Group 31 Code Connoisseurs D2 report

1. Summary of your software

a. Our software is a web-based creative city-building game that aims to connect users closer to the real world. To obtain inventory objects for the game, the user need to scan objects (ex: cars, buildings, infrastructures, etc.) in the real world with the AR camera in the Spotstich mobile app (Spotstich mobile app is a multi-purpose social app developed by our partner Project: Human City). Our product could potentially revolutionize how users engage their physical environment with virtual games. It will encourage people to explore the local environment and surrounding objects, capture and record them with a vivid 3D model instead of a 2D picture. For real life urban planners, this product might improve their efficiency of initializing city planning and inspire them with creative and effective design. With this interactive game developed, we hope to enhance users' real world exploration and social interaction which could yield significant positive social impact. This also aligns with the mission of our partner, Project: Human City. As a non-profit organization, Project: Human City aims to enhance the lives of city residents on and off the app with various initiatives and projects. We together want to create a difference in people's lives.

b. There isn't any existing software or implementation for the game, so we are starting from scratch. However, the partner has other developers working on features that our game will utilize such as the AR camera, and the rest of the web and mobile app that our project will be integrated into. Our group will only be responsible for the game as well as the integration of the game into the web and mobile apps.

2. How you decide to divide the project and why

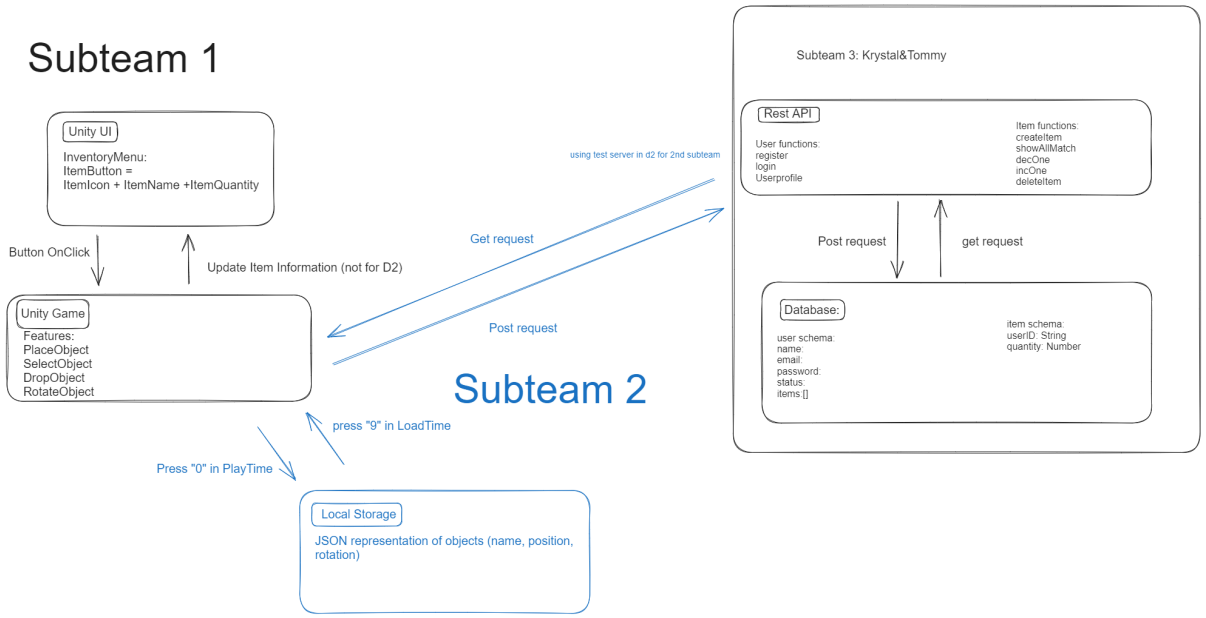
We originally wanted to divide into 2 sub teams of 3 students where one sub team would work on mobile related parts (AR camera, Spotstich mobile app inventory) and the other sub team would focus on developing the web game (Unity city builder game). However, since the AR scanning function and mobile Spotstich app are still in development on the partner side, and the partner is still in the process of recruiting people and organizing teams to work on them, these parts will not be ready for us to work on until a couple weeks later. So the partner and our team both decided that we would focus on developing the unity game for now (D2).

Given this circumstance, we decided to split into 3 sub teams of 2 people:

1. Unity UI and game logic: place objects from inventory to game map, delete objects from game map
2. Connection from Unity game to the server: saving and loading game map state to json file and send/receive to server, retrieve inventory information from server
3. Connection from the server to the database: receive game map/inventory information and make corresponding changes in the database

Since the front-end and back-end of Unity are closely related and are hard to divide into 2 teams to work on separately, we decided to have one subteam focus on Unity game development: game design, making/manipulating the game assets in the unity editor and implementing game functionalities. To save game data and information, we need to have a well-designed database. To this end, we split one subteam to specifically focus on the database. To connect the Unity game and database, we assign the remaining subteam to serialize data and communicate with the server, working with saving, loading, map generation etc.

The following diagram shows our final decision of subteam splits and also how each subteam connects to each other.



3. What each sub-team is responsible for

Sub team 1 (Ricky and Sirui(Ariel)): Responsible for developing user facing game elements such as inventory UI and game map editing in Unity. For D2, we focus on the following user story: *As someone who wants to design a city/town layout, I want to be able to select an item from my inventory and place it at a desired position on the map in the city builder game, in order to carry out my city plans and create a simulation of the city I want to design.* To accomplish this user story, we carried out tasks including working with the partner to design game mechanics, making 3d models in Magica Voxel, writing the game logic for features like selecting and adding game objects to the map and designing and implementing functional inventory UI.

Sub team 2 (Ruiting and Yahya): Responsible for transferring unity game map data (information about objects placed on the game map, including game tiles) to the server. Will be responsible for creating serializable data structures to store game object data and send the serialized data to the server, and receive data from server and convert it back to game object. This is done to satisfy the user story: *I, a user that designs their city/town want to have those changes stay for the next time I log on to the app.* Also implemented a local saving system that saves/loads game object data to the local machine as backup to satisfy this user story. Since the server is not yet connected to the database sub team in this deliverable, we utilize the local saving system to store the actual game object information on the machine. This allows for a more convenient testing of a longer sequence of load and save functions.

Sub team 3 (Tommy and Krystal): Our sub team is responsible for handling the backend development of the game using Node.js. For this particular deliverable, our focus has been on implementing crucial functionalities, including user registration and login, as well as features for creating, deleting, and managing inventory quantities. The core user story we aim to fulfill with this deliverable is as follows: "Users should have the capability to place or remove objects from the game map using items stored in their inventory. To facilitate this, we've provided the means to alter the quantity of inventory items, making it possible to increase or decrease these quantities, with user-specific identifiers provided from client side.