Name: Morgan Brockman Mark _____/50

1. Brief introduction __/3

My feature for Tile Hero is the tile class. The tile class will facilitate much of the variability of the game, such as which monsters are in a particular tile, how obstacles might prevent movement in a tile, and where a tile can be entered or exited.

2. Use case diagram with scenario __14

Scenarios

Name: Create Tile Object
Summary: Tiles will be created from the Tile Manager and placed onto the grid through the Grid Manager
Actors: Tile Manager, Grid Manager
Preconditions: Information about the tile is present in the Tile Manager and a grid for the tile to be placed in is present in the Grid Manager
Basic sequence:

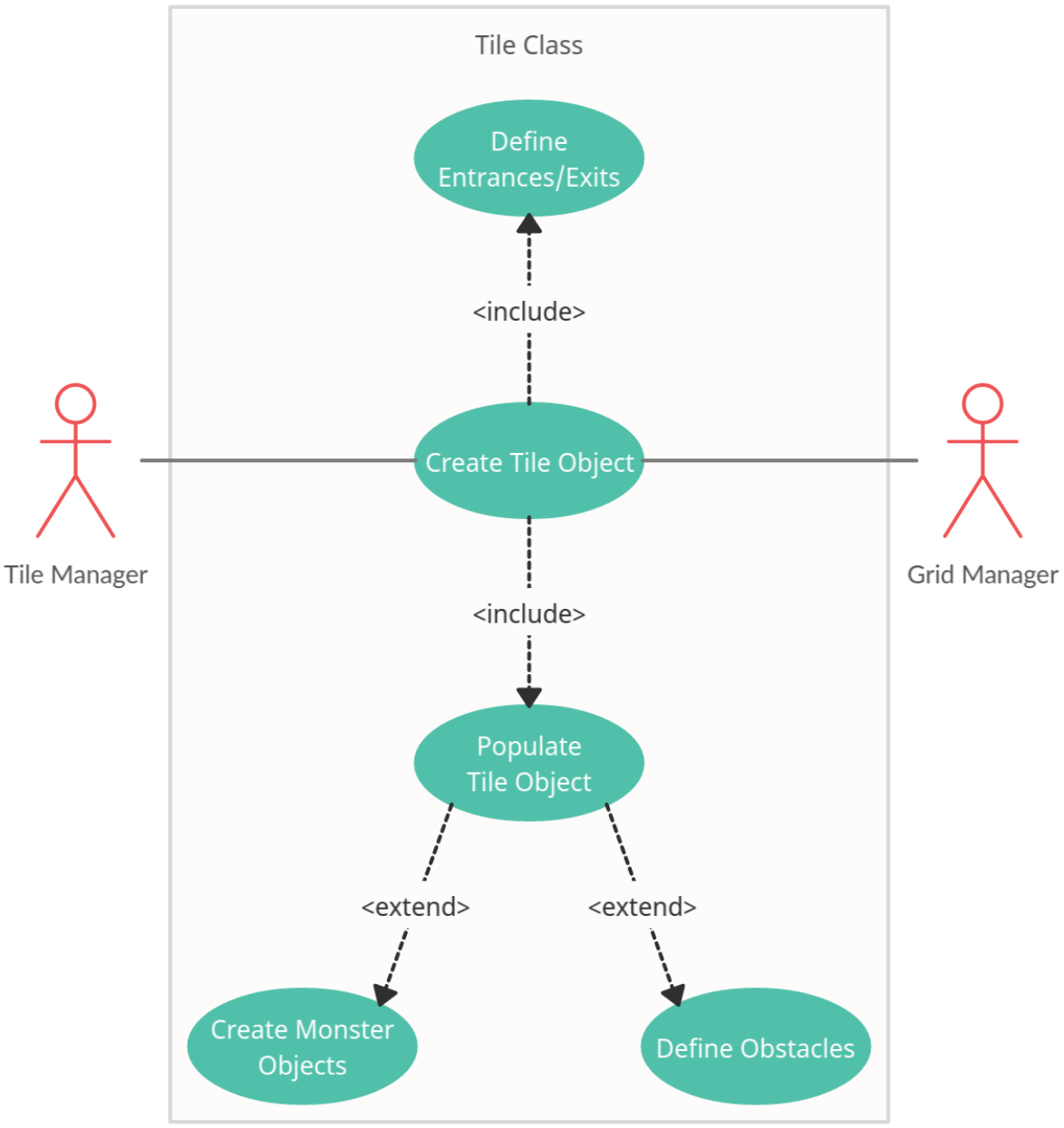
Step 1: The Tile Manager gives information about entrances/exits and tile population to create a tile object.
Step 2: The tile object is populated and entrances/exits are defined.
Step 3: Depending on information, monster objects are created and/or obstacles are defined.

