



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

EXAMEN COMPLEXIVO PRÁCTICO

Sumario

EXAMEN COMPLEXIVO PRÁCTICO	1
Indicaciones Generales	2
SECCIÓN 1: BASE DE DATOS RELACIONAL (POSTGRESQL)	3
SECCIÓN 2: BASE DE DATOS NO RELACIONAL (MONGODB)	4
SECCIÓN 3: BACKEND – DJANGO REST	5
SECCIÓN 4: FRONTEND – REACTJS	6
SECCIÓN 5: APLICACIÓN MÓVIL – REACT NATIVE	6
SECCIÓN 6: SISTEMAS OPERATIVOS – UBUNTU (MÁQUINA VIRTUAL)	7

**Cohorte 2024-1
Ciclo 2025-2**



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

Indicaciones Generales

Todas las capturas deben mostrar:

- Fecha y hora visible del sistema.
- Terminal o navegador completo (sin recortes parciales).
- Las capturas deben estar numeradas consecutivamente.
- Deben entregarse en un único documento PDF o Word ordenado según esta plantilla.
- No se aceptarán imágenes editadas o fragmentadas.



SECCIÓN 1: BASE DE DATOS RELACIONAL (POSTGRESQL)

Captura 1 – Creación de Base de Datos

Mostrar en terminal el comando de creación de la base de datos y el resultado exitoso.

A screenshot of a Windows PowerShell window titled "Windows PowerShell". The command entered is "psql -U postgres". The output shows a warning about code page differences and a successful database creation command: "CREATE DATABASE lab_db;". Below the PowerShell window, there is a redacted section of the document. At the bottom of the page, there are several numbered capture sections with their descriptions.

Captura 4 – Tablas Generadas por Migración

Mostrar el listado de tablas creadas mediante migraciones (comando \dt).

Captura 5 – Estructura de Tablas

Mostrar la estructura detallada de las dos tablas principales (comando \d nombre_tabla), donde se evidencien tipos de datos.

Captura 6 – Creación de Índice

Mostrar el comando de creación del índice y la verificación de su existencia.

Captura 7 – Creación de Vista

Captura 2 – Creación de Usuario y Asignación de Permisos

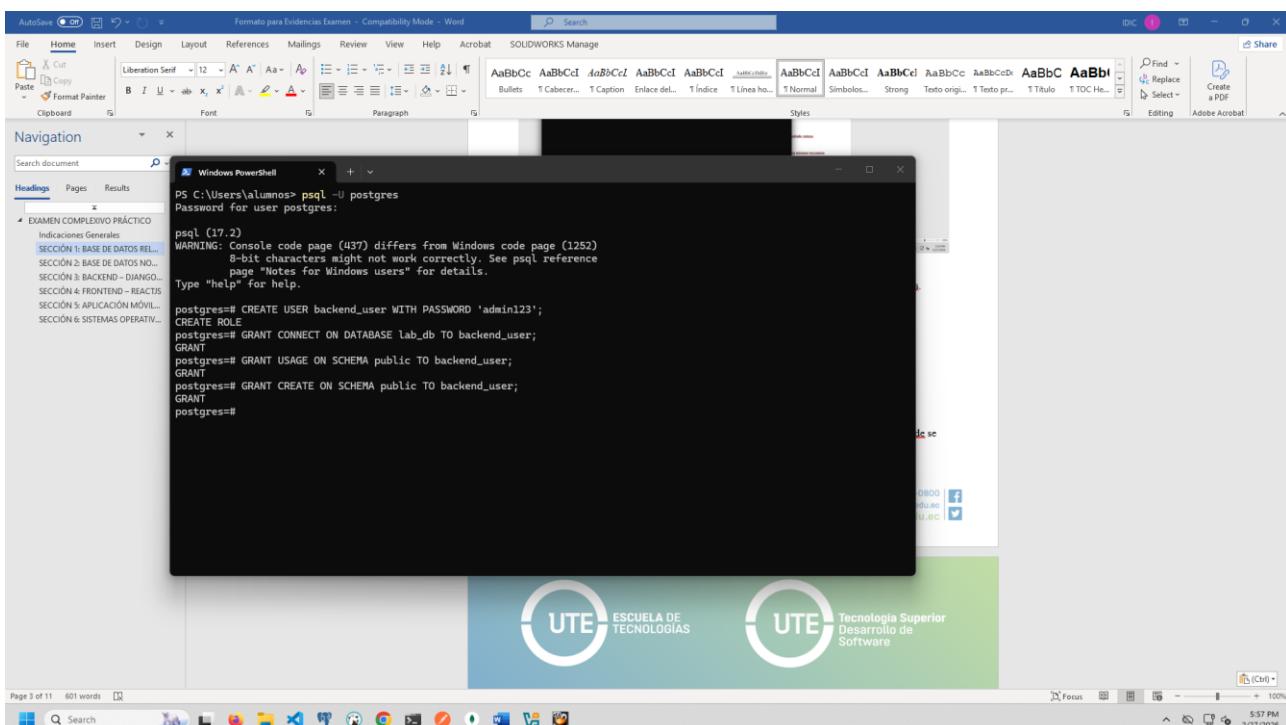
Mostrar los comandos utilizados para crear el usuario y asignar permisos mínimos necesarios.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



Captura 3 – Conexión con el Usuario Creado

Mostrar conexión exitosa mediante psql y listado de bases de datos.



```
PS C:\Users\alumnos> psql -U backend_user -b lab_db
Password for user backend_user:
psql (17.2)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See pgsql reference
page "Notes for Windows users" for details.
Type "help" for help.

lab_db=> \l
                                         List of databases
   Name    | Owner | Encoding | Locale Provider | Collate           | Ctype            | Locale
   postgres |    postgres | UTF8      | libc              | English_United States.1252 | English_United States.1252 |
   |
   | postgres=CTc/postgres +          |               |                   |                   |                   |                   |
   |
   | backend_user=c/postgres          |               |                   |                   |                   |                   |
   |
   | postgres |    postgres | UTF8      | libc              | English_United States.1252 | English_United States.1252 |
   |
   | programacion |    openpg | UTF8      | libc              | C                 | English_United States.1252 |
   |
   | template0 |    postgres | UTF8      | libc              | English_United States.1252 | English_United States.1252 |
   |
   | template1 |    postgres | UTF8      | libc              | English_United States.1252 | English_United States.1252 |
   |
   | postgres=CtC/postgres           |               |                   |                   |                   |                   |
(5 rows)

lab_db=>
```

Captura 4 – Tablas Generadas por Migración

Mostrar el listado de tablas creadas mediante migraciones (comando `\dt`).

Captura 5 – Estructura de Tablas

Mostrar la estructura detallada de las dos tablas principales (comando `\d nombre_tabla`), donde se evidencien tipos de datos.



