:

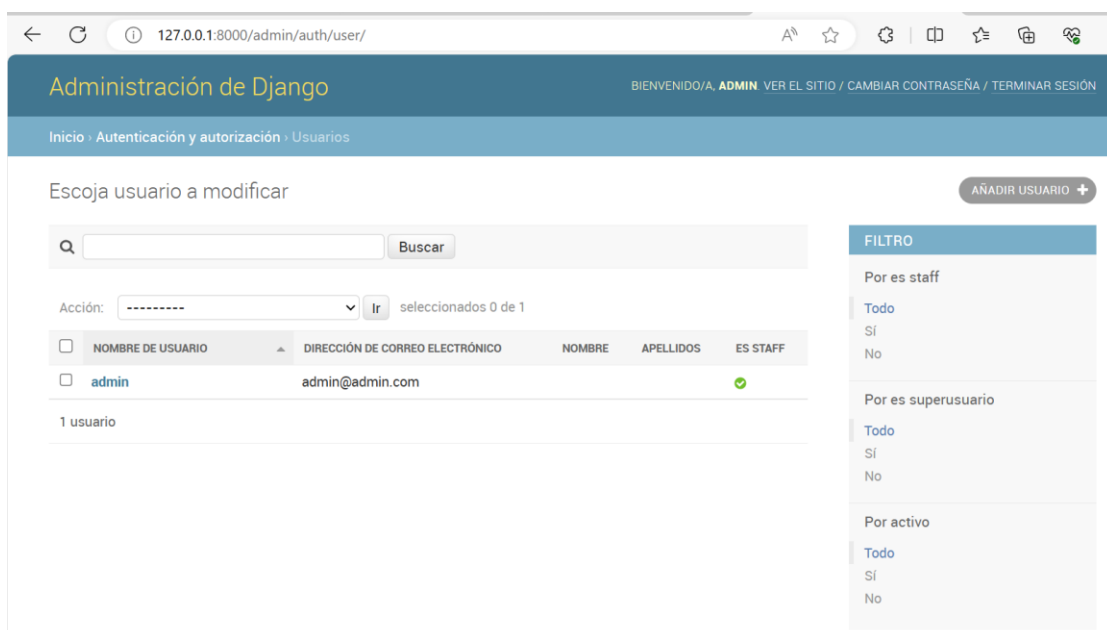
```

OYECTO_SALUD_MENTAL> python manage.py createsuperuser
Username (leave blank to use 'giova'): admin
Email address: admin@admin.com
Password:
Password (again):
The password is too similar to the email address.
This password is too short. It must contain at least 8 characters.
This password is too common.
Password:
Password (again):
The password is too similar to the email address.
Password:
Password (again):
Superuser created successfully.
PS C:\Users\giova\OneDrive\Escritorio\Personal\Estudios\2 - OYECTO_SALUD_MENTAL>

```

email: [admin@admin.com](mailto:admin@admin.com)

Con lo anterior ya tenemos acceso al panel de administración de a DJANGO:





Con lo anterior, creamos un paciente de prueba, a fin de desplegar nuestro portal con el CRUD definido:

The screenshot shows the Django Admin interface for modifying a user. The browser address bar indicates the URL is 127.0.0.1:8000/admin/auth/user/2/change/. The page title is 'Administración de Django'. The user is logged in as 'ADMIN'. The breadcrumb trail is 'Inicio > Autenticación y autorización > Usuarios > paciente'. A green message bar states: 'The usuario "paciente" was added successfully. You may edit it again below.' The main heading is 'Modificar usuario' with a 'HISTÓRICO' button. The form includes fields for 'Nombre de usuario' (set to 'paciente'), 'Contraseña' (showing algorithm and iterations), and 'Información personal' (Name: 'Paciente', Surnames: 'Prueba', Email: 'paciente@prueba.com'). There is also a 'Permisos' section at the bottom.

Ilustración 4: Vista Django para crear usuarios.


A fin de poder realizar las pruebas con el porta, creamos unas categorías de sesión online.

