

Rotisserie Chicken's Design Doc for *P01 – ArRESTed Development*

↳ Roster: Jake Liu, Veronika Duvanova, Robert Chen, Cindy Liu

PROJECT NAME: Rotisserie Rumble

TARGET SHIP DATE: 2025-12-15

Program Components

sqlite3 (backend data storage system)

user_info (to store account information)

Python (application layer)

app.py (to run everything else)

Flask (web server/delivery framework)

HTML (frontend display):

login.html

register.html

profile.html

floor.html

tarot.html

solitaire.html

blackjack.html

poker.html

APIs:

Deck of Cards API - this API can be utilized in all of our games for basic card functions (shuffling, drawing, etc.)

Pokerkit - this library will be used to facilitate the implementation of complex poker rules and hands.

Tarot Card Meanings API - this will be used to facilitate the implementation of tarot card rules.

Component Explanation

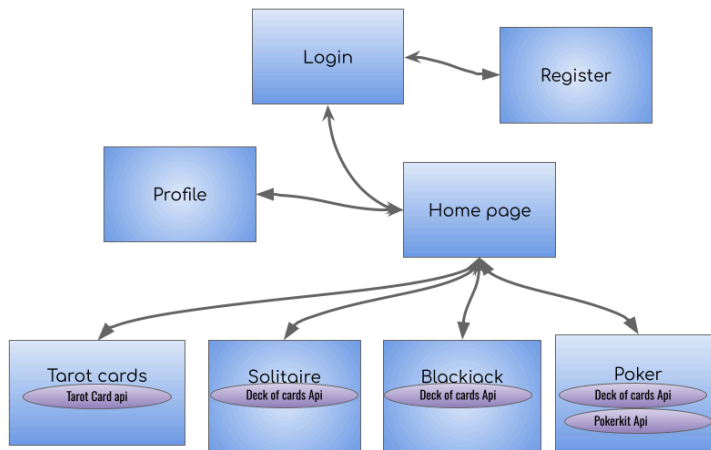
Our sqlite3 database *user_info* will be accessed by:

- **register.html** (to add new user information)
- **login.html** (to access existing users' login information)

- **profile.html** (to access a user's information)
- **tarot.html, solitaire.html, blackjack.html, poker.html** (to access and change balance)

Every page is dynamic because they have user-specific outcomes and/or require access to the *user_info* database to add/remove displayed information.

Component Map:



Database Organization:

USER_INFO			
STRING	user	PK	
STRING	password		
FLOAT	balance		start at \$0, can go into negatives

Member Tasks/Roles:

Jake Liu	Poker, Homepage
Cindy Liu	Solitaire, Login + Register
Veronika Duvanova	Tarot, Profile
Robert Chen	Blackjack, Homepage

3 REST API Selection:

1. <https://www.deckofcardsapi.com/>
2. <https://github.com/wuHakureReimu/SolitaireAPI-for-Python?tab=readme-ov-file>
3. <https://github.com/jeremylee73/heads-up-poker>
4. <https://github.com/chuan-yu/Solitaire-Game-API>
5. <https://www.freepublicapis.com/tarot-card-meanings>
6. <https://pypi.org/project/pokerkit/>

FEF Selection: TAILWIND

Features like the hover, focus, active, disabled, and group-hover psuedo-classes make Tailwind intuitive for us to use. Tailwind also makes components easy to use with @apply.

Site Map (front end):