The screenshot shows a Windows PowerShell window with the following content:

```
lab_db> \d lab_laborder
Table "public.lab_laborder"
 Column | Type | Collation | Nullable | Default
-----+-----+-----+-----+-----+
 id | bigint | not null | generated by default as identity
 patient_name | character varying(120) | not null
 status | integer | not null
 result_summary | character varying(300) | not null
 created_at | timestamp with time zone | not null
 test_id | bigint | not null |
Indexes:
 "lab_laborder_pkey" PRIMARY KEY, btree (id)
 "lab_laborder_test_id_2ba60e7f" btree (test_id)
Foreign-key constraints:
 "lab_laborder_test_id_2ba60e7f_fk_lab_labtest_id" FOREIGN KEY (test_id) REFERENCES lab_labtest(id) DEFERRABLE INITIALLY DEFERRED

lab_db> \d lab_labtest
Table "public.lab_labtest"
 Column | Type | Collation | Nullable | Default
-----+-----+-----+-----+-----+
 id | bigint | not null | generated by default as identity
 test_name | character varying(150) | not null
 sample_type | character varying(60) | not null
 price | numeric(10,2) | not null
 is_available | boolean | not null |
Indexes:
 "lab_labtest_pkey" PRIMARY KEY, btree (id)
Referenced by:
 TABLE "lab_laborder" CONSTRAINT "lab_laborder_test_id_2ba60e7f_fk_lab_labtest_id" FOREIGN KEY (test_id) REFERENCES lab_labtest(id) DEFERRABLE INITIALLY DEFERRED

lab_db>
```

Captura 6 – Creación de Índice

Mostrar el comando de creación del índice y la verificación de su existencia.



The screenshot shows a desktop environment with several open windows:

- A browser window titled "GUÍA RÁPIDA – Consultas PostgreSQL (SQL) y MongoDB (Shell)" containing a MongoDB shell session. The session shows commands for creating a database, collections, users, and roles, as well as creating indexes on collections.
- An Microsoft Word document titled "2026-02-27_Exámenes_Prácticos_Gracia - Compatibility Mode - W..." which contains a list of tasks related to MongoDB and Django REST API.
- A terminal window titled "Backend - Django REST" showing a list of steps to set up a Django project named "lab_backend".

Captura 7 – Creación de Vista

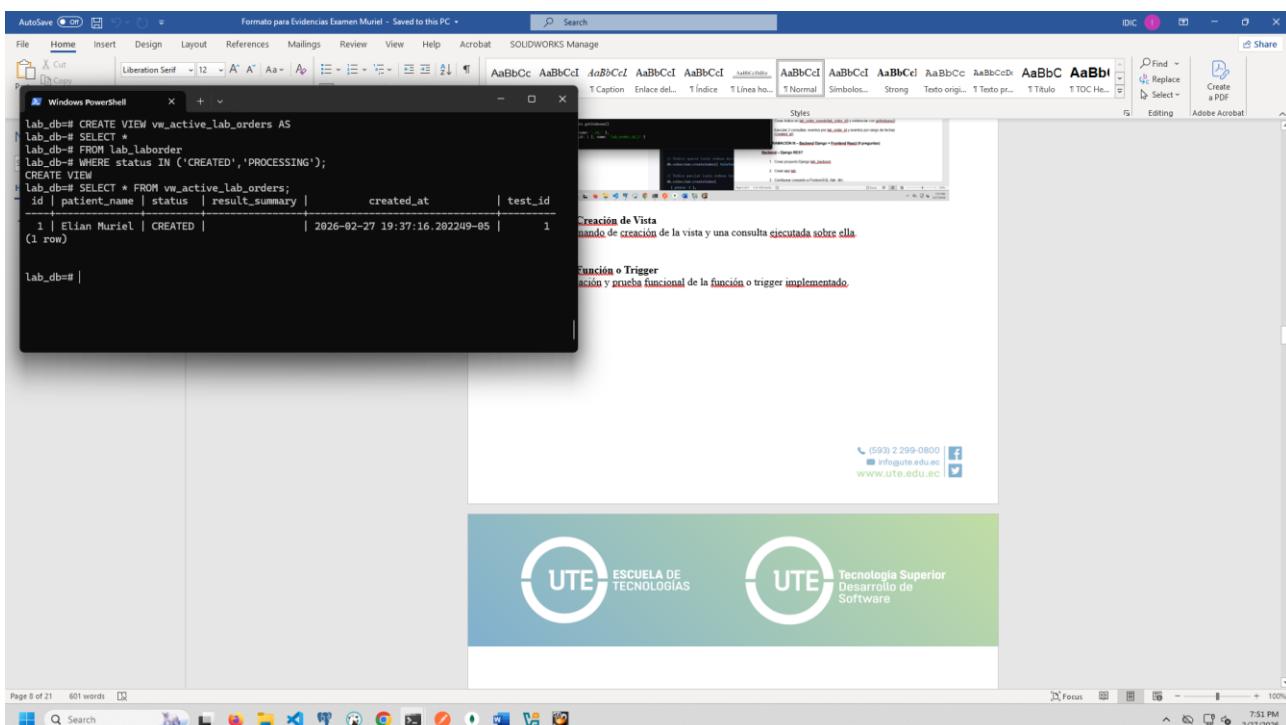
Mostrar el comando de creación de la vista y una consulta ejecutada sobre ella.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



Captura 8 – Función o Trigger

Mostrar la creación y prueba funcional de la función o trigger implementado.



**ESCUELA DE
TECNOLOGÍAS**



**Tecnología Superior
Desarrollo de
Software**



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



SECCIÓN 2: BASE DE DATOS NO RELACIONAL (MONGODB)

Captura 9 – Creación y Selección de Base de Datos

Mostrar el uso de la base de datos desde mongosh.

The screenshot shows a Microsoft Word document with a terminal window embedded. The terminal window displays the following MongoDB shell session:

```
PS C:\Users\alumnos> mongosh
Current Mongosh Log ID: 69a2221999631617957c2996
Connecting to: mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+0.7.0
Using MongoDB: 8.0.4
Using Mongosh: 2.7.0

For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/

The server generated these startup warnings when booting
2026-02-27T16:53:42.437-05:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted

test> show databases
admin 48.00 KiB
config 48.00 KiB
local 88.00 KiB
test> use lab_logs
switched to db lab_logs

mongosh>
```

Below the terminal window, there is a section titled "PROGRAMACIÓN III – Backend Django + Frontend React (9 preguntas)" with a numbered list of tasks:

1. Crear proyecto Django `lab_backend`.
2. Crear app `lab`.
3. Configurar conexión a PostgreSQL (`lab_db`).
4. Configurar conexión a MongoDB (`lab_logs`).
5. Crear modelos Django para `lab_tests` y `lab_orders` según campos definidos.
6. Crear endpoint GET `/lab-tests` que liste pruebas (SQL).

