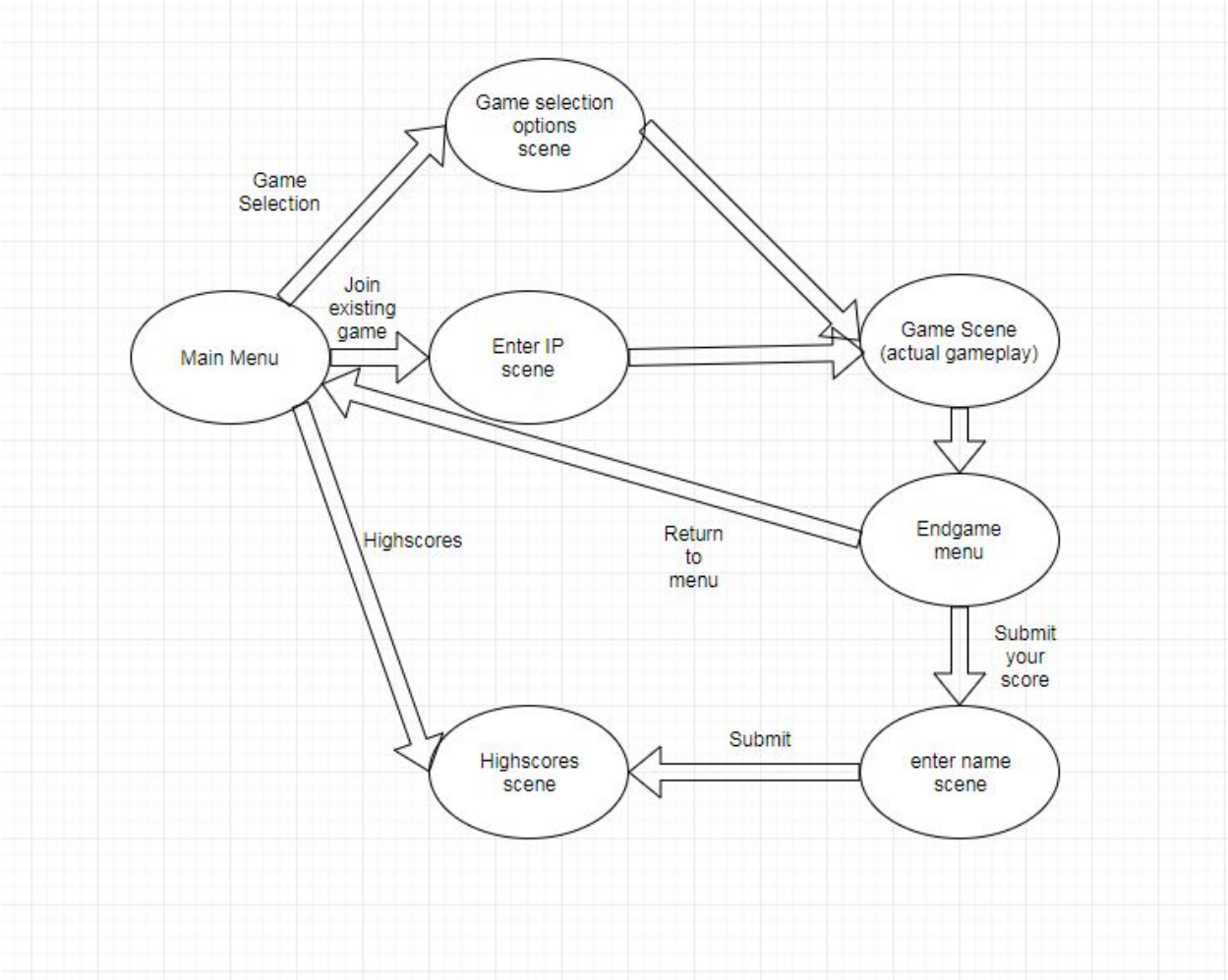
Team 15 Incremental and Regression Testing

Project Name: Crater Clash!

