CSCI 0320 Final Project Outline

**Github Repository:** <https://github.com/cs032-2020/term-project-aboden-hlucco-jzhang30-nahmed12-1/blob/master/README.md>

**Group Members:**

Nasheath Ahmed (15/16)

* Strengths: Databases, Data Structures
* Weaknesses: GUI

Andrew Boden (17/18)

* Strengths: Data structures, algorithms, spikeball
* Weaknesses: Databases, GUI

Henry Lucco (17/18)

* Strengths: GUI, Artwork, JavaScript/TypeScript
* Weaknesses: Optimization, Performance

Jake Zhang (Transferred)

* Strengths: Backend, duolingo
* Weaknesses: Artwork, Databases

**Project Ideas:**

**Project 1: Farming Simulator (With Friends)**

There is a finite two dimensional space and every player can pick a plot of land in this space and grow crops for profit. After a certain amount of time, you can harvest the crops and make a profit (Think Stardew Valley). However, there is a catch: if your friend(s) gets to your crops before you, they can steal a portion of your crops and profit. With the money you can make, you can buy more things that increase profit (such as upgrading your farm to not only sell crops but also handle livestock), cosmetic upgrades, or more fun cosmetic items. After development is finished we can add a virtually infinite amount of crops and animals.

Features:

* A simple (or complex if we have time) farming system where users can raise their crops from planting to harvesting and raise animals for greater profit
  + This one is pretty self explanatory, players plant crops and water them, over time the crops grow, the amount of time (real life time) based on the soil and fertilizer given. When the crops have grown they can be harvested and sold for profit. Higher level crops generate more profit.
  + Players do not have to stay online for their crops to grow.
  + A bonus feature here could be email notifications when a player’s crops are ready for harvest.
* The ability to steal from friends, with restrictions: e.g. cooldown after each steal, have to leave a minimum amount for the owner, etc.
  + Players can go through a list of all other players in their world and choose a player to steal from. Within the stealing system players can buy different upgrades or powerups in order to make stealing easier or harder. After a player has been raided their money will have been reduced to the minimum and parts of their farm may have been destroyed. Players will also be able to purchase upgrades such as a coyo that will help them defend their farm against theft.
* Some reward system for users to spend their hard earned money on (maybe in-game real estate or even real world charities if we can find partners)
* Account system, duh.
* Research system
  + Players can “accidentally” stumble upon or intentionally look for newer technologies that can greatly enhance their farming experience. For example, someone who has grown a lot of tomatoes might be able to find a new variation of their current tomatoes that yield higher profit by cross breeding.
  + Players can complete/speed up their research by simply staying online, interacting with others, or playing minigames.
  + Research progress can also be (partially) stolen. And this is what we call a big grief.
* Economy system
  + Players can have the options to trade with each other and the price of each type or product may fluctuate
  + There is only a finite amount of land and players have the option to expand their farm if they have enough money, which means the price of land will also become an interesting factor
  + In addition, we want to set points of interest on the map (river where players can get higher quality water and etc.), so the location of a player’s farm matters too, which means players can compete for land/locations that they desire.
* Friend system. This feature is critical since the focus of this game is to bring friends together and allow people to interact with each other as much/as little as they want

- Friend network to discover more friends who play this game

- Leave an anonymous note for your friend after stealing

- Help your friends build their farm, water their plants, feed their animals, etc.

* Good graphics and animations cuz it's no longer the late 2000s

Potential Challenges:

Creating one unified instance in which all players data is saved with be the biggest challenge of this project. Especially as the number of players grow, the server must be able to expand and hold all of these players in the same world. Thus the worldspace in which the farms exist must be able to handle players deleting their accounts, players adding new accounts, players taking over other players farms, and other player related features that we may implement. In addition to this, we must find a way to solve the problem that in order for people to play the server must always be running thus we will need to find a way to host the server somewhere that is not one of our personal laptops. Finally, we must implement an account system that keeps track of each player and their data and so we will need a database that contains each player's login information and the details of their account and what they have in the game. This will be difficult as none of us are particularly skilled at writing databases but we are excited to learn and believe that we are up to the challenge of creating this database. There will also be some challenges with player interaction for example, we need to be able to facilitate updating in real time for example if a player is playing and gets raided they need to see that it is happening or be able to catch it and have a way to stop it.

Concept art (copied from the original):



**Project 2: Calendar (With Friends)**

It's hard keeping track of a busy schedule. Every time a friend wants to hang out, it's hard finding a spot that fits your time slots. In addition I always have to move things around because something will come up. I'm proposing a user interface which will make finding time and fitting everything in your schedule a breeze.

Core Functionality:

- Ability to create calendar events with a priority field

- App will propose most convenient times for given event

- Can be given the option of having a "hard" time slot, can't be moved, or can occur at a range of times

- Will move other events around based on priority

Complexities:

The complexity of this project lies within implementing an algorithm that maximizes utility of a busy schedule. For example, if three users are trying to find a time to hang out in the coming week, the app will suggest times that work best for all three users. It will be difficult engineering a way to map multiple events to finite time frame in a way that doesn’t overlap. This biggest challenge will be managing different accounts. There will need to be a large central database.

The project will require a lot of querying. When a user inputs a time frame for a new event. The database will search the users calendar for open time lots that match the desired time. In the case where multiple people are planning an event, the results for the search of each user will need to be cross referenced so only the times that work for all users will be suggested.