Project Participants:

Katie Westcott

Title:

D&D Dungeon Master Tool

Executive Summary:

I want to create a D&D management service that can track players of a campaign, monsters of a campaign, and do the legwork of game “administrative” actions such as de-conflicting initiative rolls in an encounter for me.

Initial Features:

- Create a new Campaign
    - Endpoint: (POST) /campaigns/create
        - BODY: CreateCampaignRequest object
        - Returns: campaignId
- Get Campaign by id
    - Endpoint: (GET) /campaigns/{campaignId}
        - {campaignId}: Campaign id
        - Returns: Campaign object
- Update Campaign by id
    - Endpoint: (PATCH) /campaigns/{campaignId}
        - {campaignId}: Campaign id
- Delete Campaign by id (and associated Encounters)
    - Endpoint: (DELETE) /campaigns/{campaignId}
        - {campaignId}: Campaign id

- Create a new Encounter
    - Endpoint: (POST) /encounters/create
        - BODY: CreateEncounterRequest object
        - Returns: encounterId
- Get Encounter by id
    - Endpoint: (GET) /encounters/{encounterId}
        - {encounterId}: Encounter id
        - Returns: Encounter object
- Add PlayerCharacter to Encounter by ids
    - Endpoint: (PUT) /encounters/{encounterId}/playerCharacters/{playerCharacterId}
        - {encounterId}: Encounter id
        - {playerCharacterId}: Id of the PlayerCharacter to add to the Encounter

- Remove PlayerCharacter from Encounter by ids
    - Endpoint: (DELETE) /encounters/{encounterId}/playerCharacters/{playerCharacterId}
        - {encounterId}: Encounter id
        - {playerCharacterId}: Id of the PlayerCharacter to remove from the Encounter
- Delete Encounter by id
    - Endpoint: (DELETE) /encounters/{encounterId}
        - {encounterId}: Encounter Id

- Create a new Player
    - Endpoint: (POST) /players/create
        - BODY: CreatePlayerRequest object
        - Returns: playerId
- Create a new PlayerCharacter
    - Endpoint: (POST) /players/{playerId}/playerCharacters/create
        - {playerId}: Id of the Player to create the new PlayerCharacter for
        - BODY: CreatePlayerCharacterRequest object
        - Returns: playerCharacterId
- Get Player by id
    - Endpoint: (GET) /players/{playerId}
        - {playerId}: Player id
        - Returns: Player object
- Get PlayerCharacter by id
    - Endpoint: (GET) /players/{playerId}/playerCharacters/{playerCharacterId}
        - {playerId}: Player id
        - {playerCharacterId}: PlayerCharacter id
        - Returns: PlayerCharacter object
- Update Player by id
    - Endpoint: (PATCH) /players/{playerId}
        - {playerId}: Player id
        - BODY: UpdatePlayerRequest object
- Update PlayerCharacter by id
    - Endpoint: (PATCH) /players/{playerId}/playerCharacters/{playerCharacterId}
        - {playerId}: Player id
        - {playerCharacterId}: PlayerCharacter id
        - BODY: UpdatePlayerCharacterRequest object
- Delete Player by id
    - Endpoint: (DELETE) /players/{playerId}
        - {playerId}: Player id
- Delete PlayerCharacter by id
    - Endpoint: (DELETE) /players/{playerId}/playerCharacters/{playerCharacterId}
        - {playerId}: Player id
        - {playerCharacterId}: PlayerCharacter id

Stretch Goals (to be completed if time allows, or after graduation):

- Add the ability to have a frontend interact with the backend and allow for user input
- Expand encounters to include battles to track and calculate damages based on user input

One-Person Project Requirements:

- Database design which contains at least 3 entities and 3 tables
    - Tables:
        - Campaigns
        - Encounters
        - Players
        - PlayerCharacters
        - Encounter_PlayerCharacter
    - Entities:
        - Campaign
        - Encounter
        - Player
        - PlayerCharacter
- Contains all CRUD operations (Create, Read, Update & Delete)
- Each entity should have CRUD operations with one entity having all 4 CRUD operations (Create, Read, Update & Delete).
    - Campaigns
        - Create
        - Read
        - Update
        - Delete
    - Encounters
        - Create
        - Read
        - Delete
    - Players
        - Create
        - Read
        - Update
        - Delete
    - PlayerCharacters
        - Create
        - Read
        - Update
        - Delete

- Contains at least 1 one-to-many relationship
    - Campaigns to Encounters
    - Players to PlayerCharacters

- Contains at least 1 many-to-many relationship with one or more CRUD operations on this relationship
    - Encounters to Players