Exceptions:

Step 2: The Tile Manager does not define some points as entrances/exits: entrances/exits are not made.
Step 3: Monsters are not called for by the Tile Manager: no monster objects are created.
Step 3: No obstacles are defined by the Tile Manager: no obstacles are created.

Post conditions: Grid Manager points to the newly created Tile Object
Priority: 1
ID: T01

Use Case Diagram



3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

Process Descriptions

Creation of a tile object is called.

The Tile Inventory, according to what tile type is called, passes that information to x.3, as well as passing information about the types of obstacles in that tile type to x.1 and how many monsters and what types in that tile type to x.2.

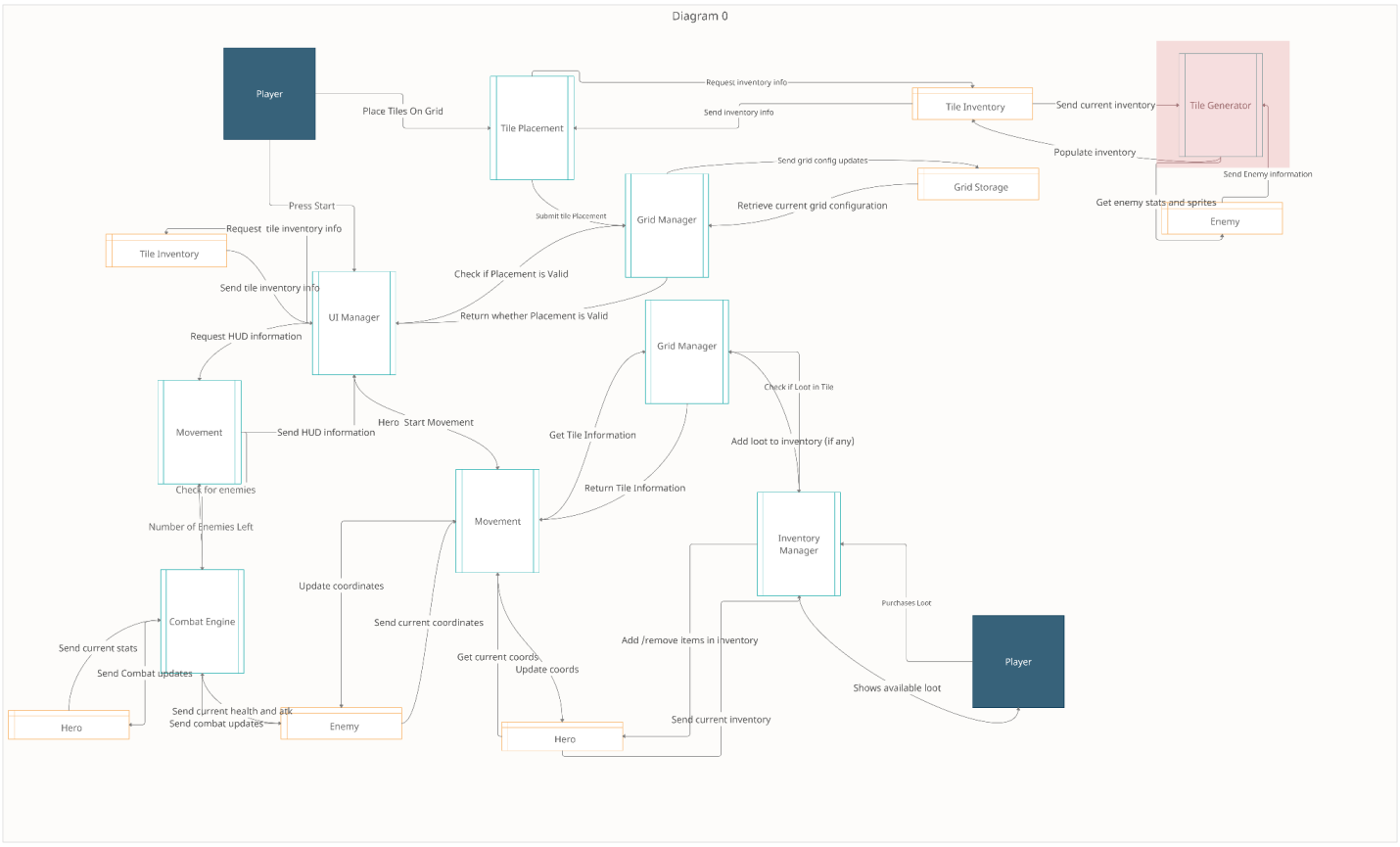
After defining the obstacle information, x.1 passes the information to x.3.