Captura 10 – Creación de Usuario

Mostrar el comando de creación del usuario con roles asignados.



AutoSave 0:00 File Home Insert Design Layout References Mailings Review View Help Acrobat SOLIDWORKS Manage

Cut Copy Format Painter

Clipboard Font Paragraph Styles Editing Share Adobe Acrobat

base de datos II – MongoDB (7 preguntas)

```
mongosh mongoDB://127.0.0.1:27017
Using MongoDB: 8.0.4
Using Mongosh: 2.7.0
For mongosh info see: https://www.mongodb.com/docs/mongodb-shell/
-----
The server generated these startup warnings when booting
2026-02-27T16:53:42.437-05:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
-----
test> show databases
admin 40.00 KiB
config 48.00 KiB
local 88.00 KiB
test> use lab_logs
switched to db lab_logs
lab_logs> db.create
db.createUser() db.createCollection() db.createEncryptedCollection() db.createView()
db.createRole()

lab_logs> db.createUser({
| user: "mongo_backend_user",
| pwd: "exa_2026_ute",
| roles: [
|   { role: "readWrite", db: "lab_logs" }
| ]
| })
{ ok: 1 }
lab_logs>
```

4. Configurar conexión a MongoDB (lab_logs)

5. Crear modelos Django para lab_tests y lab_orders según campos definidos.

6. Crear endpoint GET /lab-tests que liste pruebas (SQL).

7. Crear endpoints GET /lab-orders y POST /lab-orders (SQL), y en POST registrar evento en lab_order_events (Mongo) con lab_order_id

Page 6 of 8 1439 words

Focus 6:05 PM 2/27/2026

Captura 11 – Creación o Verificación de Colecciones

Mostrar las colecciones existentes y la inserción de un documento de prueba.



Objetivo: Diseñar e implementar una base de datos NoSQL para eventos y registros del sistema

```
mongosh mongodb://127.0.0.1:27017
2026-02-27T16:53:42.437-05:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
test> show databases
admin 0.00 KiB
config 0.00 KiB
local 0.00 KiB
test> use lab_logs
switched to db lab_logs
lab_logs> db.create
db.createUser() db.createCollection() db.createEncryptedCollection() db.createView()
db.createRole()
lab_logs> db.createUser({
| user: "mongo_backend_user",
| pwd: "exa_2026_ute",
| roles: [
| | role: "readWrite", db: "lab_logs"
| ]
| })
{ ok: 1 }
lab_logs> db.createCollection("test_catalog")
{ ok: 1 }
lab_logs> db.createCollection("lab_order_events")
{ ok: 1 }
lab_logs> show collections
lab_order_events
test_catalog
lab_logs>
```

6. Crear endpoint GET /lab_tests que liste pruebas (SQL).

7. Crear endpoints GET /lab-orders y POST /lab-orders (SQL), y en POST registrar evento en lab_order_events (Mongo) con lab_order_id.

Frontend – ReactJS (consume TABLAS SQL)

Captura 12 – Creación de Índice

Mostrar el comando createIndex() y la verificación con getIndexes().



The screenshot shows a Windows desktop environment. In the center, there is a Microsoft Word document titled "DE DATOS II – MongoDB (7 preguntas)". The document contains several sections of text and code snippets related to MongoDB setup and indexing. On the left side of the screen, a browser window is open, showing a MongoDB shell session in progress. The session starts with switching to the 'lab_logs' database and creating various collections like 'lab_logs', 'test_catalog', and 'lab_order_events'. It then creates users ('mongo_backend_user' with role 'readWrite') and roles ('exa_2026_ute'). Finally, it creates indexes on the 'lab_order_id' field in both collections. The bottom right corner of the Word document shows a footer with page numbers and a timestamp.

```
mongosh mongodb://127.0.0.1:27017
switched to db lab_logs
lab_logs> db.create
db.createUser("mongo_backend_user", {pwd: "exa_2026_ute"}, {roles: [{role: "readWrite", db: "lab_logs"}]}){ ok: 1 }
lab_logs> db.createCollection("test_catalog")
{ ok: 1 }
lab_logs> db.createCollection("lab_order_events")
{ ok: 1 }
lab_logs> show collections
lab_order_events
test_catalog
lab_logs> db.lab_order_events.getIndexes()
[ { v: 2, key: { _id: 1 }, name: '_id_1' } ]
lab_logs> db.lab_order_events.createIndex({ lab_order_id: 1 })
lab_order_id_1
lab_logs> db.lab_order_events.getIndexes()
[ { v: 2, key: { _id: 1 }, name: '_id_1' },
{ v: 2, key: { lab_order_id: 1 }, name: 'lab_order_id_1' } ]
lab_logs>
```

Backend

DE DATOS II – MongoDB (7 preguntas)

o: Diseñar e implementar una base de datos NoSQL para eventos y registros del sistema.

Crear/definir la base de datos `lab_logs` (evidenciar uso `lab_logs` en `mongosh`).

Crear el usuario `mongo_backend_user` con contraseña `exa_2026_ute`.

Asignar roles mínimos para leer/escribir en `lab_logs` (sin permisos administrativos globales).

Probar autenticación conectándose con `mongo_backend_user` y verificar que puede operar sobre `lab_logs`.

Verificar/rear colecciones `test_catalog` y `lab_order_events` con los campos definidos.

Crear índice en `lab_order_events`(`lab_order_id`) y evidenciar con `getIndexes()`.

Ejecutar 2 consultas: eventos por `lab_order_id` y eventos por rango de fechas (`created_at`)

RAMACIÓN III – Backend Django + Frontend React (9 preguntas)

Backend – Django REST

1. Crear proyecto Django `lab_backend`
2. Crear app `lab`
3. Configurar conexión a PostgreSQL (lab_db)

Page 6 of 8 | 1 of 1439 words

Captura 13 – Consulta por Identificador

Mostrar una consulta filtrando por el identificador relacionado con la tabla relacional.



