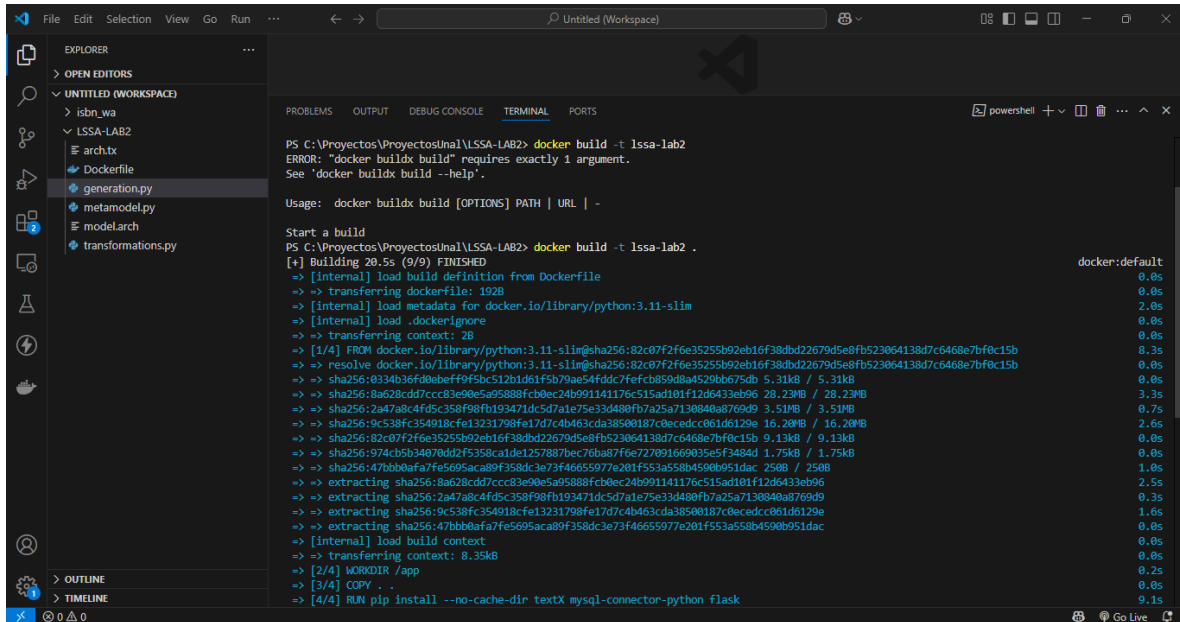


Yilver Alirio Ramírez Ochoa

Create a *Docker image* from the *Dockerfile* file.



