

Informe de Laboratorio 7

Tema: DJANGO II

Nota

Estudiante	Escuela	Asignatura
Sergio Hanco Mullisaca shanccom@unsa.edu.pe	Escuela Profesional de Ingeniería de Sistemas	Programación Web 2 Semestre: II Código:

Laboratorio	Tema	Duración
7	DJANGO II	04 horas

Semestre académico	Fecha de inicio	Fecha de entrega
2024 - A	Del 29 Mayo 2024	Al 05 Junio 2024

1. Tarea

- Informe de laboratorio
- Video en Flip
- Ejercicios Propuestos

2. Equipos, materiales y temas utilizados

- VS
- Git 2.39.2.
- Cuenta en GitHub con el correo institucional.

3. URL de Repositorio Github

- URL del video en yt.
- <https://youtu.be/jTnHxFnj-ZI>
- URL del video en flip.
- <https://flip.com/s/sdBCVHN1UHXB>
- URL del GITHUB.
- https://github.com/shanccom/Programacion_Web_2.git

4. Actividades

4.1. EJERCICIO PROPUESTO

CREACION ENTORNO VIRTUAL CON GITIGNORE

```
PS C:\Sergio\SEMESTRE 2024A\PMEB\Programacion_Web_2\LAB_06_DJANGO\ejercicio_propuesto> ls

Directorio: C:\Sergio\SEMESTRE 2024A\PMEB\Programacion_Web_2\LAB_06_DJANGO\ejercicio_propuesto

Mode                LastWriteTime         Length Name
----                -
-a-----         28/05/2024   22:40             0 .gitignore.txt

PS C:\Sergio\SEMESTRE 2024A\PMEB\Programacion_Web_2\LAB_06_DJANGO\ejercicio_propuesto> python -m venv env
PS C:\Sergio\SEMESTRE 2024A\PMEB\Programacion_Web_2\LAB_06_DJANGO\ejercicio_propuesto> .\env\Scripts\activate
(env) PS C:\Sergio\SEMESTRE 2024A\PMEB\Programacion_Web_2\LAB_06_DJANGO\ejercicio_propuesto> pip install django
Collecting django
  Using cached Django-5.0.6-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref<4,>=3.7.0 (from django)
  Using cached asgiref-3.8.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse>=0.3.1 (from django)
  Using cached sqlparse-0.5.0-py3-none-any.whl.metadata (3.9 kB)
Collecting tzdata (from django)
  Using cached tzdata-2024.1-py2.py3-none-any.whl.metadata (1.4 kB)
Using cached Django-5.0.6-py3-none-any.whl (8.2 MB)
Using cached asgiref-3.8.1-py3-none-any.whl (23 kB)
Using cached sqlparse-0.5.0-py3-none-any.whl (43 kB)
Using cached tzdata-2024.1-py2.py3-none-any.whl (345 kB)
Installing collected packages: tzdata, sqlparse, asgiref, django
Successfully installed asgiref-3.8.1 django-5.0.6 sqlparse-0.5.0 tzdata-2024.1
(env) PS C:\Sergio\SEMESTRE 2024A\PMEB\Programacion_Web_2\LAB_06_DJANGO\ejercicio_propuesto> |
```

ARCHIVOS MODIFICADOS (TELUSKO)

Listing 1: settings.py

```
INSTALLED_APPS = [
    'travello.apps.TravelloConfig',
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
]
```

```
'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'telusko.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [os.path.join(BASE_DIR, 'templates')],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    ],
]

WSGI_APPLICATION = 'telusko.wsgi.application'

# Database
# https://docs.djangoproject.com/en/2.2/ref/settings/#databases

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.postgresql',
        'NAME': 'telusko',
        'USER': 'postgres',
        'PASSWORD': '123',
        'HOST': 'localhost'
    }
}
```

Listing 2: urls.py

```
from django.contrib import admin
from django.urls import path, include
from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('', include('travello.urls')),
    path('admin/', admin.site.urls),
    path('accounts/', include('accounts.urls'))
]

urlpatterns = urlpatterns + static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

Listing 3: wsgi.py