Team : Brendan Raftery, Benjamin Loisch, Mangkorn Yuan, Conner Isaacs, Nada Alnoory

Component Diagram:



Component Explanations:

Main Menu - The menu that the game opens on. There are 3 options: Game selection, Join existing game, and highscores. Game selection is where you can create your own game, join existing game is joining a multiplayer game, and highscores allows you to view highscores.

Game Selection Scene - The menu the user is taken to upon clicking ‘Game Selection’ on the main menu. Here, the user is given options for the game they want to create. They can choose to use a default map or randomly generated one, the map size, and whether they want it to be a hotseat game or online multiplayer.

Enter IP scene - The scene the user is taken to upon clicking the ‘Join Existing Game’ option. When a user creates a multiplayer game, they are given the IP to give to someone else. This can be entered by the other player at this scene, and it will connect them to the game.

Highscores Scene - The scene the user is taken to after either clicking it in the main menu or finishing submitting their own high score. From here, you can browse highscores and see player’s ranks and scores.

Game Scene - The scene that contains the actual game. Here, players battle eachother and spawn units, move units, etc. The scene ends when one player’s base is taken over.

Endgame menu - The scene the user is taken to after the game completes. From here the user can choose to either submit their score or they can return to the main menu.

Enter name scene - If the user chooses to submit their highscore, they are taken to a scene that prompts them for a name. The player can enter their name and then click “submit” to submit their score to the highscores, and then be taken to the highscores scene.

Art Assets - While the art spans through much of the game’s UI and gameplay, testing is still needed to ensure tilesets work properly together and UI fits properly in the screen.

Incremental and Regression Testing:

Severity: 1 = low, 4 = high

Form of Incremental Testing

Since the majority of the implementations were in the main modules and some of the sub modules are yet to be implemented, we thought it would be better to do top-down incremental testing. The gameplay module is tested by combining all the sub-modules in development (scripts, units, resource tiles, prefabs + their properties, etc.) and then testing it as a whole.

Incremental Testing

Module		**Gameplay**	
Defect No.	Description	Severity	How To Correct
1	Unit spawns on base even when another unit is already there	3	Create a check to see if unitTileInstances[] array at player base is null.
2	Default values for unit.cs script that is attached to melee1 and melee2 prefabs not being set.	4	Delete Start() default method in unit.cs class and set all properties found in that class to a value.
3	Health bar not resizing correctly	1	Set pivot point x value to 0 to align health bar to start on left of melee unit sprite.

Module		**Scoring**	
Defect No.	Description	Severity	How To Correct
1	Searches are only matching if the beginning is the same as the query	2	Make the matching function use .Contains() instead of comparing a substring of the

			beginning
2	Searches only showing one response	2	Increment the search array index after every match
3	IndexOutOfBounds exception on multiple searches	3	Reset the searchIndex to 0 after every search so it starts matching from the beginning
4	Submitting a score with empty nickname	1	Use the function IsNullOrWhiteSpace() to check if the nickname is empty or valid

Regression Testing:

Module		**Gameplay**	
Defect No.	Description	Severity	How To Correct
1	Artwork sprites for melee units can cover up entire terrain tile.	1	Still an issue as recorded in 2/8/19 “Bug List CS 408” document. Fix will come as more artwork is generated and clipped down to size.
2	Unit can move to any place on the map.	3	An issue since it was recorded in the Bug List document; add an integer “maxMoveDistance” to attached unit.cs script of melee prefabs and check the click move distance against that for the selected unit

			to fix the problem.

Module		**Scoring**	
Defect No.	Description	Severity	How To Correct
1	Submitting a score with nickname that has white spaces at the start or at the end	1	Use the functions TrimStart() and TrimEnd() to trim white spaces at the start and the end before submitting the score

Module		**Art Assets**	
Defect No.	Description	Severity	How To Correct
1	Tileset made did not allow for proper transitions around interior corners	2	Restructure of tileset layout to a 5x5 set rather than a 3x3 set. This allows for interior corners to be added
2	Moveable tiles tileset was set to be a half and half tileset. Causing moveable tiles to seem out of bounds	1	Tileset changed to have the majority of the tiles be moveable ground with the edges being out of bound textures