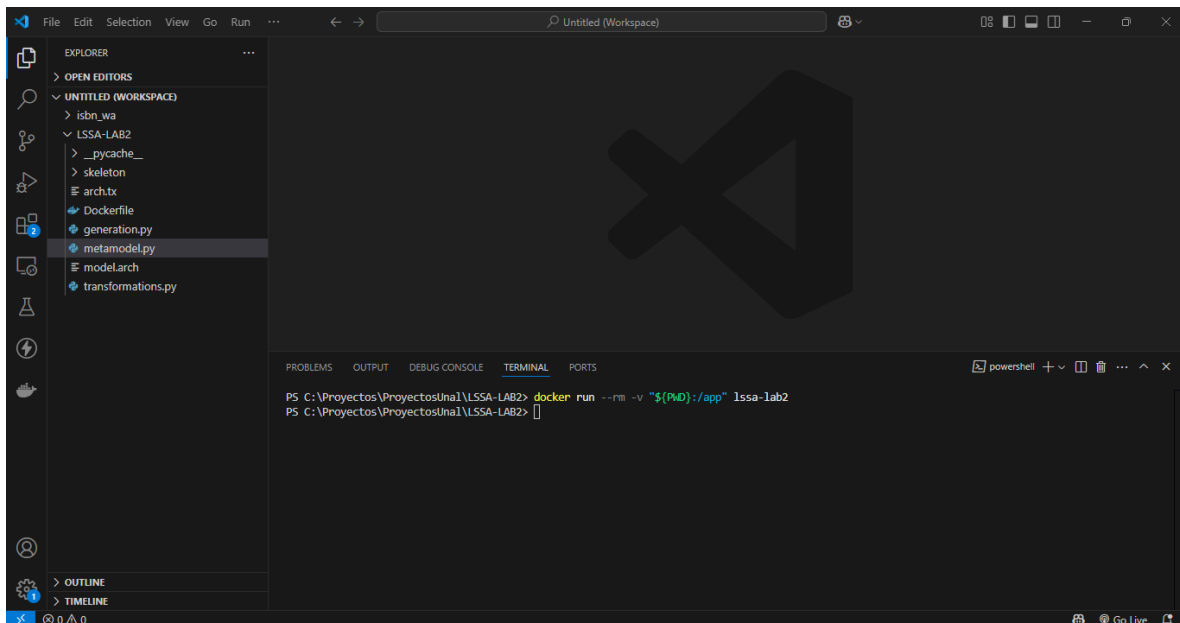
The screenshot shows the Visual Studio Code interface with the Explorer panel on the left displaying a project structure. The file `generation.py` is selected. The Terminal panel at the bottom shows the execution of the `docker build` command. The output indicates that the build is successful, with a progress bar showing the completion of the build process. The build log includes details about the Dockerfile, the context, and the layers being built.

```
PS C:\Proyectos\ProyectosUnal\LSSA-LAB2> docker build -t lssa-lab2
ERROR: "docker buildx build" requires exactly 1 argument.
See 'docker buildx build --help'.

Usage: docker buildx build [OPTIONS] PATH | URL | -

Start a build
PS C:\Proyectos\ProyectosUnal\LSSA-LAB2> docker build -t lssa-lab2 .
[*] Building 20.5s (9/9) FINISHED
-> [internal] load build definition from Dockerfile
-> => transferring dockerfile: 192B
-> [internal] load metadata for docker.io/library/python:3.11-slim
-> [internal] load .dockerignore
-> => transferring context: 2B
-> [1/4] FROM docker.io/library/python:3.11-slim@sha256:82c07f2f6e3525b92eb16f38bd22679d5e8fb523064138d7c6468e7bf0c15b
-> => resolve docker.io/library/python:3.11-slim@sha256:82c07f2f6e3525b92eb16f38bd22679d5e8fb523064138d7c6468e7bf0c15b
-> => sha256:834b36f40ebef9f95c512b1d51f5b79ae5fddc7fe6c859d8a45290b675db 5.31kB / 5.31kB
-> => sha256:8b02edd7cc8b9e9e5a95888fcb0ec24b991141176c515ad101f12d6433eb96 28.23MB / 28.23MB
-> => sha256:2a47a8c4fd5c358f98fb193471dc5d7a1e75e33d480fb7a25a7130840a8769d9 3.51MB / 3.51MB
-> => sha256:9c538fc354918cfe13231798fe17d7c4b463cda38500187c0ecedcc061d6129e 16.20MB / 16.20MB
-> => sha256:82c07f2f6e3525b92eb16f38bd22679d5e8fb523064138d7c6468e7bf0c15b 9.13kB / 9.13kB
-> => sha256:974cb5b34070dd2f5358c1de1257887bec76ba87f6e727091669035e5f3484d 1.75kB / 1.75kB
-> => sha256:47bbb0afa7fe5695aca89f358dc3e73f46655977e201f553a558b4590b951dac 250B / 250B
-> => extracting sha256:8a628cdd7ccc83e90e5a95888fcb0ec24b991141176c515ad101f12d6433eb96
-> => extracting sha256:2a47a8c4fd5c358f98fb193471dc5d7a1e75e33d480fb7a25a7130840a8769d9
-> => extracting sha256:9c538fc354918cfe13231798fe17d7c4b463cda38500187c0ecedcc061d6129e
-> => extracting sha256:47bbb0afa7fe5695aca89f358dc3e73f46655977e201f553a558b4590b951dac
-> [internal] load build context
-> => transferring context: 8.35kB
-> [2/4] WORKDIR /app
-> [3/4] COPY . .
-> [4/4] RUN pip install --no-cache-dir textX mysql-connector-python flask
```

Create a *Docker container* to execute the program and generate the modeled software system.

