

hi's Design Doc for *P02 – Makers Makin' It, Act I*

↳ Roster: Carrie Ko (PM), Christine Chen, Cindy Liu, Joyce Lin

PROJECT NAME: A Disastrous Campaign

TARGET SHIP DATE: 2026-1-16

A Disastrous Campaign (ADC) is a single-player, turn-based RPG web application based on the game AAC. Players progress through encounters and battles to grow stronger, ultimately aiming to defeat the final boss. Our project uses Bootstrap for minor design components.

Program Components + Explanation

sqlite3 (backend data storage system)

- Contains our different databases including:
 - PLAYER
 - ENEMIES
 - CLASSES
 - ENCOUNTERS
 - Comprises all interactions (combat + non-combat)
 - ITEMS
 - ATTACKS

Python (application layer)

- combat.py
 - Contains functions pertaining to the combat system (such as calculations of damage, status effects, etc.)
 - Contains variables such as arrays holding the current alive enemies (*function that generates it through pulling from ENEMIES database*)

Flask (web server/delivery framework)

- __init__.py
 - Sends information to the front end as well as process information sent back (GET/POST)
 - Utilizes functions from additional python files to process the game

- Will manage turns in combat as well as general game progression
- Also contains functions pertaining to database interactions that will be needed
 - I.e. creation of a database, fetching of certain data from different databases, etc.

HTML (frontend display):

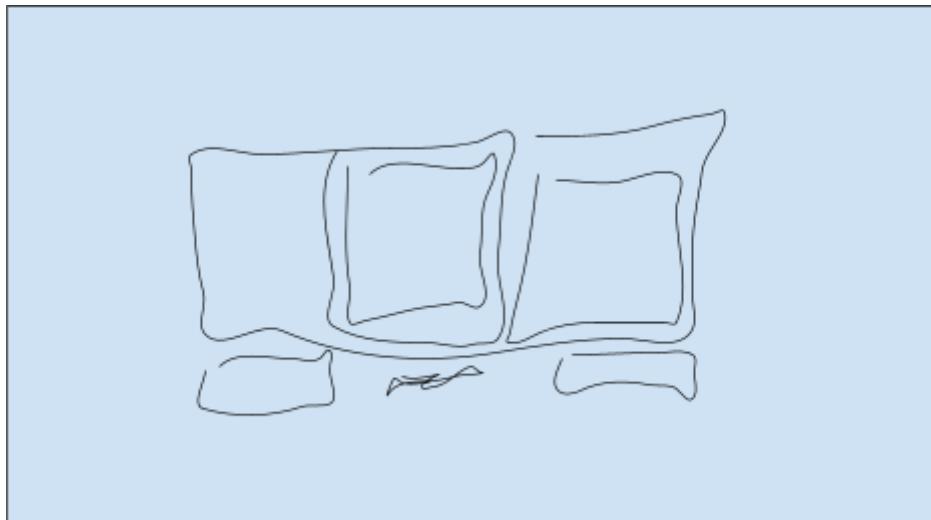
- menu.html
 - PLAY button
 - Begins the game → to first generated battle
- campfire.html
 - Options to:
 - inventory modal (allow users to equip or use items),
 - When opening the inventory modal, a separate modal on the right will hold the player's current stats and allow them to invest stat points obtained from leveling up
 - Include a sell button as well as displays current amount of gold
 - As well as maybeeeeeee an image of your character (equipping what it currently has on)
 - or begin the next encounter.

Each option will be denoted with a button

 - Displays current player hp, level, xp, as well as the current turn
- battle.html
 - Left HUD displays:
 - Current turn (how many rounds of attacks have gone through)
 - Player, enemies
 - HP bar, initiative (determines the order in which each combatant goes), name of player or type of mob
 - All combatants are ordered in terms of highest to lowest initiative
 - Slot is highlighted (or just a different color) when it is that combatant's turn