The screenshot shows the Django Admin interface for managing therapy categories. The page title is 'Administración de Django'. The user is logged in as 'ADMIN'. The breadcrumb trail is 'Inicio > Therapy > Categorías de Terapias'. A green message bar states: 'The Categoría de Terapia "Sesión de seguimiento" was added successfully.' The main heading is 'Escoja Categoría de Terapia a modificar' with an 'AÑADIR CATEGORÍA DE TERAPIA +' button. Below the heading is a form with an 'Acción:' dropdown menu and a 'seleccionados 0 de 3' indicator. The list of categories includes: 'CATEGORÍA DE TERAPIA', 'Sesión de seguimiento', 'Urgencia', and 'Evaluación Diagnóstica'. At the bottom, it says '3 Categorías de Terapias'.

## PASO 3: PRUEBAS CRUD

Con las configuraciones anteriores, el paciente puede solicitar una sesión online, basado en las diferentes categorías, cumpliéndose entonces la parte del CREATE del CRUD.

) 127.0.0.1:8000/crear/ A ☆ ⚙ 📄 ☆ 🔍

NeuroHelp

Sesion-Online Recursos Comunidad Seguimiento Salir Bienvenido 'paciente'

Descripcion

Necesito una evaluación

Categoria

-----

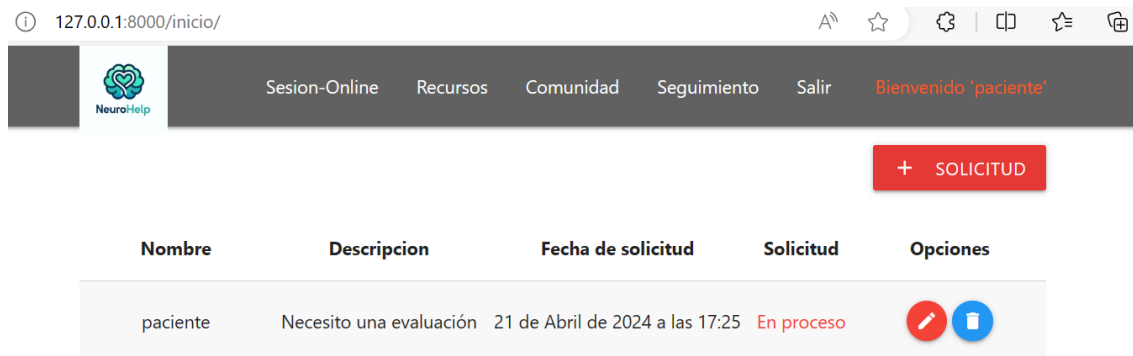
Evaluación Diagnóstica

Urgencia



Sesión de seguimiento

Ilustración 5: Formulario para solicitar sesión Online.

Luego de ingresada la solicitud, entonces tenemos un listado de solicitudes presentadas en una tabla, es decir se cumple aquí el READ del CRUD.



The screenshot shows a web application interface for 'NeuroHelp'. At the top, there is a navigation bar with links: 'Sesion-Online', 'Recursos', 'Comunidad', 'Seguimiento', 'Salir', and a welcome message 'Bienvenido 'paciente''. Below the navigation bar, there is a red button labeled '+ SOLICITUD'. The main content area displays a table with the following columns: 'Nombre', 'Descripcion', 'Fecha de solicitud', 'Solicitud', and 'Opciones'.

Nombre	Descripcion	Fecha de solicitud	Solicitud	Opciones
paciente	Necesito una evaluación	21 de Abril de 2024 a las 17:25	En proceso	 

*Ilustración 6: Lista de sesiones online solicitadas por paciente.*

Para efectos de las pruebas, vamos a “EDITAR” la solicitud, indicando otra descripción, cumpliéndose el EDIT (UPDATE) del CRUD.



The screenshot shows the 'editar' (edit) form in the NeuroHelp application. The URL in the browser is '127.0.0.1:8000/1/editar/'. The navigation bar is the same as in the previous screenshot. The form contains two main fields: 'Descripcion' and 'Categoria'. The 'Descripcion' field has the text 'Requiero evaluación urgente!!!'. The 'Categoria' field is a dropdown menu with the selected option 'Evaluación Diagnóstica'. At the bottom right of the form, there is a red button labeled 'ENTRAR' with a right-pointing arrow.

*Ilustración 7: Formulario para editar solicitud de sesión online.*

Para fines de prueba, creamos una sesión adicional a fin de poder probar el DELETE, del CRUD.