```
import os

from django.core.wsgi import get_wsgi_application

os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'telusko.settings')

application = get_wsgi_application()
```

ARCHIVOS MODIFICADOS (CALC)

Listing 4: apps.py

```
from django.apps import AppConfig

class CalcConfig(AppConfig):
    default_auto_field = 'django.db.models.BigAutoField'
    name = 'calc'
```

Listing 5: urls.py

```
from django.urls import path

from . import views

urlpatterns = [path("", views.home, name="home"), path("add", views.add, name="add")]
```

Listing 6: views.py

```
from django.shortcuts import render
from django.http import HttpResponse

# Create your views here.

def home(request):
    return render(request, "home.html", {"name": "Navin"})

def add(request):

    val1 = int(request.POST["num1"])
    val2 = int(request.POST["num2"])
    res = val1 + val2

    return render(request, "result.html", {"result": res})
```

ARCHIVOS MODIFICADOS (TRAVELLO-APP)

Listing 7: admin.py

```
from django.contrib import admin
from .models import Destination

admin.site.register(Destination)
```

Listing 8: apps.py

```
from django.apps import AppConfig

class TravelloConfig(AppConfig):
    name = 'travello'
```

Listing 9: models.py

```
from django.db import models

class Destination(models.Model):

    name = models.CharField(max_length=100)
    img = models.ImageField(upload_to='pics')
    desc = models.TextField()
    price = models.IntegerField()
    offer = models.BooleanField(default=False)
```

Listing 10: urls.py

```
from django.urls import path

from . import views

urlpatterns = [path("", views.index, name="index")]
```

Listing 11: views.py

```
from django.shortcuts import render
from .models import Destination
# Create your views here.

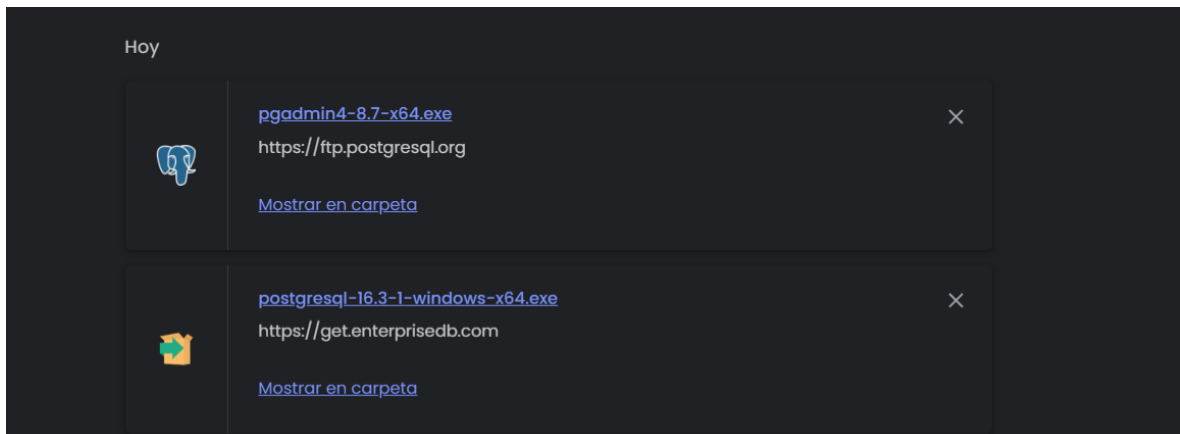
def index(request):

    dests = Destination.objects.all()