The screenshot shows a Windows desktop environment. In the center, there's a Microsoft Word document titled "2026-02-27_Exámenes_Prácticas_Gracia - Compatibility Mode - W...". The document contains several sections of text, some of which are highlighted in red. On the left, a browser window is open to a GitHub repository named "ElianMuriel/base". Below the browser, a terminal window titled "mongosh mongodb://127.0.0.1" is running MongoDB commands. At the bottom, the Windows taskbar is visible with icons for various applications like File Explorer, Task View, and the Start button.

```
mongosh mongodb://127.0.0.1
lab_logs> db.createCollection("test_catalog")
{
  "ok": 1
}
lab_logs> db.createCollection("lab_order_events")
{
  "ok": 1
}
lab_logs> show collections
lab_order_events
test_catalog
test_catalog
lab_logs> db.lab_order_events.getIndexes()
[ { v: 2, key: { _id: 1 }, name: '_id_1' } ]
lab_logs> db.lab_order_events.createIndex({ lab_order_id: 1 })
lab_order_id_1
lab_logs> db.lab_order_events.getIndexes()
[
  { v: 2, key: { _id: 1 }, name: '_id_1' },
  { v: 2, key: { lab_order_id: 1 }, name: 'lab_order_id_1' }
]
lab_logs> db.lab_order_events.find({ lab_order_id: 1 })
lab_logs> db.lab_order_events.find({ lab_order_id: 1 })
[
  {
    "_id": ObjectId('69a238bc66d8da4c68e1753f'),
    "lab_order_id": 1,
    "event_type": "CREATED",
    "source": "WEB",
    "note": "certificado de tipo sanguíneo",
    "created_at": ISODate("2026-02-28T00:37:16.208Z")
  }
]
lab_logs>
```

DE DATOS II – MongoDB (7 preguntas)

1. Diseñar e implementar una base de datos NoSQL para eventos y registros de la actividad.

Crear/definir la base de datos `lab_logs` (evidenciar uso `lab_logs` en mongosh).

Crear el usuario `mongo_backend_user` con contraseña `exa_2026_ute`.

Asignar roles mínimos para leer/escibir en `lab_logs` (sin permisos administrativos globales).

Probar autenticación conectándose con `mongo_backend_user` y verificar que puede operar sobre `lab_logs`.

Verificar/crear colecciones `test_catalog` y `lab_order_events` con los campos definidos.

Crear índice en `lab_order_events(lab_order_id)` y evidenciar con `getIndexes()`.

Ejecutar 2 consultas: eventos por `lab_order_id` y eventos por rango de fechas (`created_at`).

RAMACIÓN III – Backend Django + Frontend React (9 preguntas)

1. Crear proyecto Django `lab_backend`.

2. Crear app `lab`.

3. Configurar conexión a PostgreSQL (`lab_db`).

4. Configurar conexión a MongoDB (`lab_logs`).

Captura 14 – Consulta por Rango de Fechas

Mostrar una consulta utilizando el campo de fecha.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

The screenshot shows a Microsoft Word document titled "Formato para Evidencias Examen - Compatibility Mode - Word". In the center of the page is a screenshot of a MongoDB terminal window titled "mongosh mongodb://127.0.0.1:27017". The terminal displays the following command and its results:

```
lab_logs> db.lab_order_events.find({ lab_order_id: 1 })
[{"_id": ObjectId('69a238bc66d0da4c60e1753f'), "lab_order_id": 1, "event_type": "CREATED", "source": "WEB", "note": "Certificado de tipo sanguíneo", "created_at": ISODate("2026-02-28T00:37:16.208Z")}]
```

Below the terminal window, there are two footer banners: "UTE ESCUELA DE TECNOLOGÍAS" and "UTE Tecnología Superior Desarrollo de Software". The Windows taskbar at the bottom shows various application icons.



SECCIÓN 3: BACKEND – DJANGO REST

Captura 15 – Creación del Proyecto y Aplicación

Mostrar estructura del proyecto y aplicación creada.

The screenshot shows a Windows desktop environment. In the center, there is a terminal window titled "Windows PowerShell" with the command "dir" run in the directory "Y:\prueba_complexivo\laboratorio\lab_backend". The output lists several files and directories:

```
PS Y:\prueba_complexivo\laboratorio\lab_backend
Directory: Y:\prueba_complexivo\laboratorio\lab_backend

Mode                LastWriteTime         Length Name
-->                -->-->-->-->
d----        2/27/2026  6:42 PM          config
d----        2/27/2026  6:42 PM          lab
d----        2/27/2026  6:45 PM          venv
-a--        2/27/2026  6:55 PM          .env
-a--        2/27/2026  6:37 PM          584 manage.py
-a--        2/27/2026  6:37 PM          182 requirements.txt
```

Below the terminal window is a code editor interface. On the left, the "EXPLORER" sidebar shows the project structure under "LABORATORIO". The "lab_backend" folder contains subfolders like "config", "lab", and "venv", and various Python files such as "admin.py", "apps.py", "auth_views.py", "exceptions.py", "models.py", "mongo.py", "permissions.py", "serializers.py", "tests.py", "urls.py", and "views.py". It also includes ".env", "manage.py", and "requirements.txt".

The main workspace shows a terminal tab labeled "powershell - lab_backend" with the command "dotnet run" and its output. The output indicates successful installation of dependencies and running the application.

```
[dotnet] A new release of pip is available: 26.3.1 > 26.8.1
[dotnet] To update, run: python -m pip install --upgrade pip
(venv) PS Y:\prueba_complexivo\laboratorio\lab_backend> dotnet run
(venv) PS Y:\prueba_complexivo\laboratorio\lab_backend> cd ..
(venv) PS Y:\prueba_complexivo\laboratorio> cd \Lab_backend
(venv) PS Y:\prueba_complexivo\laboratorio\lab_backend> [REDACTED]
```

Captura 16 – Migraciones Ejecutadas

Mostrar ejecución de makemigrations y migrate.



The screenshot shows a Windows desktop environment. In the foreground, there is a Microsoft Word document titled "Formato para Evidencias Examen Mundial". The document contains a table of contents for an exam, including sections like "EXAMEN COMPLEJO PRÁCTICO", "Indicaciones Generales", and several sections labeled "SECCIÓN 1", "SECCIÓN 2", etc. In the background, a PowerShell window titled "Windows PowerShell" is open. The command `(venv) PS Y:\prueba_complexivo\laboratorio\lab_backend> python manage.py makemigrations` is run, followed by `(venv) PS Y:\prueba_complexivo\laboratorio\lab_backend> python manage.py migrate`. The output shows the migration process running successfully. Below this, another command `(venv) PS Y:\prueba_complexivo\laboratorio\lab_backend> python manage.py runserver 0.0.0.0:8000` is run, and the server starts successfully. A browser window at the bottom shows the UTE website (www.ute.edu.ec) with a phone number (593 2 299-0800) and social media links.

Captura 17 – Servidor en Ejecución

Mostrar el servidor corriendo correctamente.



The screenshot shows a Windows desktop environment. In the center is a terminal window titled "Windows PowerShell" running on a path starting with "Y:\prueba_complexivo\laboratorio\lab_backend". The command entered is "python manage.py runserver 0.0.0.0:8000". The output shows the server is starting at "http://0.0.0.0:8000" and a warning about using a development server in production. Below the terminal is a code editor window for "laboratorio" showing the project structure and files. The bottom of the screen shows the taskbar with various icons and the system tray.