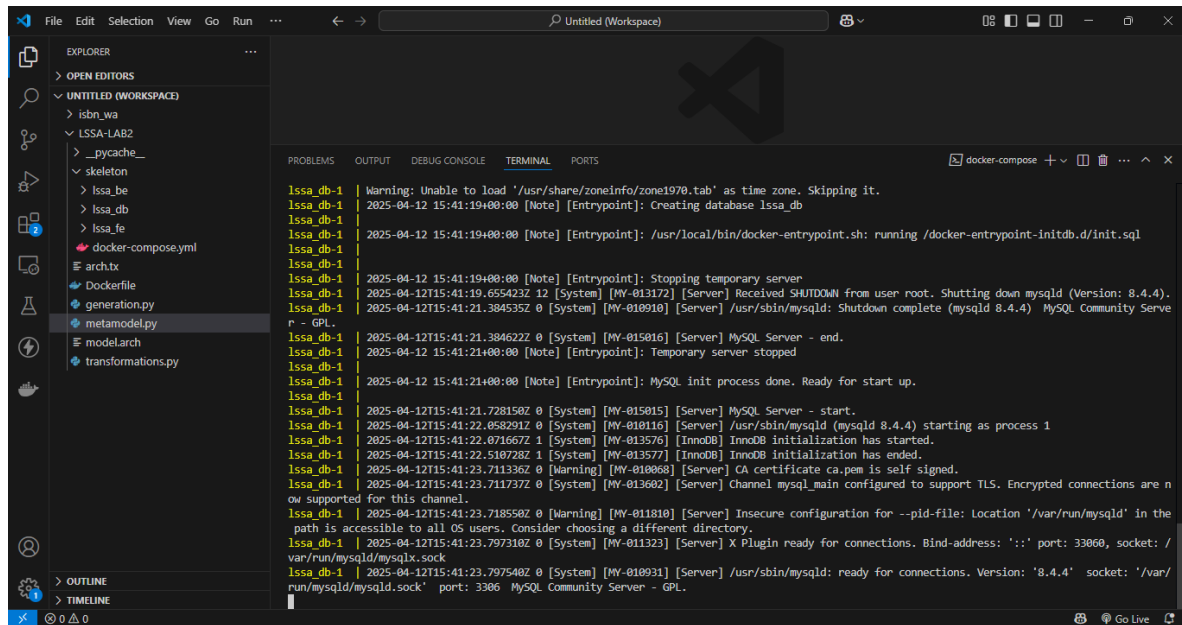


The screenshot shows the Visual Studio Code interface with the Explorer panel on the left displaying a project structure. The file `generation.py` is selected. The Terminal panel at the bottom shows the execution of the `docker run` command. The output indicates that the container is successfully created and started, and the program is executed. The build log includes details about the Dockerfile, the context, and the layers being built.

```
PS C:\Proyectos\ProyectosUnal\LSSA-LAB2> docker run --rm -v "${PWD}:/app" lssa-lab2
PS C:\Proyectos\ProyectosUnal\LSSA-LAB2>
```

