

# sllLySaUsAgEsQuAD Design Document

P1M\_Jason\_Chao, DALEX luo, Devo Leon Huang, DeV0(00) Stanley H00(H000)

**TARGET SHIP DATE: {2025-1-10}**

## Overview

sllLySaUsAgEsQuAD's image-based web app integrates image recognition, generation, and user authentication systems with Flask, SQLite, and Bootstrap. The landing page will be available to all logged in users, who will be able to access their profiles and the app's image recognition page. In it, users can either upload their own images or generate one from the Unsplash API. Using other APIs, the app will then specify and show the names of any identifiable objects in said image, as well as their definitions.

## Site Preview

### Home Page

- Access Control: for logged-in users only. Non authenticated users redirect to the login page.
- Navbar with links to profile page and image request form.
- Overview of application (RESTful APIS)

### User Profile Page

- Displays past image requests and detection history
- Allows users to revise/delete prior entries

### Login/Registration Page

- Form fields for username and password
- Displays error messages for failed login attempts
- Redirects users to home page after login

### API Results Page

- Form fields for image source selection. Allow users to input their own images.
- Options for object detection & text conversion
  - Identifies objects in image
  - Loads text output from AI API sources based on objects

## Program Components

### Flask App

- `__init__.py`

### HTML Templates

- `home.html`:
- `profile.html`:

- login.html: (may be integrated into home page)
- register.html
- api\_results.html:

#### Middleware Modules

- database.py
- api\_handler.py
- image\_processor.py

#### Static Files

- CSS: style.css

## APIs

[Unsplash](#) | returns random image (needs a card)

[Clarifai](#) | returns a description of the image

[Pixabay](#) | returns videos using the Clarifai description as the search phrase

## Frontend Framework

#### Bootstrap

Bootstrap is both easily usable and readable. Its class attributes are simple and follow the framework's pre-determined design so that code lines are not excessively long (noticed this is the case with Tailwind). In addition, it's one of the most developed and expansive frameworks we have available.

## Database Organization

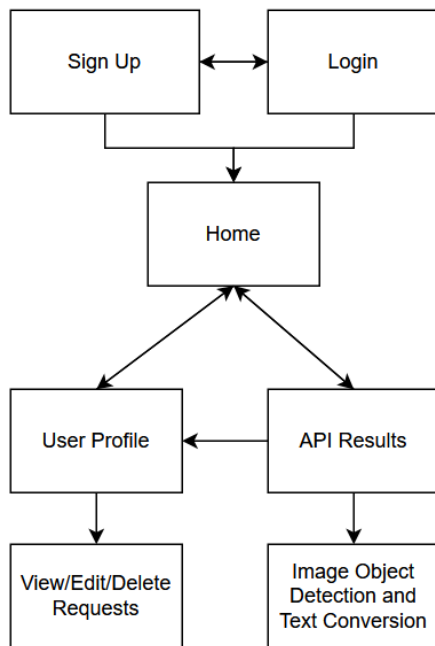
users | user IDs, API keys, login info

id	user_id	username	password_hash
#	AUTOINCREMENT	TEXT	TEXT

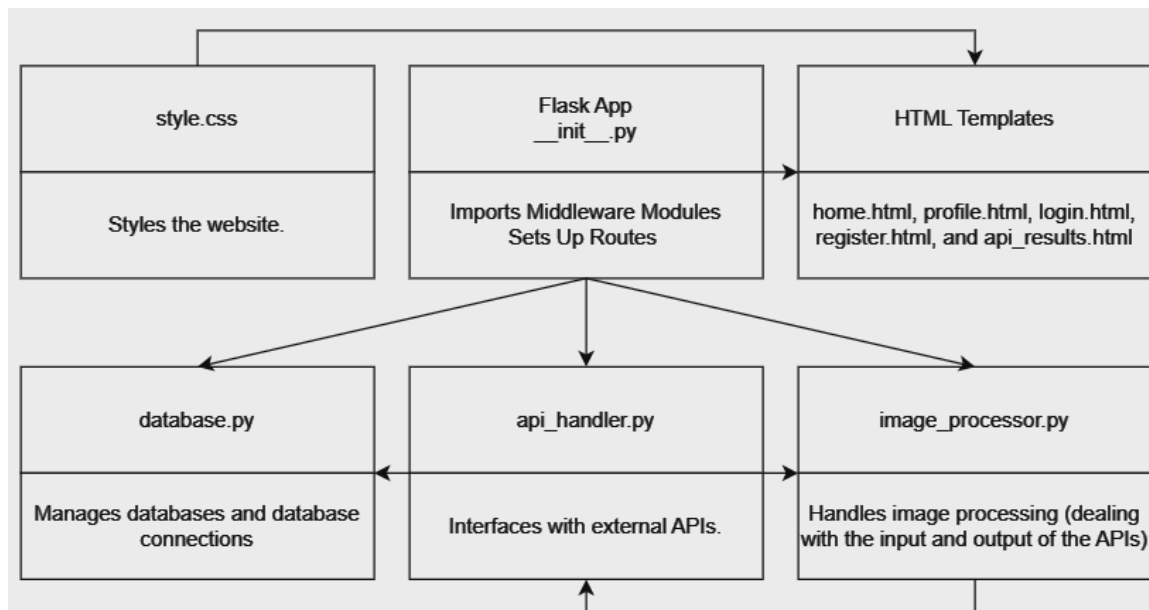
requests | user API requests & responses

id	request_id	user_id	request	response
#	AUTOINCREMENT	REFERENCE users(user_id)	TEXT	TEXT

# Front-End Site Map



# Component Map



# Task Assignments

**Disclaimer:** These task assignments are NOT rigid. Group members may assist each other to ensure punctuality and proper completion of this project.

<b>P1MJC</b>	SQLite Databases
<b>LUO</b>	HTML Templates
<b>H000</b>	API Requests & Responses
<b>DLH</b>	Bootstrap, CSS Styling