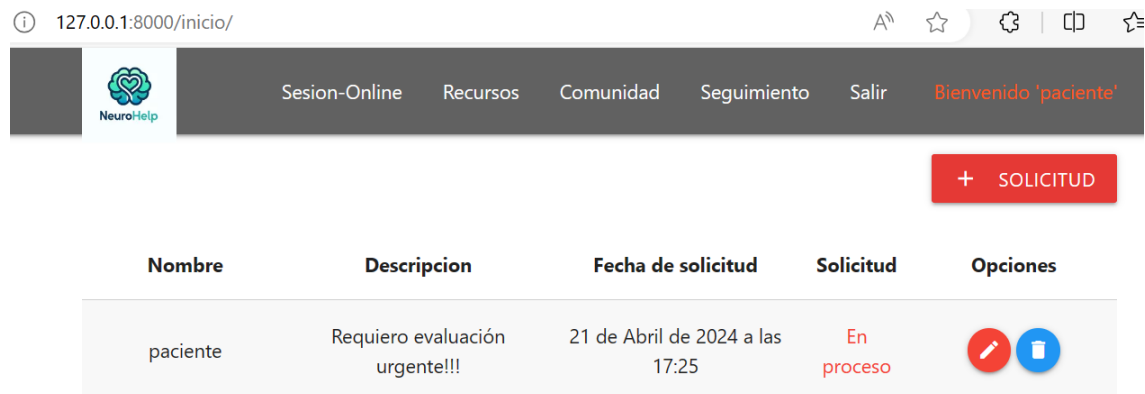


The screenshot shows a web browser at 127.0.0.1:8000/inicio/. The header includes the NeuroHelp logo and navigation links: Sesion-Online, Recursos, Comunidad, Seguimiento, Salir, and Bienvenido 'paciente'. A red button labeled '+ SOLICITUD' is visible. Below is a table with the following data:



Nombre	Descripcion	Fecha de solicitud	Solicitud	Opciones
paciente	Requiero evaluación urgente!!!	21 de Abril de 2024 a las 17:25	En proceso	 
paciente	Solicito segunda sesión para seguimiento	21 de Abril de 2024 a las 17:30	En proceso	 

*Ilustración 8: Listado de sesiones previos al DELETE*

Al presionar el botón rojo de “eliminar” de la tabla, se elimina la solicitud.



The screenshot shows the same web application after deleting one session. The table now contains only one row:

Nombre	Descripcion	Fecha de solicitud	Solicitud	Opciones
paciente	Requiero evaluación urgente!!!	21 de Abril de 2024 a las 17:25	En proceso	 

*Ilustración 9: Tabla posterior al DELETE*

## PASO 4: OTRAS VISTAS

Todo lo anterior aplica de la misma manera para las otras secciones desarrolladas en el portal como son:

### 1. Recursos

127.0.0.1:8000/recursos/crear/

NeuroHelp

Sesión-Online Recursos Comunidad Seguimiento Salir Bienvenido 'paciente'

Título

Texto

Icono

Elegir archivo No se ha seleccionado ningún archivo

Categoría

ENTRAR ➤

*Ilustración 10: Vista para CREAR recursos para el paciente.*

### 2. Comunidad:

127.0.0.1:8000/comunidad/crear/

NeuroHelp

Sesión-Online Recursos Comunidad Seguimiento Salir Bienvenido 'paciente'