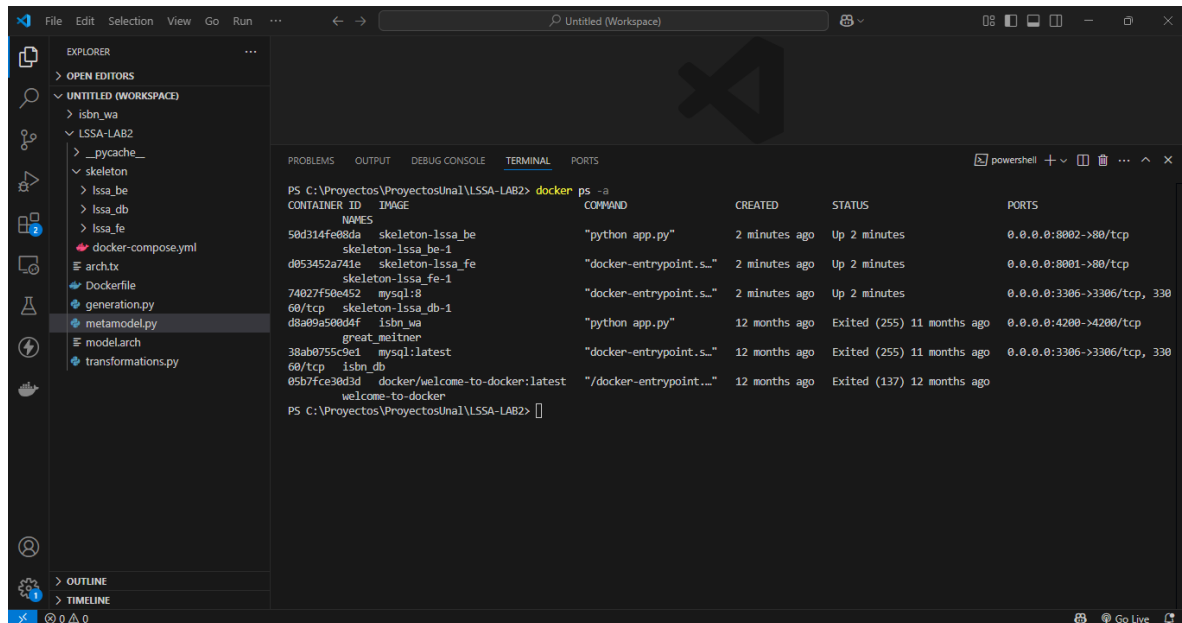
d. Enter to the `/skeleton` directory.

e. Execute the generated skeleton of the modeled software system.