- Bottom, center UI:
 - Holds player name, health bar, and energy bar
 - Contains four buttons that changes the UI:
 - Attack
 - Makes the UI display all usable attacks as well as their cooldowns and energy costs (formatted with a flex grid)
 - Item
 - Makes the UI display all items that the player is currently holding
 - Displays empty boxes if player is holding none
 - Possible addition: player can select what items to bring to each encounter in campsite but there is a limit on how many types of items you can bring
 - Allows the player to select an item and use immediately
 - Guard
 - Skips player's turn but reduces incoming damage for one round
 - Focus
 - Skips player's turn but increases the energy by one extra point (at the end of each turn, all combatants gains one energy point)
 - Displays the player on the left and all enemies on the right
 - Above each combatant is a health bar and their name (name is optional),
 - Displays the current turn in the upper right corner
- encounters.html
 - Contains three possible actions for the player to take (each their own square and is a button)
 - Three random unique encounters
 - Each one will denote the name of the place as well as a short description
 - An image will be displayed of the place (cropped version of the encounter background)

- Displays the current turn in the upper right corner
- One scavenge (consumes the turn + skips battle)
- One rest (consumes the turn + skips battle)
- Supposed format (***for Joyce Lin's eyes only ty (we saw ur comments pls have mercy)***):