    return render(request, "index.html", {'dests': dests})
```

IMAGENES-PRUEBAS

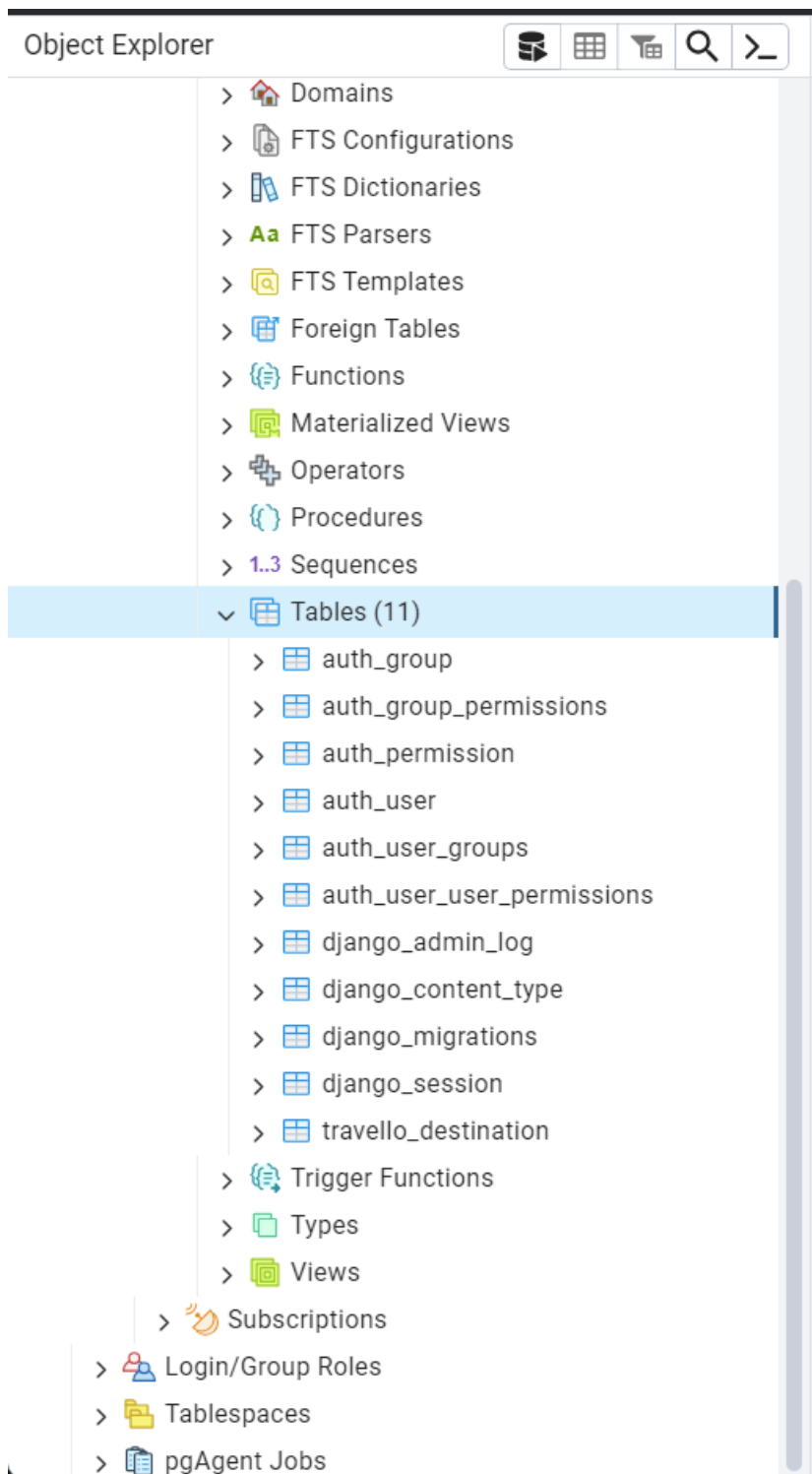