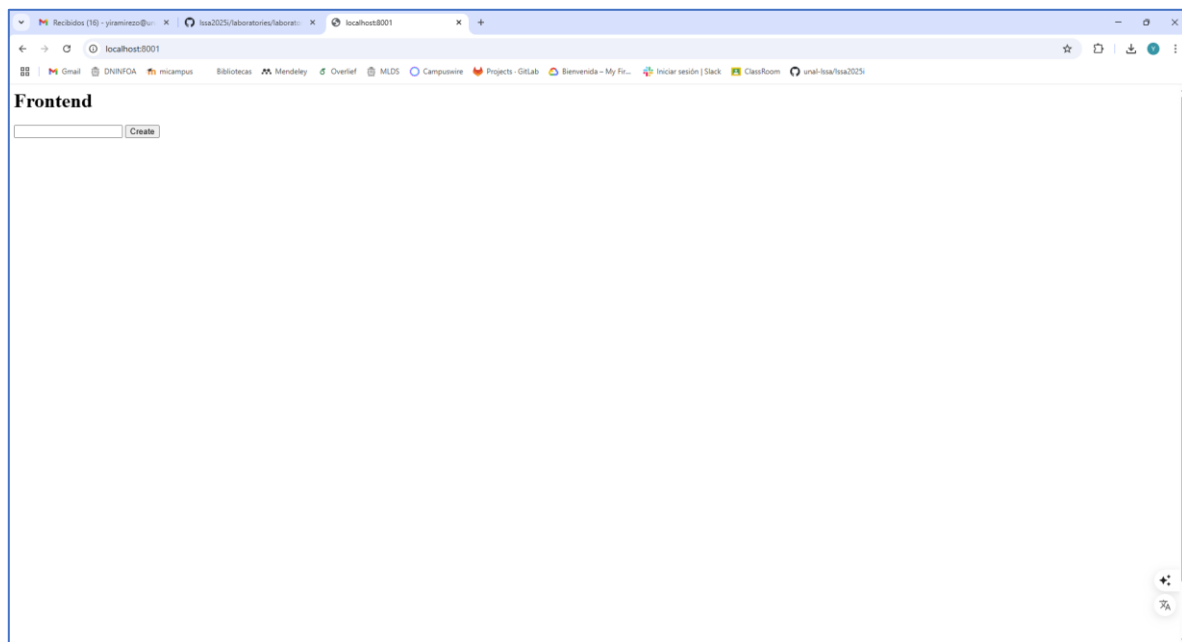
```
Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone. Skipping it.
lssa_db-1 | 2025-04-12 15:41:19+00:00 [Note] [Entrypoint]: Creating database lssa_db
lssa_db-1 | 2025-04-12 15:41:19+00:00 [Note] [Entrypoint]: /usr/local/bin/docker-entrypoint.sh: running /docker-entrypoint-initdb.d/init.sql
lssa_db-1 | 2025-04-12 15:41:19+00:00 [Note] [Entrypoint]: Stopping temporary server
lssa_db-1 | 2025-04-12 15:41:19.655423Z 12 [System] [MY-013172] [Server] Received SHUTDOWN from user root. Shutting down mysqld (Version: 8.4.4).
lssa_db-1 | 2025-04-12 15:41:21.384535Z 0 [System] [MY-010910] [Server] /usr/sbin/mysqld: Shutdown complete (mysqld 8.4.4) MySQL Community Server - GPL.
lssa_db-1 | 2025-04-12 15:41:21.384622Z 0 [System] [MY-015016] [Server] MySQL Server - end.
lssa_db-1 | 2025-04-12 15:41:21+00:00 [Note] [Entrypoint]: Temporary server stopped
lssa_db-1 | 2025-04-12 15:41:21+00:00 [Note] [Entrypoint]: MySQL init process done. Ready for start up.
lssa_db-1 | 2025-04-12 15:41:21.728158Z 0 [System] [MY-015015] [Server] MySQL Server - start.
lssa_db-1 | 2025-04-12 15:41:22.058291Z 0 [System] [MY-010116] [Server] /usr/sbin/mysqld (mysqld 8.4.4) starting as process 1
lssa_db-1 | 2025-04-12 15:41:22.071667Z 1 [System] [MY-013576] [InnoDB] InnoDB initialization has started.
lssa_db-1 | 2025-04-12 15:41:22.510728Z 1 [System] [MY-013577] [InnoDB] InnoDB initialization has ended.
lssa_db-1 | 2025-04-12 15:41:23.711336Z 0 [Warning] [MY-010068] [Server] CA certificate ca.pem is self signed.
lssa_db-1 | 2025-04-12 15:41:23.711737Z 0 [System] [MY-013602] [Server] Channel mysql_main configured to support TLS. Encrypted connections are now supported for this channel.
lssa_db-1 | 2025-04-12 15:41:23.718558Z 0 [Warning] [MY-011810] [Server] Insecure configuration for --pid-file: Location '/var/run/mysqld' in the path is accessible to all OS users. Consider choosing a different directory.
lssa_db-1 | 2025-04-12 15:41:23.797310Z 0 [System] [MY-011323] [Server] X Plugin ready for connections. Bind-address: '::' port: 33060, socket: /var/run/mysqld/mysqld.sock
lssa_db-1 | 2025-04-12 15:41:23.797540Z 0 [System] [MY-010931] [Server] /usr/sbin/mysqld: ready for connections. Version: '8.4.4' socket: '/var/run/mysqld/mysqld.sock' port: 3306 MySQL Community Server - GPL.
```

f. Verify the executed containers (`skeleton-lssa_db-1`, `skeleton-lssa_be-1` and `skeleton-lssa_fe-1`):

