**Class Design Ideas**

**Server - Marko**

Saves all data for game (format of the data still to be determined, waiting for the Prof)

Any changes to the world that a User makes must be approved by the Server

* globalWorld
  + stores the current state of the game as a World object
  + every “heartbeat” receives updates from the Users and updates the World if that actions made by the Users are valid
* updateWorld()
  + updates the data the local data of each User
  + should only send data that the User currently needs to know (most likely just their current LocationTile)
* requestUpdate(data)
  + called by a User that has made a local change to their world
  + if request is valid then globalWorld is updated, and all Users will have their worlds updated the next time updateWorld() is called
  + if the request is invalid then do nothing

**Time**

Represents “game time”

* heartbeat()
  + every 0.1 seconds (maybe faster, depending on how responsive we want the game to be) informs the Server to perform updateWorld()

**World - Sonny**

Represents the overall world of the game (duh…)

* locations[]
  + stores all LocationTiles that represent that world
* users[]
  + stores all the Users and their Characters currently logged into the world
  + maybe just store Users or Characters?
* startLocation
  + defines where a Character goes when they enter the world for the first time

**Interface - Bryan**

Displays output to the user, and returns input to the main thread.

Runs as it’s own thread

**LocationTile - Ben, Sunny**

Represents a location in the game

* currentWidgets[]
  + list of all widgets currently at that location
* nearbyLocations[]
  + list of all LocationTiles that can be visted by the the LocationTile
* entryMessage
  + message that is displayed to the User when it’s character enters the LocationTile
  + Should probably list all widgets in the currentWidgets List, and any other useful data about the LocationTile

**CharacterProfile - Michael**

Represents the account info associated with each character.

* username
* password
* lastLocation[]
  + stores all worlds the character has visited and the LocationTile in that world when the character was logged off by the User

**Widget - Jason**

Represents any object in the game.

* name
* life
* usage
  + defines what function is called when a character interacts with the widget
* currentLocation
  + current LocationTile the widget is in

**Character** - *Inherits from Widget - Louie*

Represents the User in the game

* inventory[]
  + lists all Widgets currently being carried by the Character