Captura 18 – Endpoint GET Funcional

Mostrar respuesta JSON en navegador o Postman.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

The screenshot shows the Postman application interface. On the left, the sidebar lists various collections and environments. The main workspace is titled "elian Muriel's Workspace". A collection named "API Lab Backend" is expanded, showing several requests under "Autenticación" and "Pruebas de laboratorio". The "Listar pruebas" request is selected, showing its details. The "Body" tab displays a JSON response:

```
1 {  
2   "count": 1,  
3   "next": null,  
4   "previous": null,  
5   "results": [  
6     {  
7       "id": 1,  
8       "test_name": "Prueba de sangre",  
9       "sample_type": "Sangre",  
10      "price": "2.00",  
11      "is_available": true  
12    }  
13  ]  
14 }
```

The status bar at the bottom indicates a "200 OK" response with a duration of "34 ms" and a size of "479 B".

Captura 19 – Endpoint POST Funcional

Mostrar inserción exitosa de un registro.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

The screenshot shows the Postman application interface. On the left, there's a sidebar with collections like 'API Lab Backend', 'Autenticación', 'Pruebas de laboratorio', 'Órdenes de laboratorio', and various other projects. The main area shows a collection named 'API Lab Backend / Pruebas de laboratorio / Crear prueba'. A POST request is selected with the URL `(baseURL) /api/lab-tests/`. The 'Body' tab is active, showing a JSON payload:

```
1 {
2   "test_name": "Prueba de orina",
3   "sample_type": "Orina",
4   "price": "0.75",
5   "is_available": true
6 }
```

Below the body, the response status is '201 Created' with a response time of '31 ms' and a size of '429 B'. The bottom of the screen shows the Windows taskbar with various icons.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

SECCIÓN 4: FRONTEND – REACTJS

Captura 20 – Proyecto React en Ejecución

Mostrar terminal con servidor activo.

The screenshot shows a Windows desktop environment. In the center, there is a web browser window titled "lab-ui" with the URL "localhost:5173". The page content is in Spanish and discusses order management. Below the browser is a terminal window titled "C:\WINDOWS\system32\cmd." showing npm audit results and VITE logs. The taskbar at the bottom includes icons for various applications like File Explorer, Task Manager, and a search bar. The system tray shows the date and time as "8:23 PM 2/27/2026".

```
C:\WINDOWS\system32\cmd. + 
68 packages are looking for funding
  run 'npm fund' for details

3 vulnerabilities (1 moderate, 2 high)

To address all issues, run:
  npm audit fix

Run 'npm audit' for details.
PS Y:\prueba_complexivo\laboratorio\lab-ui> npm run dev
> lab-ui@0.0.0 dev
> vite

VITE v7.3.1 ready in 317 ms
→ Local: http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help
8:16:32 PM [vite] (client) hr update /src/index.css
8:18:21 PM [vite] (client) hr update /src/App.tsx
8:18:42 PM [vite] (client) hr update /src/pages/AboutPage.tsx
8:18:56 PM [vite] (client) hr update /src/pages/AdminHomePage.tsx
8:19:04 PM [vite] (client) hr update /src/pages/AdminHomePage.tsx (x2)
8:20:50 PM [vite] (client) hr update /src/pages/HomePage.tsx
8:21:17 PM [vite] (client) hr update /src/pages/PublicOrdersPage.tsx
8:22:18 PM [vite] (client) hr update /src/pages/AdminHomePage.tsx
```

Captura 21 – Listado de Registros

Mostrar en navegador la lista obtenida desde el backend.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

A screenshot of a web browser window titled "lab-ui" showing the URL "localhost:5173/ordenes". The page has a blue header bar with the text "Laboratorio Muriel Complexivo" and navigation links: HOME, ACERCA DE, LISTADO DE ÓRDENES, INICIAR SESIÓN, and ADMINISTRADOR. Below the header is a table titled "Estado de Órdenes" with the following data:

ID	Prueba	Paciente	Estado	Resumen resultado
1	Prueba de sangre	Elian Muriel	CREATED	—

There is a "RECARGAR" button at the top right of the table area.

A screenshot of a Windows desktop. The taskbar shows various pinned icons and the URL "localhost:5173/ordenes" in the start button search field. The system tray shows the date and time as "8:24 PM 2/27/2026".

Captura 22 – Registro Nuevo Desde la Interfaz

Mostrar creación de un nuevo registro y actualización del listado.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

lab-ui localhost:5173/admin/pruebas Action required

Facebook Google Daddy Yankee - Sha... Ver Diarios de Vamp... Home - ROBLOX Call of War Inscripciones para L... Wormaxio | Un jute... [tutorial] PDFJunior... Home - Fuerza Aére... @POP3.rar - Googl... Calificaciones Vista All Bookmarks

Laboratorio Muriel Complexivo

HOME ACERCA DE LISTADO DE ÓRDENES INICIAR SESIÓN ADMINISTRADOR

Admin Pruebas

ID	Prueba	Tipo muestra	Precio	Disponible	Acciones
1	Prueba de sangre	Sangre	2.50	Si	
2	Prueba de orina	Orina	0.75	Si	
3	Prueba de eses	Eses	1.25	Si	

Search 8:32 PM
2/27/2026



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

Laboratorio Muriel Complexivo

HOME ACERCA DE LISTADO DE ÓRDENES INICIAR SESIÓN ADMINISTRADOR

ID	Prueba	Tipo muestra	Precio	Disponible	Acciones
1	Prueba de sangre	Sangre	2.50	Sí	
2	Prueba de orina	Orina	0.75	Sí	
3	Prueba de éses	Eses	1.25	Sí	
4	Prueba de Glucosa	Sangre	1.10	Sí	