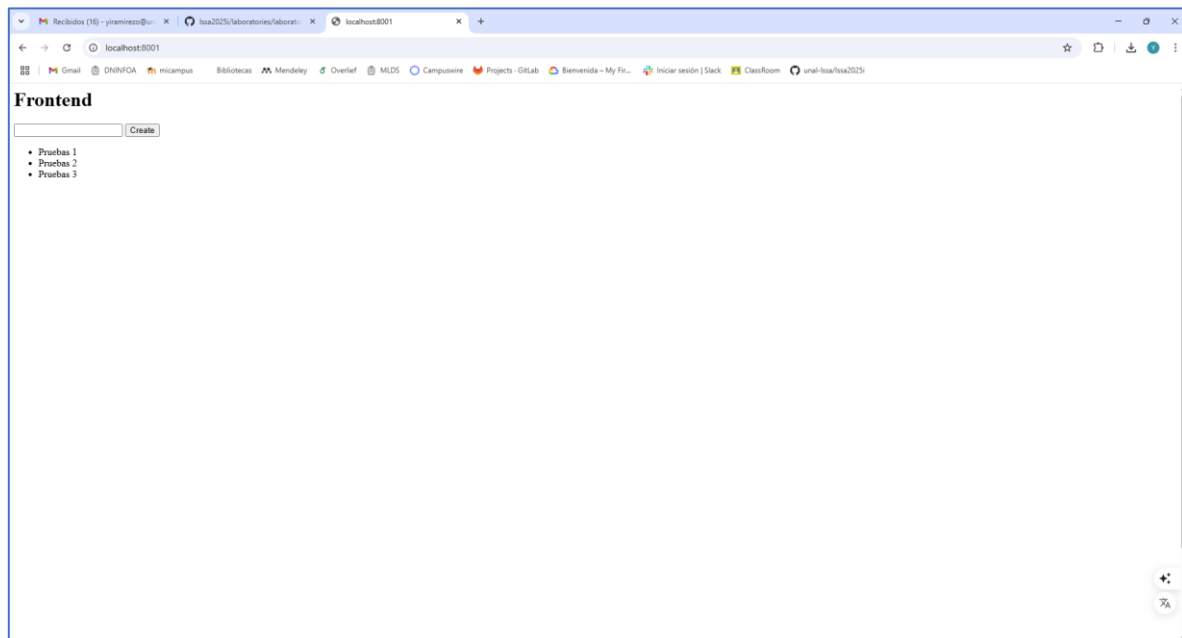


```
PS C:\Projectos\ProjectosUnal\LSSA-LAB2> docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
58d314fe88da   skeleton-lssa_be                     "python app.py"         2 minutes ago   Up 2 minutes   0.0.0.0:8002->80/tcp
d053452a741e   skeleton-lssa_fe                     "docker-entrypoint.s..." 2 minutes ago   Up 2 minutes   0.0.0.0:8001->80/tcp
74027f50e452   mysql:8                              "docker-entrypoint.s..." 2 minutes ago   Up 2 minutes   0.0.0.0:3306->3306/tcp, 3306/tcp
60/tcp        skeleton-lssa_db-1                   "python app.py"         12 months ago   Exited (255) 11 months ago   0.0.0.0:4200->4200/tcp
d8a09a500d4f   isbn_wa                              "python app.py"         12 months ago   Exited (255) 11 months ago   0.0.0.0:3306->3306/tcp, 3306/tcp
38ab0755c9e1   isbn_db                              "docker-entrypoint.s..." 12 months ago   Exited (255) 11 months ago   0.0.0.0:3306->3306/tcp, 3306/tcp
05b7fce3803d   docker/welcome-to-docker:latest     "/docker-entrypoint..." 12 months ago   Exited (137) 12 months ago
```

