

P02: Makers Makin' It, Act I

Team Name: DeadDuckies

Roster: James Lei (PM), Kiran Soemardjo (Dev 1), Emaan Asif (Dev 2), Sarah Zou (Dev 3)

TARGET SHIP DATE: 2026-01-16

## Description

We will be making a dumbed-down version of Civilization 6 combined with the research and focus systems of Hearts of Iron IV. Games will occur between two players on a pregenerated map with random spawn points.

**Tiles / Improvements:** The map will consist of hexagonal tiles, which have different terrain types. These terrain types determine the yield of a tile, which consists of the core resources: science, gold, food, and production. Tiles may have bonus resources (e.g. wheat, sheep, diamonds) which provide additional yield, or strategic resources, which are used in the production of units and are accumulated each turn once the tile is improved. Improvements cost production, but, when built, they increase the yield of the tile.

**Population / Workers:** Tiles only generate resources for a player if they are worked. The number of tiles that can be worked is determined by the population of the civilization. Population grows each turn based on food production.

**Districts:** Districts can be built over a tile, replacing the preexisting yields. Commercial hubs generate gold, campuses generate science, and industrial zones generate production. City centers generate gold production, and science. Encampments will be covered later. Districts can have multiple improvements on them called buildings. They do not need to be worked. Districts also have adjacency bonuses based on the districts and improvements surrounding them.

**Research / Focuses:** Each player starts with one research slot, which they can use to progress along the technology tree. The technology tree unlocks or improves units, districts, and buildings. Each technology has a fixed cost in science. Focuses add progress to technologies, create new research slots, and spawn units or resources.

**Expansion / War:** Players can purchase new tiles adjacent to their current ones using gold. Encampments are used to spawn in units, which cost population, gold per turn, and production. Units can fight each other, or travel into enemy territory. They can pillage enemy improvements, which removes the improvement and generates yield for the pillaging player.

Things Cut from Base-Game Civilization 6: Amenities, Religion, Culture, Settlers, Great People, Government, Naval Warfare, Post-Industrial Age Content, Natural Wonders, Turn/City-Based Production

## Dependencies

Framework: None

JavaScript: SocketIO (<https://cdn.socket.io/4.8.1/socket.io.min.js>)

Python: Flask, Sqlite3, Flask-SocketIO

APIs: None

## Tasks + Assignments

Project Manager: James Lei

- Set up SocketIO in Flask and JavaScript
- Implement District interface, Building improvements, and Lobby component

Dev1: Kiran Soemardjo

- Create database structure
- Implement Focus and Research components

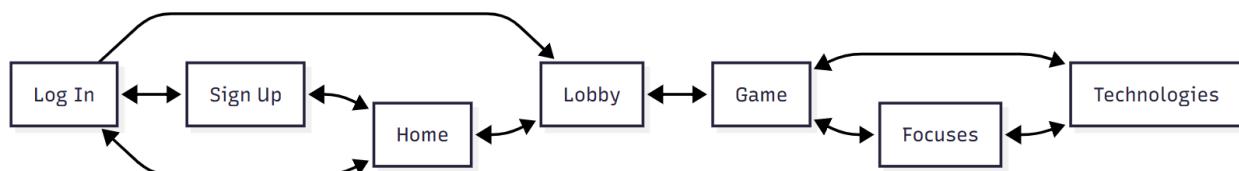
Dev2: Emaan Asif

- Implement Authentication and Unit components

Dev3: Sarah Zou

- Connect frontend to JSON storage
- Implement Map and Construction components

## Site Map



Log In / Sign Up

- Users create accounts and log in

Home

- Displays instructions to play the game

Lobby

- Users create a game and join other users' games

Game

- Displays the game
- Can only exist into lobby

Focus / Technologies

- Displays the focus / technology trees in a game

## External Resources

Civilization 6 Wiki - Resource Icons & General Game Information

([https://civilization.fandom.com/wiki/Civilization\\_VI](https://civilization.fandom.com/wiki/Civilization_VI) )