SECCIÓN 5: APLICACIÓN MÓVIL – REACT NATIVE

Captura 23 – Proyecto Móvil Creado

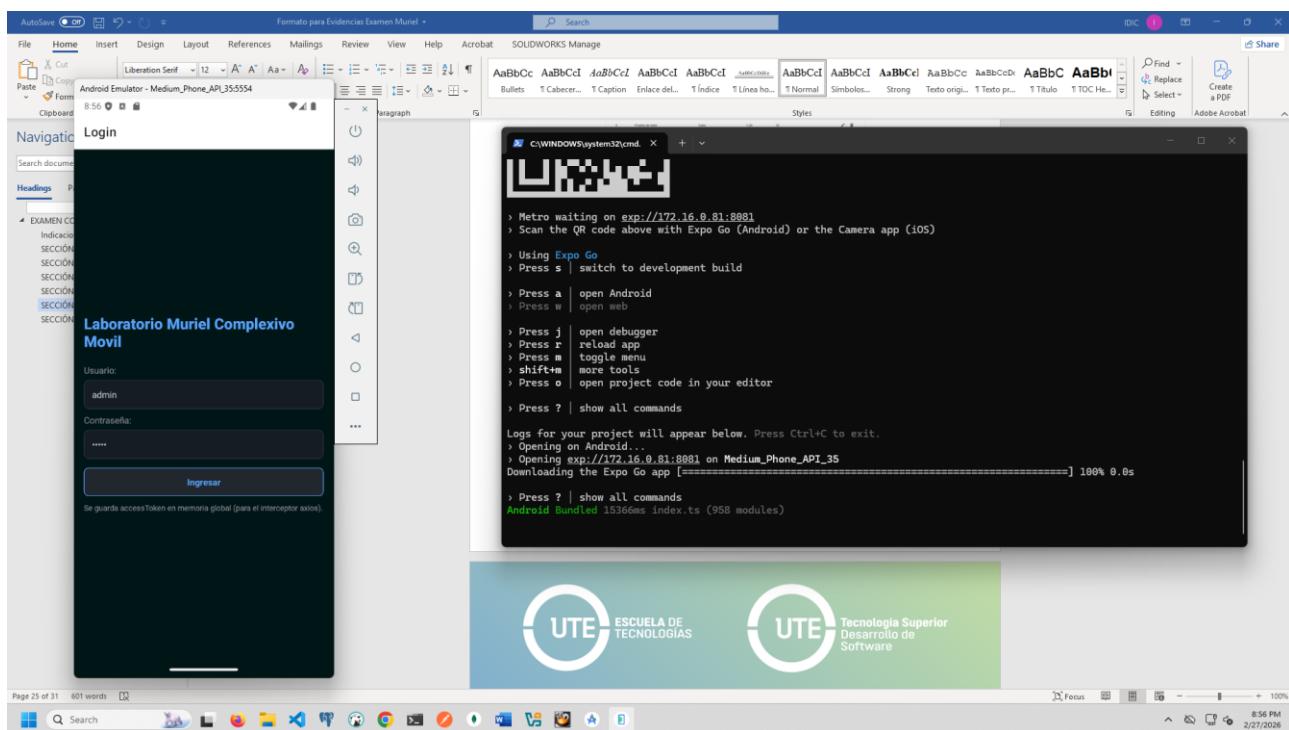
Mostrar terminal con proyecto inicializado.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



Captura 24 – Aplicación Ejecutándose

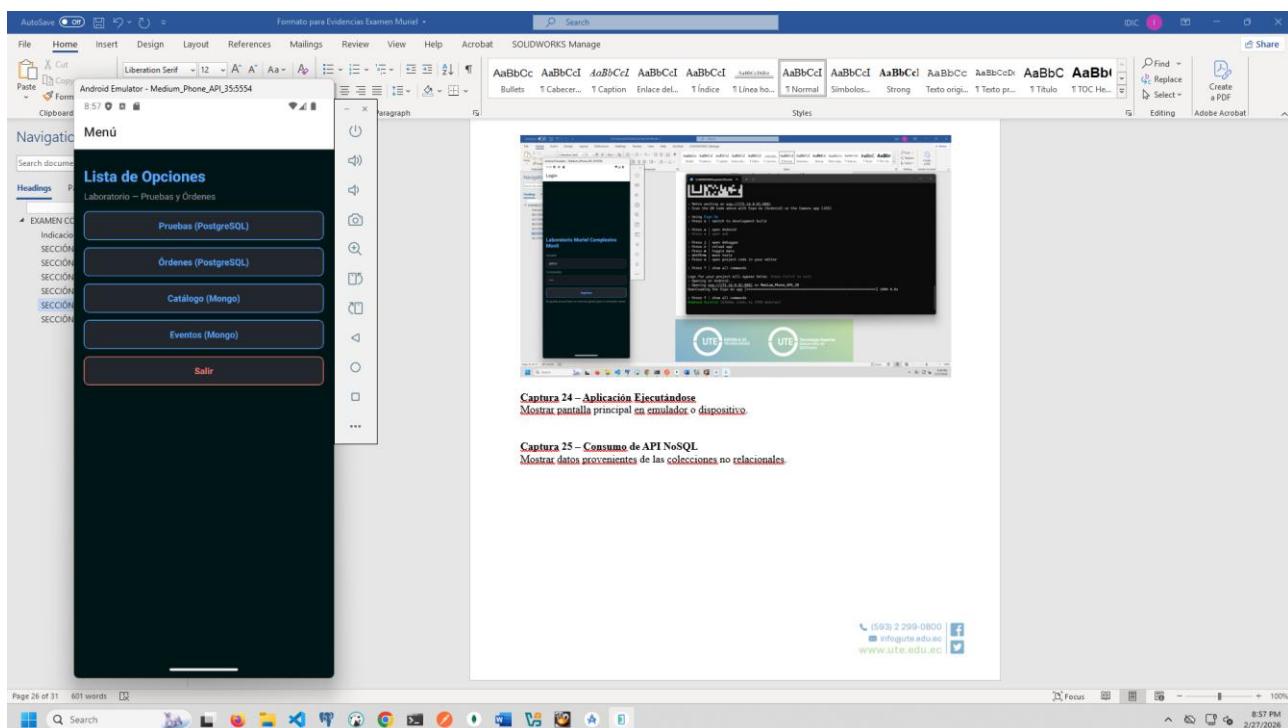
Mostrar pantalla principal en emulador o dispositivo.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



Captura 25 – Consumo de API NoSQL

Mostrar datos provenientes de las colecciones no relacionales.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software