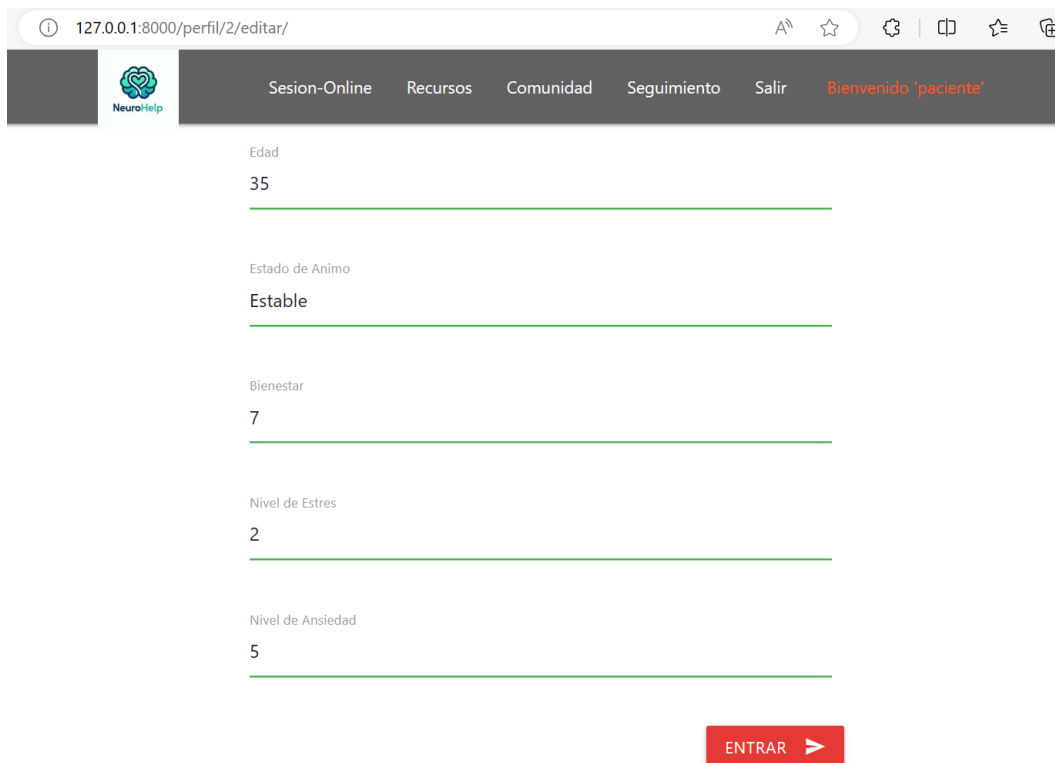
Título

Mensaje Inicial

ENTRAR ➤

*Ilustración 11: Formulario para generar tópico en la comunidad.*

### 3. Seguimiento:



127.0.0.1:8000/perfil/2/editar/

NeuroHelp

Sesion-Online Recursos Comunidad Seguimiento Salir Bienvenido 'paciente'

Edad  
35

Estado de Animo  
Estable

Bienestar  
7

Nivel de Estres  
2

Nivel de Ansiedad  
5

ENTRAR ➤

*Ilustración 12: Formulario para editar valores de seguimiento a paciente.*

## PASO 5: REVISION DE CÓDIGO

Como el estado actual del proyecto se sustenta en el proyecto base CRUD entregado en el ramo, las principales modificaciones se encuentran en la implementación personalizada de las vistas y los respectivos modelos necesarios para estas.

Un ejemplo concreto se encuentra en el archivo `models.py`, en donde principalmente se incorporaron los diferentes atributos que requerían los modelos que se ajustan a nuestros formularios en las diferentes vistas:

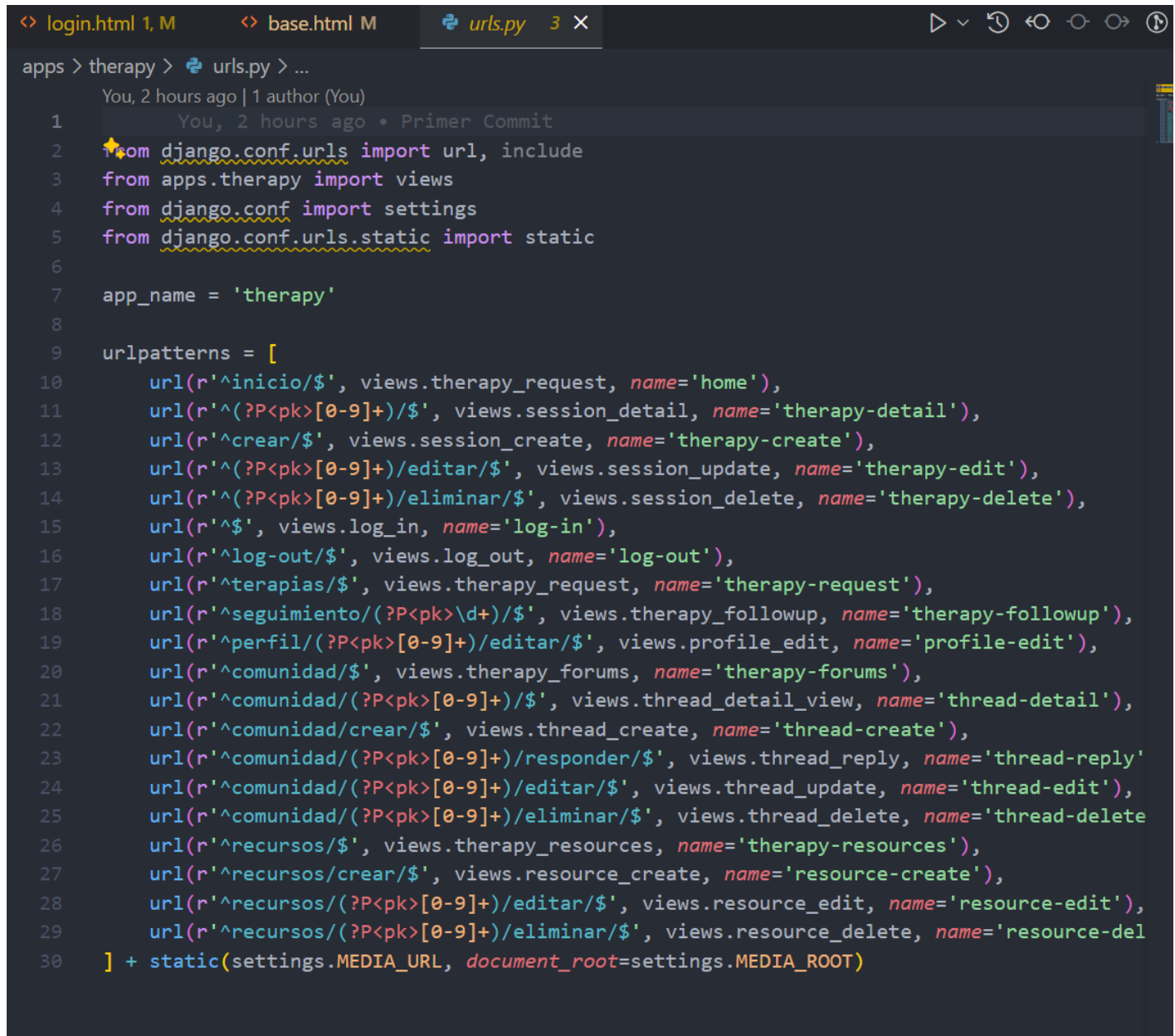
```

You, 2 hours ago | 1 author (You)
26 > class Profiles(BaseName):
27     username = models.CharField(max_length=150, verbose_name='Nombre')
28     age = models.IntegerField(verbose_name='Edad')
29     therapycount = models.IntegerField(verbose_name='Numero de Terapias')
30     wellbeing = models.IntegerField(verbose_name='Bienestar')
31     currentmood = models.CharField(max_length=150, verbose_name='Estado de Animo')
32     stresslevel = models.IntegerField(verbose_name='Nivel de Estres')
33     anxietylevel = models.IntegerField(verbose_name='Nivel de Ansiedad')
34     user = models.ForeignKey(settings.AUTH_USER_MODEL, on_delete=models.CASCADE, verbose_name='Usuario')
35
You, 2 hours ago | 1 author (You)
26 > class Meta:
27     verbose_name = 'Perfil de Usuario'
28     verbose_name_plural = 'Perfiles de Usuarios'
29
30 > def get_profile_url(self):
31     return reverse_lazy('therapy:therapy-followup', kwargs={'pk': self.pk})
32
33 > def get_edit_url(self):
34     return reverse_lazy('therapy:profile-edit', kwargs={'pk': self.pk})
35
You, 2 hours ago | 1 author (You)
36 > class Categories(BaseName):
37     You, 2 hours ago | 1 author (You)
37 > class Meta:
38     verbose_name = 'Categoria de Terapia'
39     verbose_name_plural = 'Categorias de Terapias'
40
41