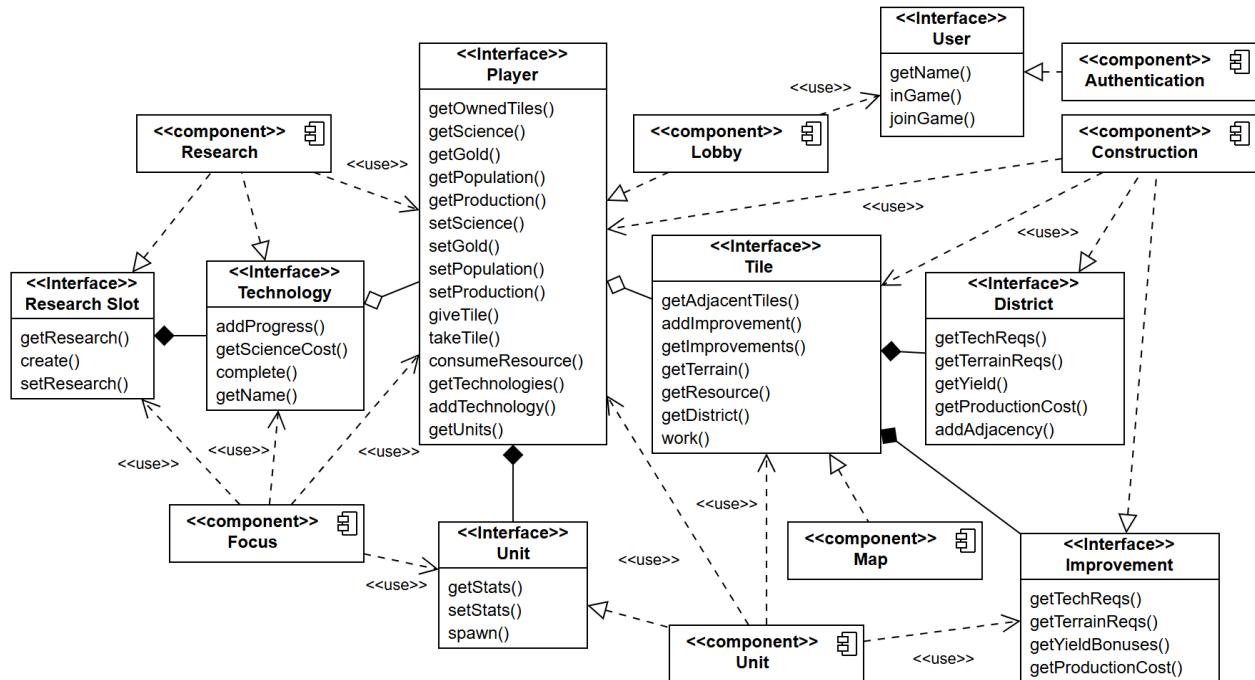
UnTile - Tileset

(<https://github.com/Malwen/UnTile-Civ6-Tileset> )

## Components

Authentication	<ul style="list-style-type: none"> <li>- Create user accounts and logs them in</li> <li>- Implements User interface</li> </ul>
Lobby	<ul style="list-style-type: none"> <li>- Displays available games and let users create/join games</li> <li>- Uses User interface to associate each game with two users</li> <li>- Instantiates Player interface on game load</li> </ul>
Map	<ul style="list-style-type: none"> <li>- Renders the map and the game's UI (i.e. tile information, resources in storage)</li> <li>- Implements the Tile interface</li> </ul>
Construction	<ul style="list-style-type: none"> <li>- Builds new improvements and districts in an empire</li> <li>- Uses Tile interface to check if an improvement can be built</li> <li>- Uses Player interface to consume resources needed for construction</li> <li>- Implements District and Improvement interfaces</li> </ul>
Unit	<ul style="list-style-type: none"> <li>- Spawns units and moves them around the map</li> <li>- Handles population growth and adds workers to tiles</li> <li>- Uses Improvement and Player interfaces to pillage tiles</li> <li>- Uses Tile interface to determine movement cost and travel</li> <li>- Uses Player interface to use resources / check technologies needed to produce units</li> </ul>
Research	<ul style="list-style-type: none"> <li>- Displays the technology tree and researches new technologies</li> <li>- Implements Research Slot and Technology interfaces</li> <li>- Uses the Player interface to associate research with a player and consume science</li> </ul>
Focus	<ul style="list-style-type: none"> <li>- Allows players to complete government programs</li> <li>- Uses Research Slot &amp; Technology interfaces to boost research</li> <li>- Uses Unit &amp; Player interfaces to spawn units, buff existing ones or spawn resources</li> </ul>

## Component Map



## Database Organization

Profiles		
TEXT	username	PK
TEXT	password	

Games		
INTEGER	id	PK AUTOINCREMENT
TEXT	player1	FK REF profiles(username)
TEXT	player2	FK REF profiles(username)

Resources		
INTEGER	game	FK REF games(id)
TEXT	player	FK REF profiles(username)
TEXT	name	
INTEGER	amount_stored	

Tiles		
INTEGER	game	FK REF games(id)
TEXT	owner	FK REF profiles(username)
INTEGER	x_pos	
INTEGER	y_pos	
TEXT	resource	
TEXT	improvement	

Technologies		
INTEGER	game	FK REF games(id)
TEXT	player	FK REF profiles(username)
TEXT	name	

Districts		
INTEGER	game	FK REF games(id)
INTEGER	id	PK AUTOINCREMENT
TEXT	name	
INTEGER	x_pos	
INTEGER	y_pos	

Buildings		
INTEGER	district	FK REF districts(id)
TEXT	name	

Cities		
INTEGER	district	FK REF districts(id)
INTEGER	health	
INTEGER	max_health	

Focuses		
INTEGER	game	FK REF games(id)
TEXT	player	FK REF profiles(username)
TEXT	name	

Units		
INTEGER	game	FK REF games(id)
TEXT	owner	FK REF profiles(username)
INTEGER	x_pos	
INTEGER	y_pos	
TEXT	name	
INTEGER	health	