INSTALACION DE POSTRGRESQL - PGADMIN



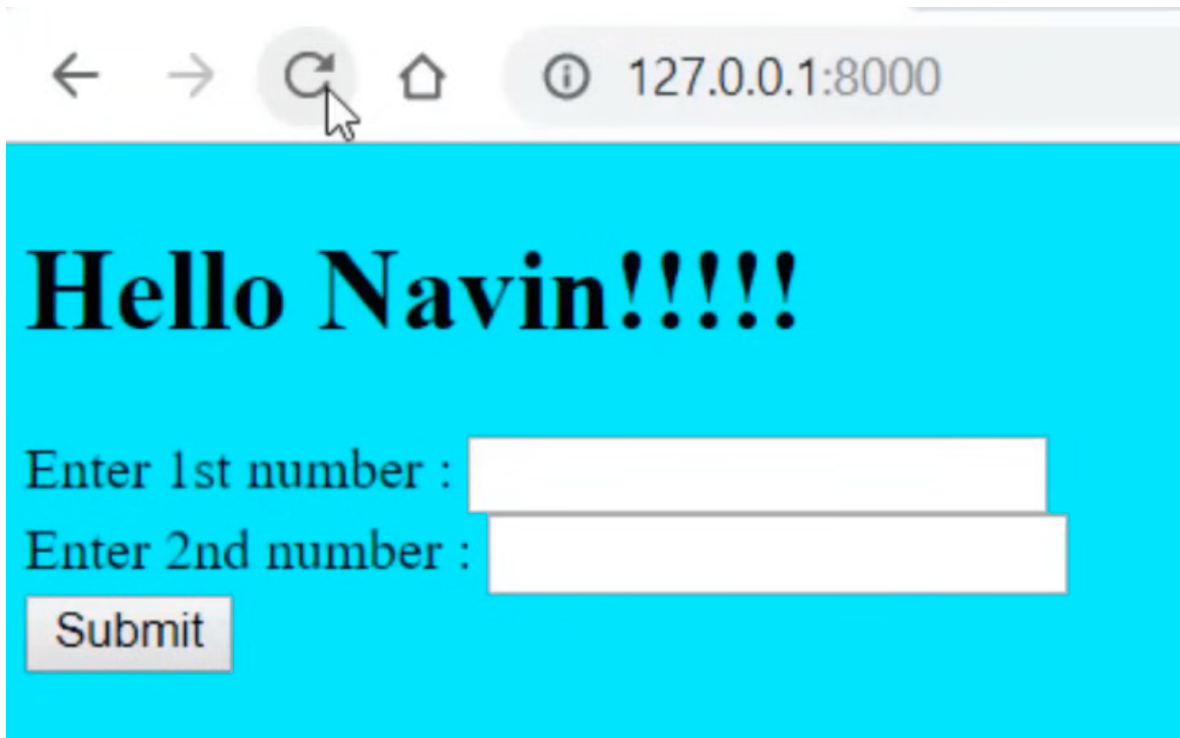
MIGRACIONES