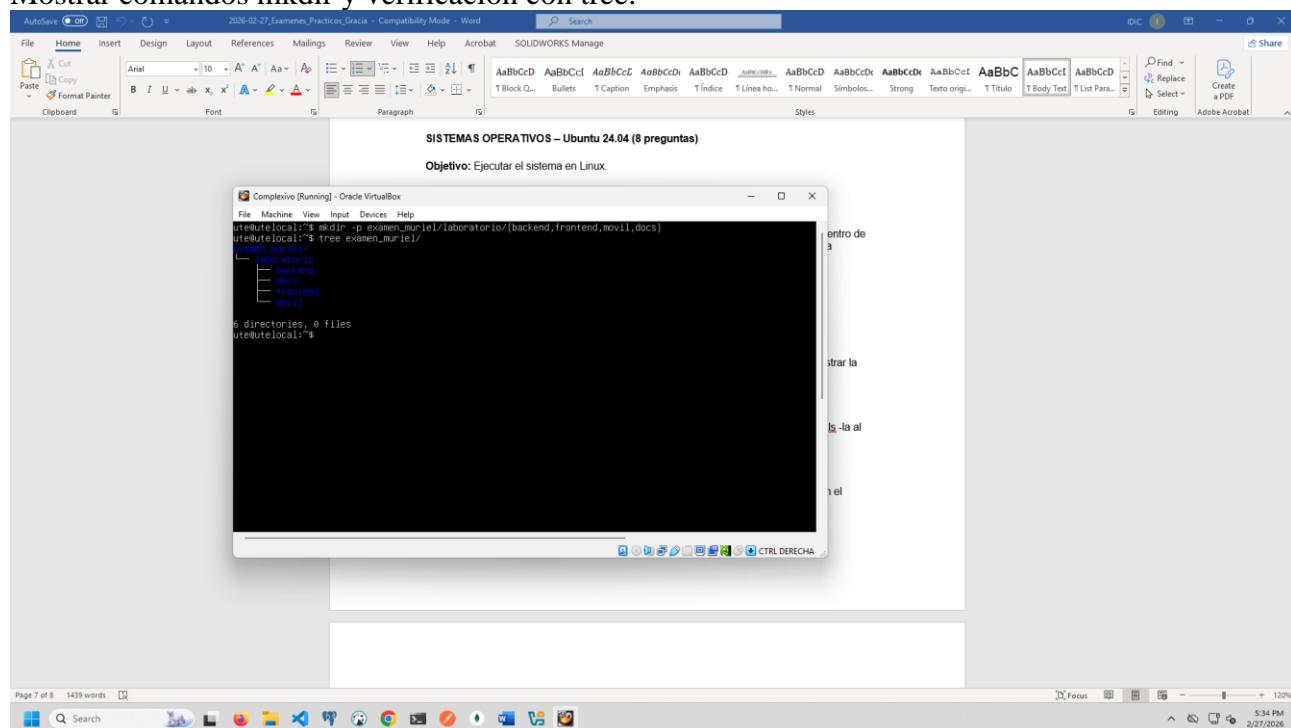
Screenshot of Microsoft Word document titled "Formato para Evidencias Examen Mural". The document contains a screenshot of a MongoDB interface showing a list of orders. A caption below the screenshot reads: "Captura 25 – Consumo de API NoSQL. Mostrar datos provenientes de las colecciones no relacionales." The Microsoft Word ribbon is visible at the top, and the Windows taskbar is at the bottom.



SECCIÓN 6: SISTEMAS OPERATIVOS – UBUNTU (MÁQUINA VIRTUAL)

Captura 26 – Creación de Estructura de Directorios

Mostrar comandos mkdir y verificación con tree.



Captura 27 – Navegación y Listado

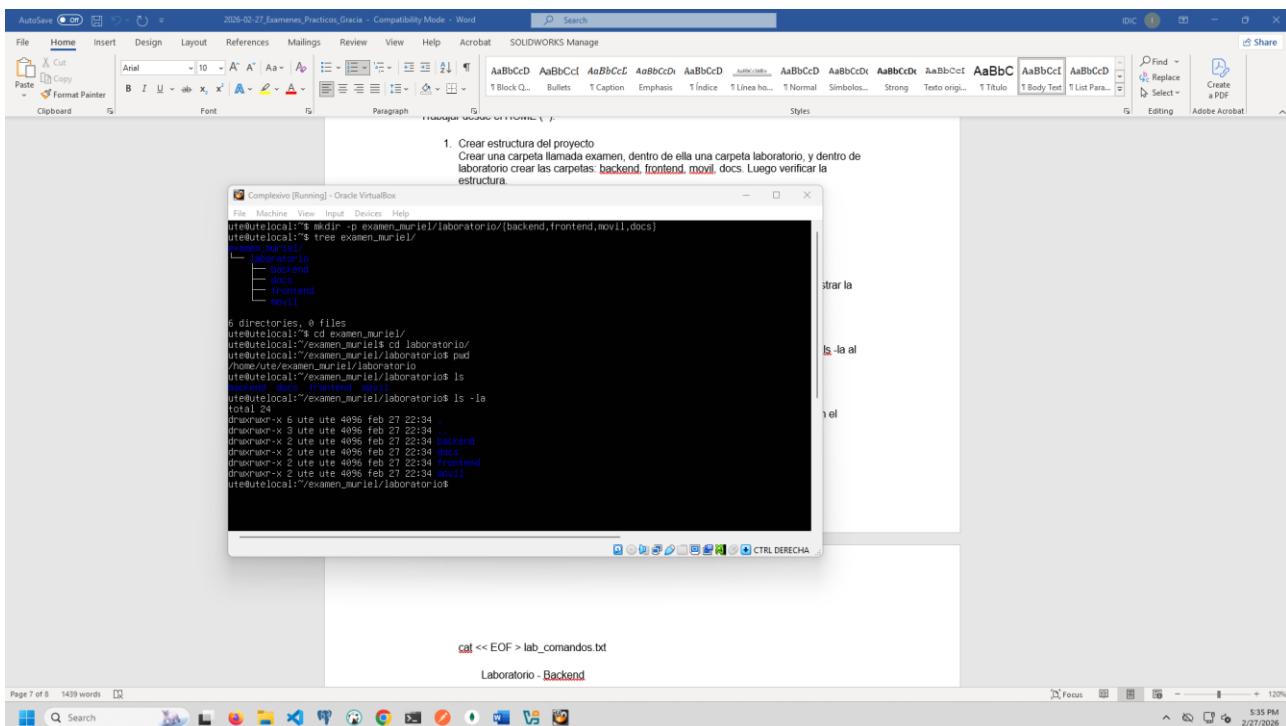
Mostrar uso de cd, pwd y ls -la.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