```

Ilustración 13: Ajuste a `models.py`

Adicionalmente, es parte de lo que se ha implementado, todas las configuraciones de rutas para las distintas urls del CRUD, las que se evidencian en la imagen siguiente y que pertenecen al archivo urls.py



```

login.html 1, M    base.html M    urls.py 3 X
apps > therapy > urls.py > ...
You, 2 hours ago | 1 author (You)
1      You, 2 hours ago • Primer Commit
2  from django.conf.urls import url, include
3  from apps.therapy import views
4  from django.conf import settings
5  from django.conf.urls.static import static
6
7  app_name = 'therapy'
8
9  urlpatterns = [
10     url(r'^inicio/$', views.therapy_request, name='home'),
11     url(r'^(?P<pk>[0-9]+)/$', views.session_detail, name='therapy-detail'),
12     url(r'^crear/$', views.session_create, name='therapy-create'),
13     url(r'^(?P<pk>[0-9]+)/editar/$', views.session_update, name='therapy-edit'),
14     url(r'^(?P<pk>[0-9]+)/eliminar/$', views.session_delete, name='therapy-delete'),
15     url(r'^$', views.log_in, name='log-in'),
16     url(r'^log-out/$', views.log_out, name='log-out'),
17     url(r'^terapias/$', views.therapy_request, name='therapy-request'),
18     url(r'^seguimiento/(?P<pk>\d+)/$', views.therapy_followup, name='therapy-followup'),
19     url(r'^perfil/(?P<pk>[0-9]+)/editar/$', views.profile_edit, name='profile-edit'),
20     url(r'^comunidad/$', views.therapy_forums, name='therapy-forums'),
21     url(r'^comunidad/(?P<pk>[0-9]+)/$', views.thread_detail_view, name='thread-detail'),
22     url(r'^comunidad/crear/$', views.thread_create, name='thread-create'),
23     url(r'^comunidad/(?P<pk>[0-9]+)/responder/$', views.thread_reply, name='thread-reply'),
24     url(r'^comunidad/(?P<pk>[0-9]+)/editar/$', views.thread_update, name='thread-edit'),
25     url(r'^comunidad/(?P<pk>[0-9]+)/eliminar/$', views.thread_delete, name='thread-delete'),
26     url(r'^recursos/$', views.therapy_resources, name='therapy-resources'),
27     url(r'^recursos/crear/$', views.resource_create, name='resource-create'),
28     url(r'^recursos/(?P<pk>[0-9]+)/editar/$', views.resource_edit, name='resource-edit'),
29     url(r'^recursos/(?P<pk>[0-9]+)/eliminar/$', views.resource_delete, name='resource-del'),
30 ] + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)

```

Ilustración 14: Archivo urls.py



# Conclusión

El desarrollo de esta aplicación de terapia online en Django ha demostrado ser un esfuerzo significativo hacia la mejora del acceso a servicios de salud mental. La implementación de una interfaz segura y amigable permite a los usuarios gestionar sus sesiones de terapia de manera eficiente, participar en comunidades de apoyo, y acceder a recursos educativos, todo desde la comodidad de su hogar. Al proporcionar un espacio digital donde los usuarios pueden interactuar de forma segura y confidencial, la aplicación ha logrado crear un entorno terapéutico accesible para una audiencia amplia.

Sin embargo, a pesar de estos avances, la evaluación crítica de la aplicación revela áreas que necesitan mejora y expansión. Por ejemplo, la adaptabilidad y escalabilidad de la plataforma deben ser continuamente revisadas para asegurar que pueda manejar un número creciente de usuarios y tipos de interacciones más complejas que puedan requerirse en el futuro. Además, aunque se ha hecho un esfuerzo por asegurar la plataforma, siempre existe el desafío constante de mantener y mejorar las medidas de seguridad digital, especialmente en un campo tan sensible como la salud mental.

Mirando hacia el futuro, hay varios pasos que se pueden tomar para mejorar y expandir la aplicación, como mejorar la escalabilidad y rendimiento, a medida que la plataforma crece, será esencial implementar soluciones tecnológicas que puedan manejar una carga de usuarios más alta sin degradar la experiencia del usuario. Integración de tecnologías emergentes, considerar la integración de inteligencia artificial y aprendizaje automático para personalizar las experiencias de los usuarios y proporcionar insights más profundos a los terapeutas sobre el progreso de los pacientes. Expansión de recursos multilingües y culturales, para diversificar los recursos disponibles en diferentes idiomas y adaptados a diversas realidades culturales para hacer la plataforma verdaderamente global. Retroalimentación y adaptación continua, implementando un sistema robusto para recoger y analizar la retroalimentación de los usuarios y los profesionales de la salud mental para guiar las mejoras continuas en la funcionalidad y la interfaz de usuario.

En conclusión, el proyecto ha logrado establecer una base sólida para una plataforma de terapia online que es segura, accesible y útil. Sin embargo, el verdadero éxito de la aplicación se medirá por su capacidad para adaptarse y evolucionar con las necesidades de sus usuarios. La aceptación y el impacto continuo en la comunidad de salud mental servirán como el barómetro definitivo para su éxito. La dedicación continua a la mejora y expansión seguirá siendo crucial para cumplir y superar las expectativas iniciales y ayudar a más personas a gestionar su salud mental de manera efectiva.



