### Development Modules:

* **Occupational Classes (Active: Damian)**
  + ~~GUI screen for viewing occupational values~~
  + ~~Performing different tasks (fighting, mining, farming, building, etc) provides you with experience towards those tasks, with a simple level-up system for spending a lot of time specializing in specific areas. High levels in different occupations give you small perks and abilities that improve your efficiency in those areas.~~
  + Make each class perk a tree of abilities, one branch focusing on passive effects, one focusing on aggressive effects.
  + ~~Base Class UI~~
  + ~~Smaller set of classes:~~

1. ~~Resource Collecting: Mining, Lumber UI~~
   1. Tree Stats
2. ~~Foods: Farming, Fishing, Cooking UI~~
   1. Tree Stats
3. ~~Building: Architect, Transport, Engineer UI~~
   1. Tree Stats
4. ~~Wizardry: Potions, Enchanting UI~~
   1. Tree Stats
5. ~~Warrior: Fighting UI~~
   1. Tree Stats

* **Data Analysis/Mining**: **(Active: Li & Yichen, Nicole)**
  + 5-Factor actions
    - We determined specific gameplay actions that would be associated as being positively or negatively correlated with the five personality factors. We need to determine exactly how to quantify these values… in other words, we should have each personality factor be rated with a value between 0 and 1, and so we need a way to map these gameplay actions to a 0-1 value.
    - Figuring out this mapping might be a machine learning challenge in itself...
  + Do text classification on the chat logs.
    - We can use the Allport and Odbert 5-factor dataset as pre-labeled data for training & cross-validation, and then use Naive Bayes to classify the chat messages.
    - (Do we want to use bag-of-words, or a vector space model?)
    - For better accuracy, use stemming, and possibly try to recognize and correct misspellings and typos.
* **Data Logging & Visualization (Active: Damian)**
  + Need to create Core Mod using ASM to create Custom Events
  + Logging player events (partially done, need to update with our extended list of logged data, as well as the 5-factor personality stuff)
    - ~~Chat~~
    - ~~Player Position~~
    - ~~Consume~~
    - ~~Item Used~~
    - ~~Item Smelted~~
    - ~~Item Crafted~~
    - ~~Anvil Used~~
    - ~~Block Mined~~
    - ~~Block Placed~~
    - ~~LivingEntity Killed~~
    - ~~Login~~
    - ~~Logout~~
    - ~~Achievement~~
    - Territory
    - Groups
    - Trading
    - Class Occupation
    - Potion Brewing
    - Item Enchanted
    - Animal Tamed
    - Portal
  + ~~Setting up database.~~
  + ~~Method for sending logged data from the game to the database.~~
  + ~~Visualizations Local~~
  + Visualization Web
    - [D3](https://d3js.org/)
    - [NPM](https://www.npmjs.com/)

* **GUI & User Interface (Active: Damian)**
  + A utility plug-in for making it easier to render menus/interfaces to the screen.
  + ~~UI plain background~~
  + Drop-down lists
  + ~~Image Buttons With Hover Text~~
  + TextBox
  + UI Tabs
  + Progress Bar
* **Admin Interface (Active: )**
  + An interface where the admin can see data about the different clans/players around the map, & their calculated personality traits.
  + Also where they can toggle events for different groups.
  + Two tabs to switch between the different screens.
  + Rather than having data being calculated and synced in real time, give the admin a request button that manually calls for an update. (Potentially also allow syncing of only specific information, rather than recalculating everything at once)
* **Player Teams (Active: Andrew)**
  + ~~A method for players to create teams.~~
  + ~~A method for leaders to invite players to their groups, or for players to request to join an existing team, and a way for those invitations to be accepted/rejected.~~
  + ~~A method for players to join/leave leadership positions, and handling of what happens to the team if the last leader leaves the group.~~
  + ~~A method for players to leave groups, and handling of what happens to the team if the last member leaves the group.~~
  + ~~Simple attributes about the teams (name, color, etc)~~
  + ~~Ability to modify those team attributes.~~
  + ~~Ability to view a list of all created teams.~~
  + ~~Ability to view information about a given team.~~
  + Implementation of things that differentiate leaders from normal members (we’ve discussed leaders having different “stats” than regular members, etc)
  + ~~Protection modes that allow/disallow players from one team from being able to attack another team.~~
  + ~~Chat mode that allows you to send chat messages only to your team members.~~
  + Players in teams should only be able to sleep in beds within their territory.
  + Ability to see all team members that are currently online, as well as some statistics (their current HP, current location, etc?)
* **Territory Acquisition (Active: Andrew)**
  + ~~Block protection for areas of land.~~
  + ~~Addition of an item that allows you claim a block of land as your team’s territory~~.
  + ~~Visualize the claimed territory (make the land faintly glow your team’s color, draw a border around the claimed territory, etc?)~~
  + Add a method for teams to steal territory from other teams (have players drop territory items when they’re killed by other players).
  + Potentially stationary machines that can be plopped down in a spot and very slowly claim/revert territory within a radius of them.
* **Finite Worlds (Active: Andrew)**
  + ~~Finite world w/ customizable world size~~
  + ~~Clustered ore generation~~
  + ~~Configurable biome generation.~~
  + Extension that allows you to link spawning of certain monsters/animals/etc to specific sections of the world.
  + Extension that makes other dimensions (ie: the nether) finite in addition to just the overworld.
  + Extension that allows configuring how special structures (villages, strongholds, mineshafts, etc) spawn within the finite world.
* **Trading System (Active: )**
  + Add a method that allows you to initiate trades with other users.
  + Create a custom UI that allows you to specify the items to be traded, and accept/reject the trade.
  + Underlying implementation of a system that allows direct transfer of item(s) from one player’s inventory to another’s.
  + Alternately (or additionally), a global bulletin board system where you can post a trade you want to make, which gets added to a public list, and anyone in the server can access it and potentially accept the trade.
  + This also requires the ability to handle trading with players that may not currently be online -- store the half-completed trade, and process the end result of the trade the next time the poster logs back in.