Captura 28 – Redirección de Salida

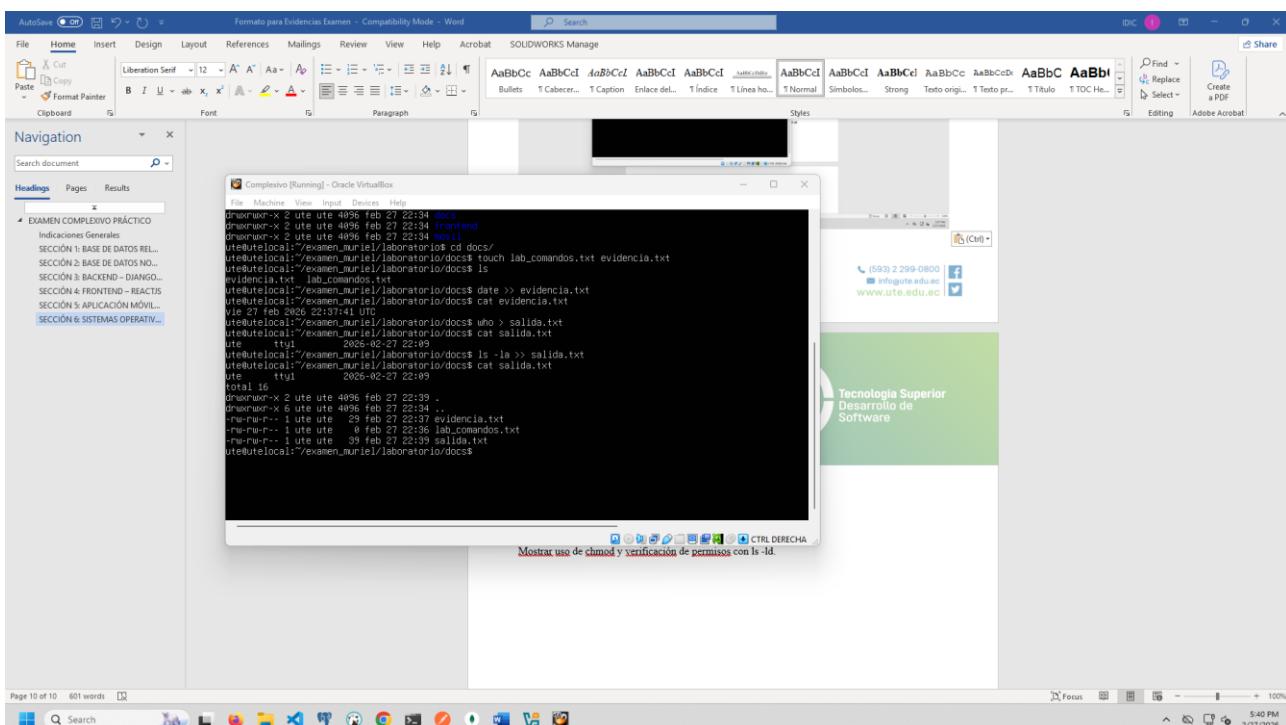
Mostrar uso de > y >> con comandos básicos.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



Captura 29 – Búsqueda con grep

Mostrar búsqueda de texto dentro de un archivo.



The screenshot shows a Microsoft Word document window titled "Complejivo [Running] - Oracle VirtualBox". The content of the document is a terminal session output:

```
Administrator:~/examen_muriel/laboratorio/docs$ cat << EOF > lab_comandos.txt
>Laboratorio - Backend
> GET /api/lab-tests/
> GET /api/lab-orders/
> POST /api/lab-orders/
> DELETE /api/lab-orders/10/
> INFO: lab order created successfully
> INFO: lab service running
> WARN: lab orders timeout detected
> EOF
ute@utelocal:~/examen_muriel/laboratorio/docs$ cat lab_comandos.txt
Laboratorio - Backend
GET /api/lab-tests/
GET /api/lab-orders/
POST /api/lab-orders/
DELETE /api/lab-orders/10/
INFO: lab order created successfully
INFO: lab service running
WARN: lab orders timeout detected
ute@utelocal:~/examen_muriel/laboratorio/docs$ grep lab-orders lab_comandos.txt
GET /api/lab-orders/
POST /api/lab-orders/
DELETE /api/lab-orders/10/
ute@utelocal:~/examen_muriel/laboratorio/docs$ grep -n lab-orders lab_comandos.txt
:1:GET /api/lab-orders/
:2:POST /api/lab-orders/
:3:DELETE /api/lab-orders/10/
ute@utelocal:~/examen_muriel/laboratorio/docs$
```

Captura 30 – Permisos y Sticky Bit

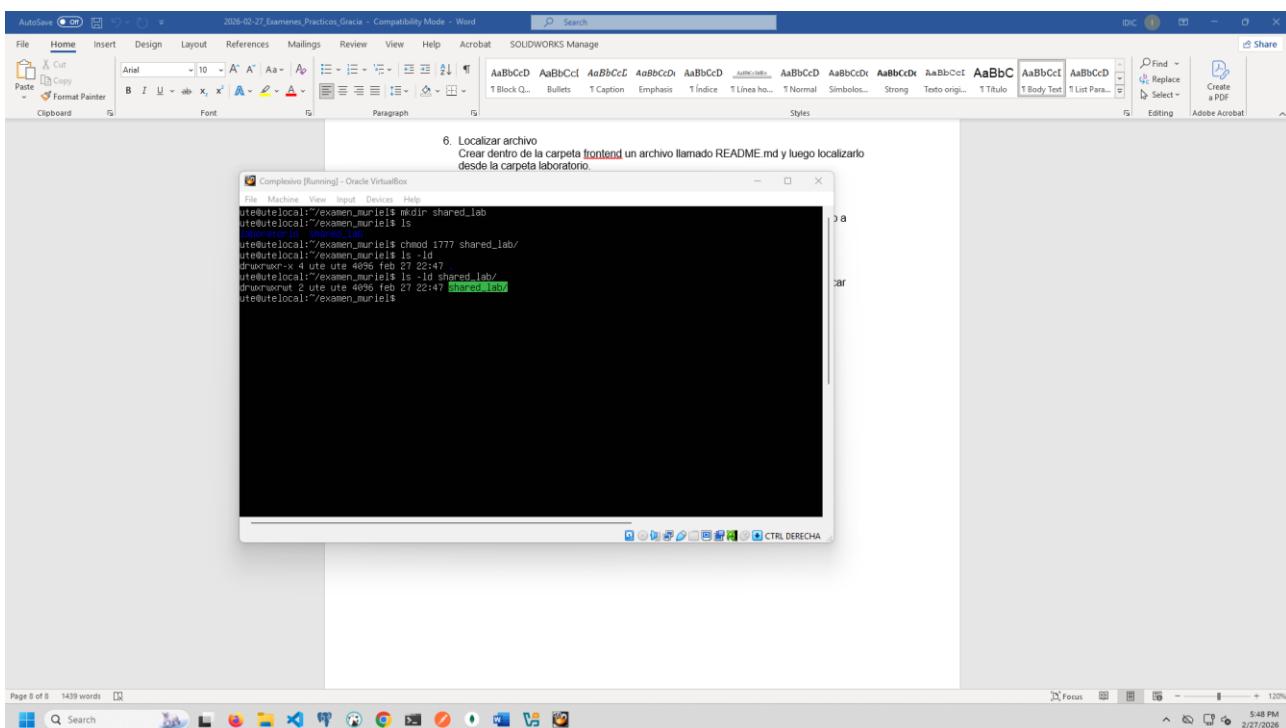
Mostrar uso de chmod y verificación de permisos con ls -ld.



ESCUELA DE
TECNOLOGÍAS



Tecnología Superior
Desarrollo de
Software



Link del repo: https://github.com/ElianMuriel/Laboratorio_complexivo_ute.git