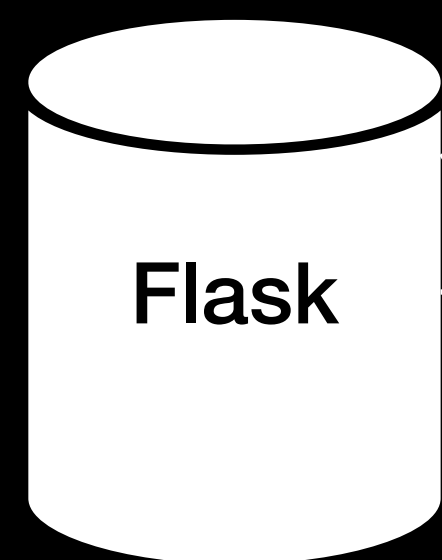
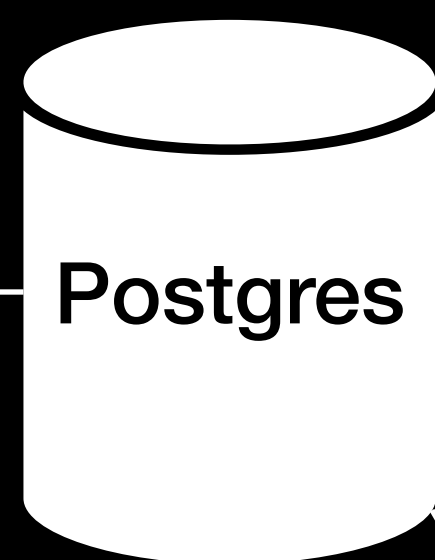


Bike Sharing

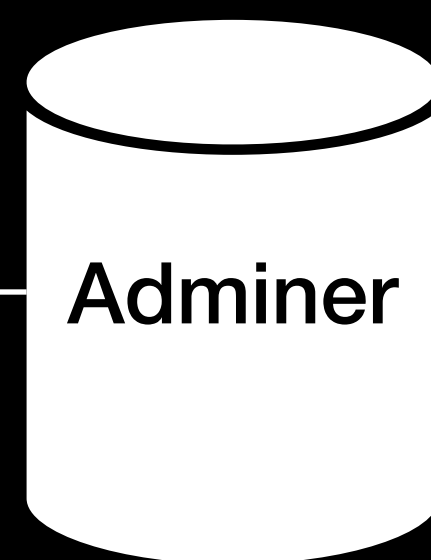
Architecture



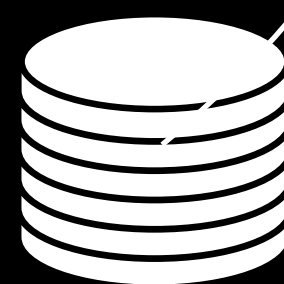
Webapp



Relational
Database



Database
Management



Volume

HTTP
Port: 80

HTTP
Port: 8080



Client



Map API

Database

= Primary Key, Underlined = Foreign Key



- PostgreSQL running in Docker Container
- Docker Volume mounted to /var/lib/postgresql/data to persist data

Bike

#id	name	x_coordinate	y_coordinate
Integer	String	Float	Float

Ride

#id	<u>bike_id</u>	<u>user_id</u>	start_time	end_time
Integer	Integer	Integer	DateTime	DateTime

User

#id	email	Password
Integer	String	String

Role

#id	name
Integer	String

RolesUsers

#id	<u>user_id</u>	<u>role_id</u>
Integer	Integer	Integer

Database Management

- Adminer running in Docker Container
- Web based
- Accessible through port 8080

Webserver



- Flask running in Docker Container
- Python based
- Accessible through port 80
- Receives request from clients and responds
- Handles authentication and authorisation (flask-security-too)
- REST API (accepts JSON)
- Database connection managed by SQLAlchemy
- Jinja templates for frontend