Client

* Graphics
* Ui
* Pathfinding
* Resource management
  + Checks the required stuff before building something
* Fog of war

Server

* Map generation
* Data storage
* Ai
* Start program
  + Load assets
    - During the splash screen
* Main menu
  + - Switch screen to main menu screen
  + Single player
  + Multiplayer
  + Settings
    - Switch screen to settings screen
* New game
  + - Switch screen to new game screen
  + Server: Generate world
    - Generate heightmap for land/ocean
    - Use same height map for elevation or generate a new one to use
      * Land only; Mountains, hills, plains
    - Generate another height map for climate
      * Grass, plains, desert, tundra, snow
    - Randomly generate lakes, then rivers
      * lakes have to be a certain distance away from any ocean/lake
    - generate features
      * forest, jungle, etc
      * diamond square first, then assign features to them based off the altitude
    - generate starting locations
      * gets valid starting point tiles
        + shore starting point, land starting point, etc
      * generate starting points randomly
        + take away tiles that are close to the spawning points
      * if all the starting points don’t get generated, then restart whole generating process
        + need to save the original lists
    - generate wonders
      * somehow
    - generate resources
      * spread them out around wonders, starting locations, then randomly
  + client: Unit management
    - list of all units
      * goes through them each turn to notify player to move them
      * unit:
        + bool hasMoved
        + float movementLeft
        + float maxMovement
        + int hp, maxhp, exp
    - list of cities
      * goes through them each turn, does their actions
      * city:
        + str name
        + ArrayList tilesOwned
        + Float food, money, science, etc
        + Int hp, maxhp, armor/damage
        + Bool hasShot
* Hex
  + Vec2 pos
  + Texture sprite
  + <land data>
    - Str landType
      * Ocean, shore, desert, plains, grassland, etc
    - Str elevation
      * Normal, hill, mountain
    - Str river
      * 000000
      * One for each side
    - Str feature
      * Forest, jungle, etc
    - Str resource
      * Streetgic
        + Iron, horses, etc
      * Luxrey
        + Citrus, cotton, etc
      * Food
        + Cows, foxes, etc
    - Int Count
      * Count of resource (stretgic)
    - Str improvement
      * Name of the improvement on this tile
    - Bool pillageImprove
      * If the improvement is pillaged
    - Str/int ownership
      * Who owns this land
    - Int cost
      * The maintance cost of the tile (per turn)
  + Float movementCost
  + Bool walkable
  + Generates the sprite from the land data