```
(myvenv) PS C:\Users\USER\Desktop\EXPO-PWEB2\django-telusko-codes\telusko_user_account\telusko> python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions, travello
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
  Applying travello.0001_initial... OK
  Applying travello.0002_destination_price... OK
(myvenv) PS C:\Users\USER\Desktop\EXPO-PWEB2\django-telusko-codes\telusko_user_account\telusko> |
```

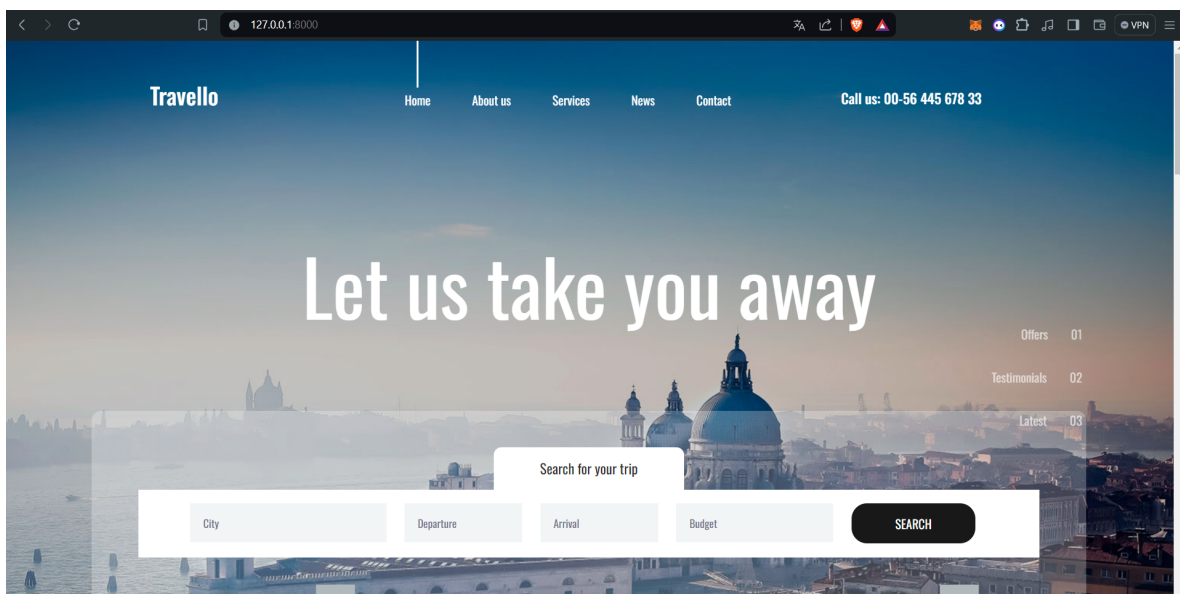
TABLAS EN PGADMIN



PRUEBAS DE FUNCIONAMIENTO (PRUEBA DE SUMA)



PRUEBAS DE FUNCIONAMIENTO (PRUEBA DE LA PAGINA EN SI)



5. Rúbricas

5.1. Entregable Informe

Tabla 1: Tipo de Informe

Informe	
Latex	El informe está en formato PDF desde Latex, con un formato limpio (buena presentación) y fácil de leer.

5.2. Rúbrica para el contenido del Informe y demostración

- El alumno debe marcar o dejar en blanco en celdas de la columna **Checklist** si cumple con el ítem correspondiente.
- Si un alumno supera la fecha de entrega, su calificación será sobre la nota mínima aprobada, siempre y cuando cumpla con todos los ítems.
- El alumno debe autocalificarse en la columna **Estudiante** de acuerdo a la siguiente tabla:

Tabla 2: Niveles de desempeño

Puntos	Nivel			
	Insatisfactorio 25 %	En Proceso 50 %	Satisfactorio 75 %	Sobresaliente 100 %
2.0	0.5	1.0	1.5	2.0
4.0	1.0	2.0	3.0	4.0

Tabla 3: Rúbrica para contenido del Informe y demostración

Contenido y demostración		Puntos	Checklist	Estudiante	Profesor
1. GitHub	Hay enlace URL activo del directorio para el laboratorio hacia su repositorio GitHub con código fuente terminado y fácil de revisar.	2	X	2	
2. Commits	Hay capturas de pantalla de los commits más importantes con sus explicaciones detalladas. (El profesor puede preguntar para refrendar calificación).	4	X	4	
3. Código fuente	Hay porciones de código fuente importantes con numeración y explicaciones detalladas de sus funciones.	2	X	2	
4. Ejecución	Se incluyen ejecuciones/pruebas del código fuente explicadas gradualmente.	2	X	1	
5. Pregunta	Se responde con completitud a la pregunta formulada en la tarea. (El profesor puede preguntar para refrendar calificación).	2	X	2	
6. Fechas	Las fechas de modificación del código fuente están dentro de los plazos de fecha de entrega establecidos.	2	X	2	
7. Ortografía	El documento no muestra errores ortográficos.	2	X	2	
8. Madurez	El Informe muestra de manera general una evolución de la madurez del código fuente, explicaciones puntuales pero precisas y un acabado impecable. (El profesor puede preguntar para refrendar calificación).	4	X	4	
Total		20		19	