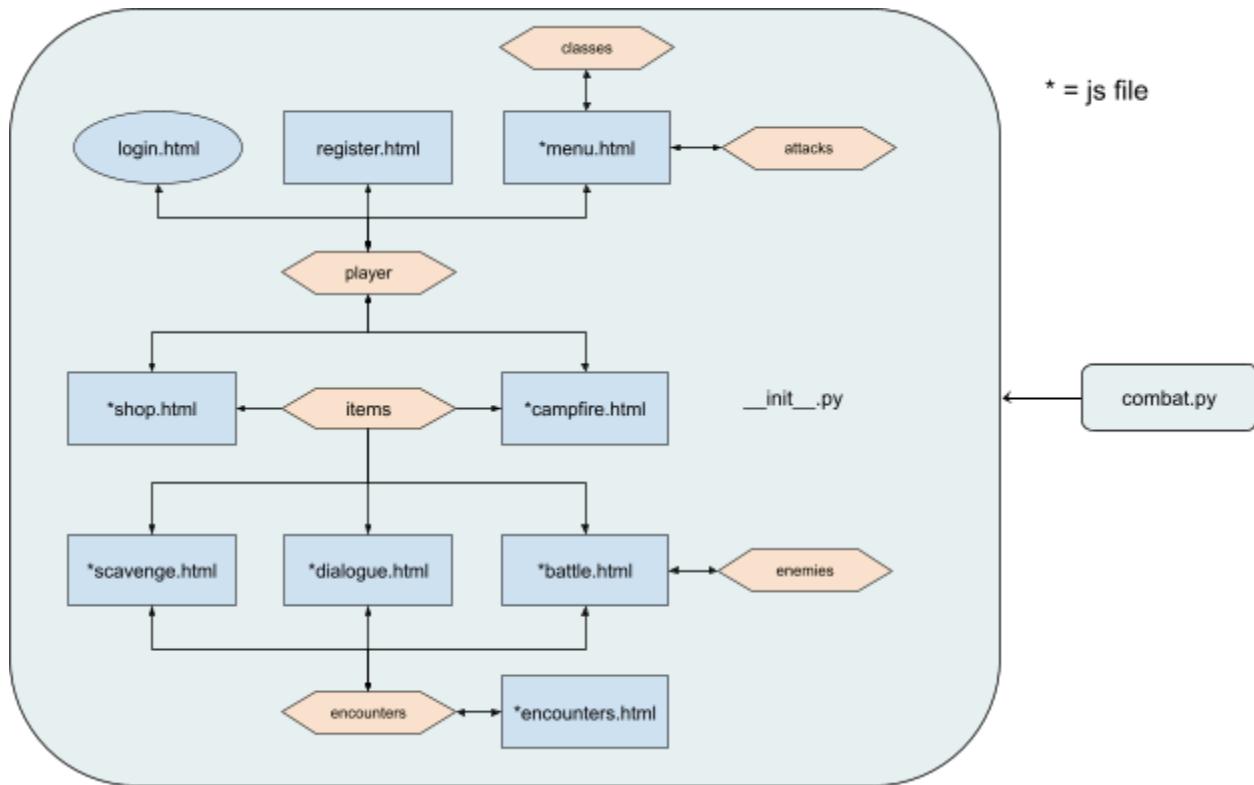


- shop.html
 - is a shop where you buy consumables and equipment (as well as random collectibles that hold no real value) with gold
- dialogue.html
 - Occurs after encounter selection
 - Will play the dialogue pertaining to the selected encounter before displaying the actions that the player can take
 - Actions will be in the form of dialogue options or simple action descriptions
 - The action the player chooses to take will either continue the dialogue or send the player to a battle (battle options are not always available depending on the type of encounter selected)
- scavenge.html
 - A point and click system
 - The player can choose which spot to interact with, and depending on the chosen spot, a certain item will be awarded to the player
 - Possible addition: a popup asking for the player to confirm their selection as well as describing the spot

Javascript:

- Works to facilitate communication between frontend and backend
 - Sending information back to `__init__.py` to be handled and receiving information back to update the page accordingly (hp of an enemy decreasing with an attack)
- Allows for user interactions in each page (such as clicking the inventory icon and opening the corresponding modal)
- Present in basically every html file save for login and register

Component Map



Database Organization

PLAYER

TEXT	username	PK NOT NULL
TEXT	password	NOT NULL

INTEGER	level	NOT NULL
INTEGER	HP	NOT NULL
TEXT	attacks	NOT NULL
TEXT	buff	
INTEGER	bMultiplier	
TEXT	debuff	
INTEGER	dMultiplier	
INTEGER	str	NOT NULL
INTEGER	dex	NOT NULL
INTEGER	con	NOT NULL
INTEGER	int	NOT NULL
INTEGER	fth	NOT NULL
INTEGER	lck	NOT NULL
TEXT	helmet	
TEXT	chestplate	
TEXT	pants	
TEXT	boots	
TEXT	weapon	NOT NULL
TEXT	accessory1	
TEXT	accessory2	
TEXT	accessory3	

* str, dex, con, int, fth, lck = strength, dexterity, constitution, intelligence, faith, luck

* buff and debuffs are they types of statuses they have on them (attack buff, defense debuff, etc) and the respective multipliers they'll have

* attacks are going to be a string of all the attacks they can do ([attack1],[attack2],[attack3]...)

ENEMIES

TEXT	species	PK NOT NULL
TEXT	attacks	NOT NULL
INTEGER	HP	NOT NULL
TEXT	weakness	
INTEGER	wMultiplier	
TEXT	res	
INTEGER	resMultiplier	
TEXT	drop	NOT NULL

* res = resistance

* drop = what the enemy drops upon death

* wMultiplier = damage multiplier for the type they're weak against

* resMultiplier = damage multiplier for the type they resist

ENCOUNTERS

TEXT	type	PK NOT NULL
TEXT	dialogue	NOT NULL
TEXT	background	NOT NULL
TEXT	description	NOT NULL
INTEGER	difficulty	NOT NULL

ITEMS

TEXT	type	PK NOT NULL
TEXT	image	
TEXT	scale	
INTEGER	str	
INTEGER	dex	

INTEGER	con	
INTEGER	int	
INTEGER	fth	
INTEGER	lck	

* str, dex, con, int, fth, lck = strength, dexterity, constitution, intelligence, faith, luck

