BOOTS: SOPHIA C, EMAAN A, JUN JIE L
DESIGN DOCUMENT

TARGET SHIP DATE: 2025-12-22

APP OVERVIEW:

Home Sweet Home is a murder mystery game where the goal is to guess who the murderer is. The website is designed like a house, with interactive features and clues posted throughout each room for the user to explore. To enter the house, the user needs to register and log in. At the end of the exploration, you will need to figure out the murderer to win. Found clues can be tracked. After all clues were explored, a question will appear that will ask for who the murderer is in the secret room.

TASKS/ASSIGNMENTS:

Sophia C. [PM & Frontend Developer]:
- maintain devlog
- create html templates using jinja:
    - startpage.html (home)
    - login.html
    - finalAccusation.html
    - register.html
    - map.html
    - bedroom.html
    - kitchen.html
    - livingRoom.html
    - frontlawn.html
    - settings.html
- write all css
- ensure navigation links work between pages

Emaan A. [Middleware]:
- initialize flask app
    - configure sessions
    - handle all APIs
- user registration/login/logout
- Set up routes for everything
- requirments.txt

Jun Jie L. [Backend]:
- database organization (tables)
- write SQLite queries for authorization/login/logout

APIS:

411_on_FreeRecipeAPI.md

411_on_JokeAPI.md

411_on_MoviesAPI.md

411_on_HolidaysAPI.md

411_on_TheCatAPI.md

411_on_The_Metropolitan_Museum_of_Art_Collection_API.md
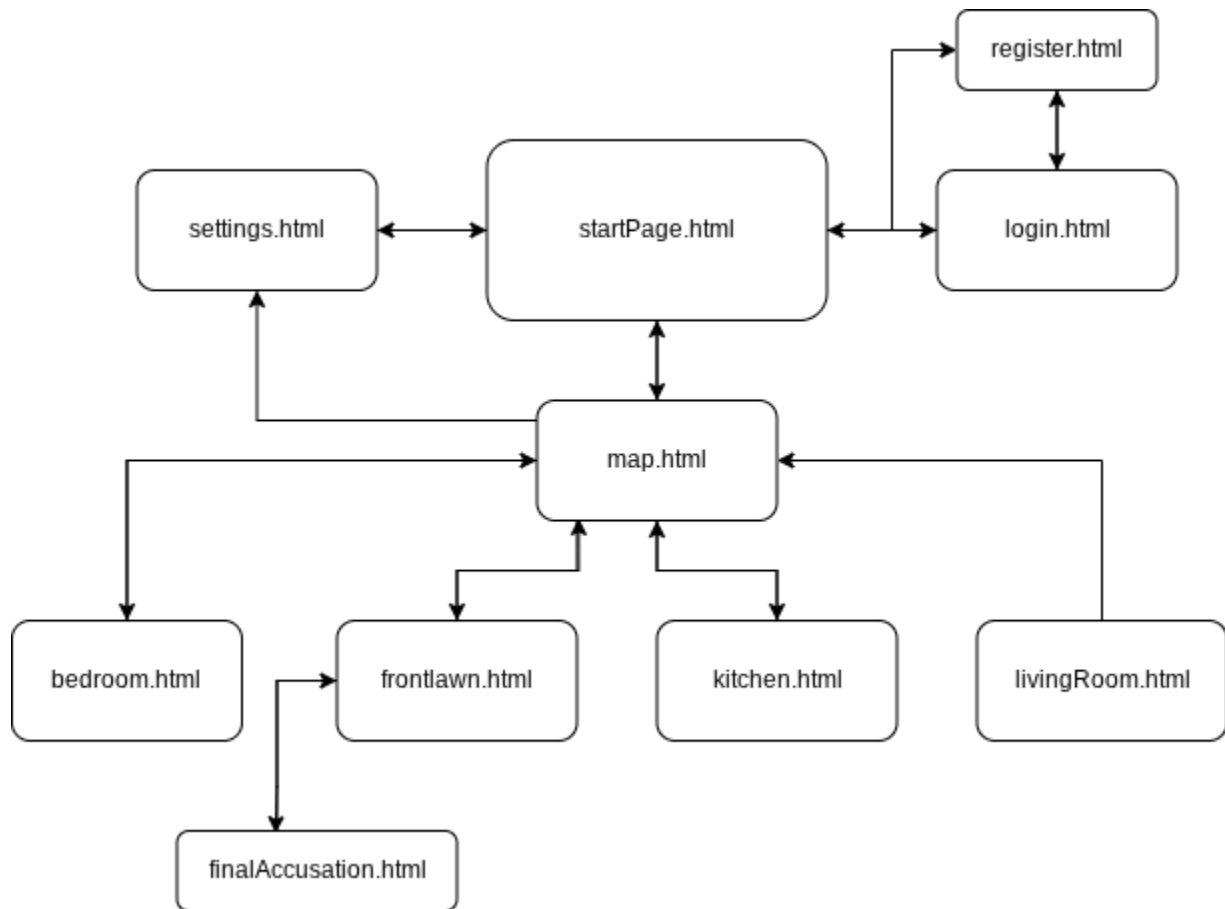
FRONTEND FRAMEWORK:

We are using the Foundations Framebook for our Frontend. The features that will be included are buttons, Icons, Modals, scrollbars, and general containers, such as grids. We used Foundations because its features suited our styling the best, chosen mainly for the reveal/model feature.

DATABASE ORGANIZATION (TABLES):

USER		
TEXT	user_id	PK
TEXT	password	
TEXT	sessionKey	

CLUES		
TEXT	username	PK
TEXT	clueID	PK

SITE MAP

register.html

settings.html ⟷ startPage.html ⟷ login.html

map.html

bedroom.html frontlawn.html kitchen.html livingRoom.html

finalAccusation.html

COMPONENT MAP:
__init__.py: configures flask app, makes routes, handles authorization/login, builds
database, handles all api calls
frontend framework/css styles html templates
jinja2 reads database and turns data to webpage
data.db: SQLite3 database

```
         ┌─────────────┐                    ┌─────────────┐
  ┌──────│  __init__.py │◄──────────────────►│    APIs     │
  │      └─────────────┘                    └─────────────┘
  │             │
  │          renders           ┌──────
  │             │
  │             ▼
┌─────────┐  ┌─────────┐  provides   ┌──────────────────┐
│ data.db │─►│  jinja2 │─ data to ──►│  HTML templates  │
└─────────┘  └─────────┘             └──────────────────┘
                  ▲                         ▲
                  │                      styles
                  ▼                     │
            ┌──────────────────┐
            │ Foundations (FEF) │
            └──────────────────┘
```