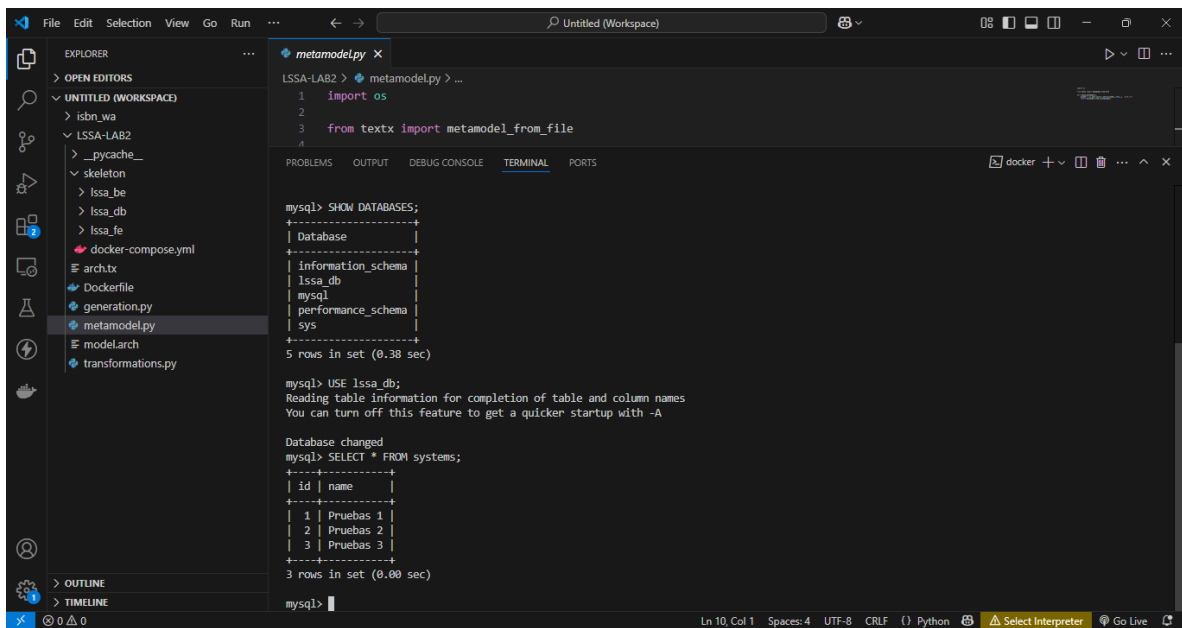
g. Open the frontend component in a web browser: <http://localhost:8001>.



h. Create a new element (system) using the user interface.



i. Check the new element in the database component:



The screenshot shows a Visual Studio Code editor with a workspace named 'Untitled (Workspace)'. The Explorer sidebar on the left shows a project structure with folders like 'isbn_wa', 'LSSA-LAB2', and files like 'docker-compose.yml', 'Dockerfile', 'generation.py', 'metamodel.py', 'modelarch', and 'transformations.py'. The 'metamodel.py' file is open in the editor, showing the following code:

```
1 import os
2
3 from textx import metamodel_from_file
4
```

The Terminal window at the bottom shows a MySQL command prompt session. The first command is 'mysql> SHOW DATABASES;', which returns a list of databases: 'information_schema', 'lssa_db', 'mysql', 'performance_schema', and 'sys'. The second command is 'mysql> USE lssa_db;', which changes the database to 'lssa_db'. The third command is 'mysql> SELECT * FROM systems;', which returns a table with 3 rows and 2 columns: 'id' and 'name'. The table data is as follows:

id	name
1	Pruebas 1
2	Pruebas 2
3	Pruebas 3

The status bar at the bottom indicates the current line and column (Ln 10, Col 1), the file encoding (UTF-8), the line ending (CRLF), the Python interpreter (Python), and the Docker container (Go Live).