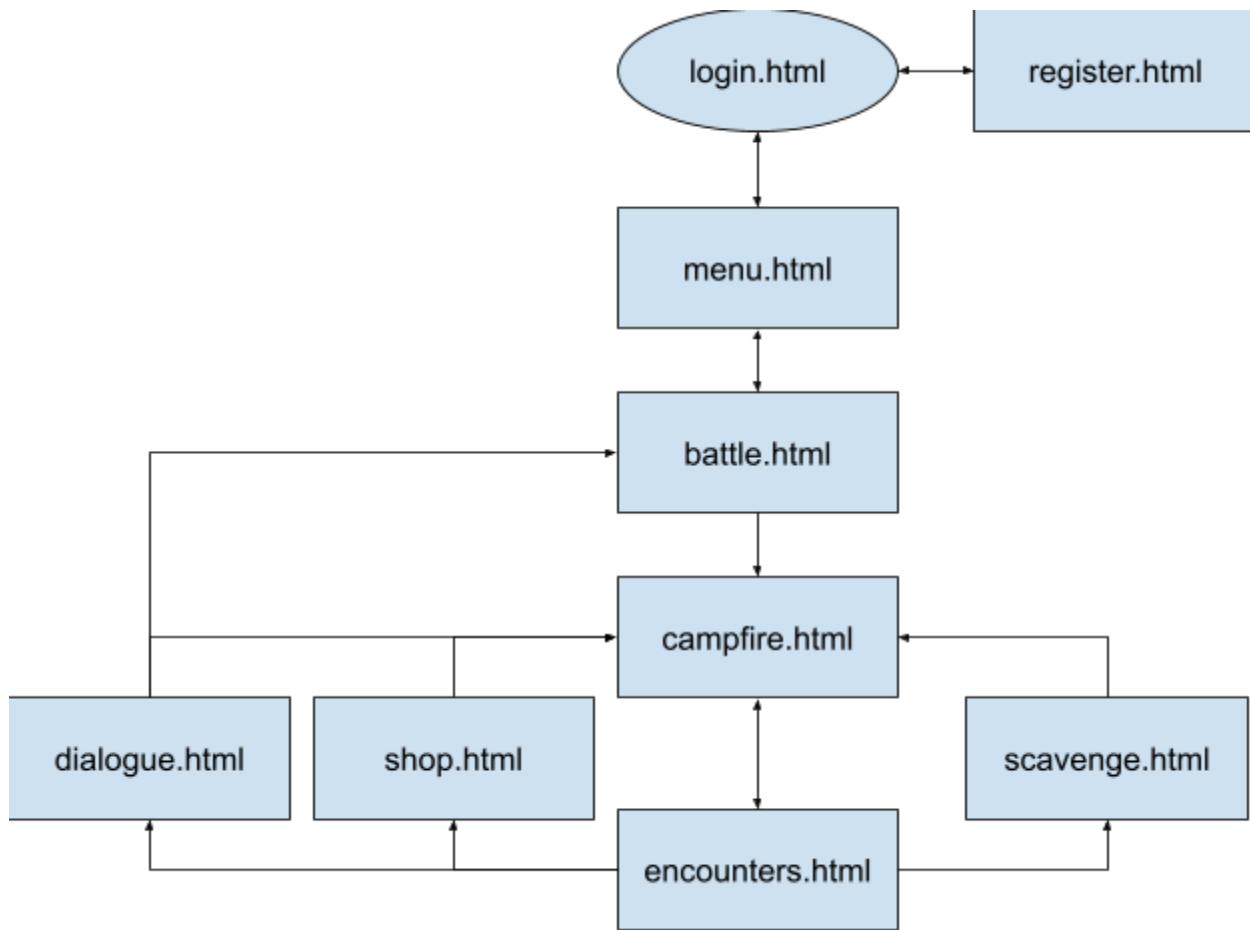
ATTACKS

INTEGER	name	PK NOT NULL
INTEGER	level	NOT NULL
INTEGER	energy	NOT NULL
INTEGER	cd	NOT NULL
TEXT	scale	NOT NULL
INTEGER	baseDamage	
TEXT	effect	

* effect = the type of buff/debuff it gives

* scale = the stat the attack scales on

Site Map (front end)



Member Tasks/Roles

TASK	DEVO	EXPECTED COMPLETION DATE
Sqlite (Database interactions)	Carrie, Christine	1/16/2026
FEF (Graphics + HTML Composition)	Carrie, Joyce, Christine	1/16/2026
Flask	Everyone	1/16/2026
Combat System	Everyone (Primarily Cindy)	1/16/2026

SFX/Music System	Joyce	1/16/2026
Animations (attack indicators)	Anyone who has time	1/16/2026
Dialogue System	Anyone who has time	1/16/2026