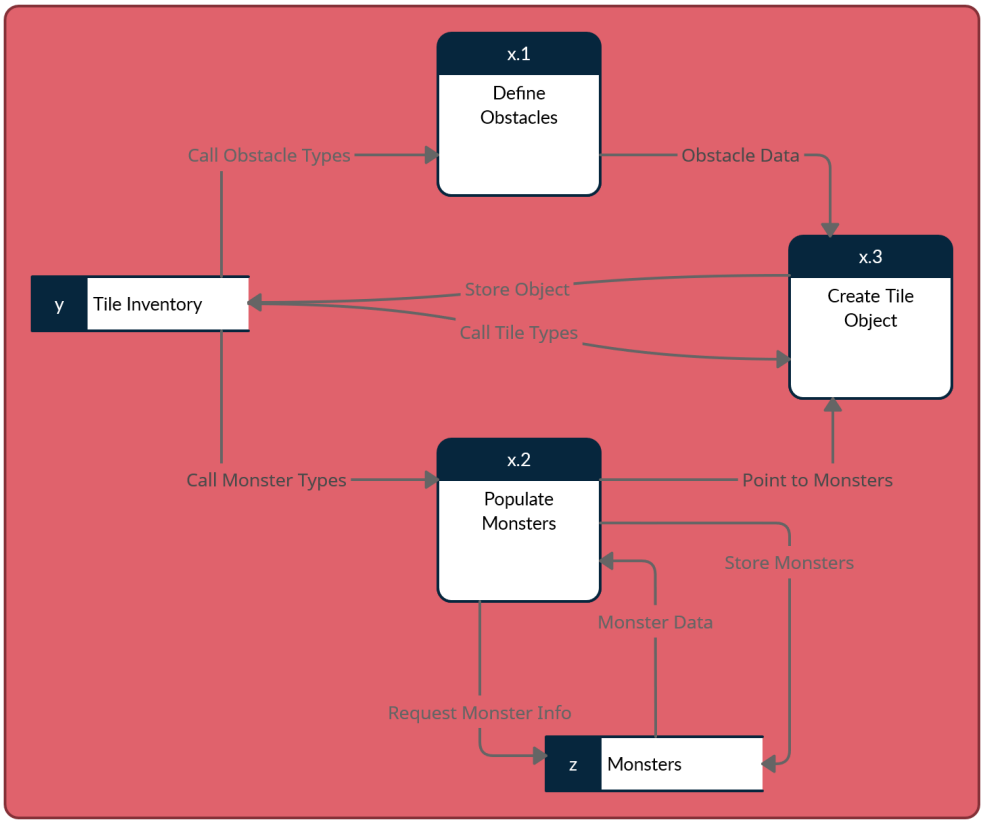
After receiving information on monsters to be populated, x.2 requests information about the monster types from the Monster database which returns that information.

x.2 then stores those monsters in the monster database and points x.3 to those monsters in the database.

x.3 then uses all this information to create the tile object and stores it in the Tile Inventory.

Data Flow Diagrams





4. Acceptance Tests _____9

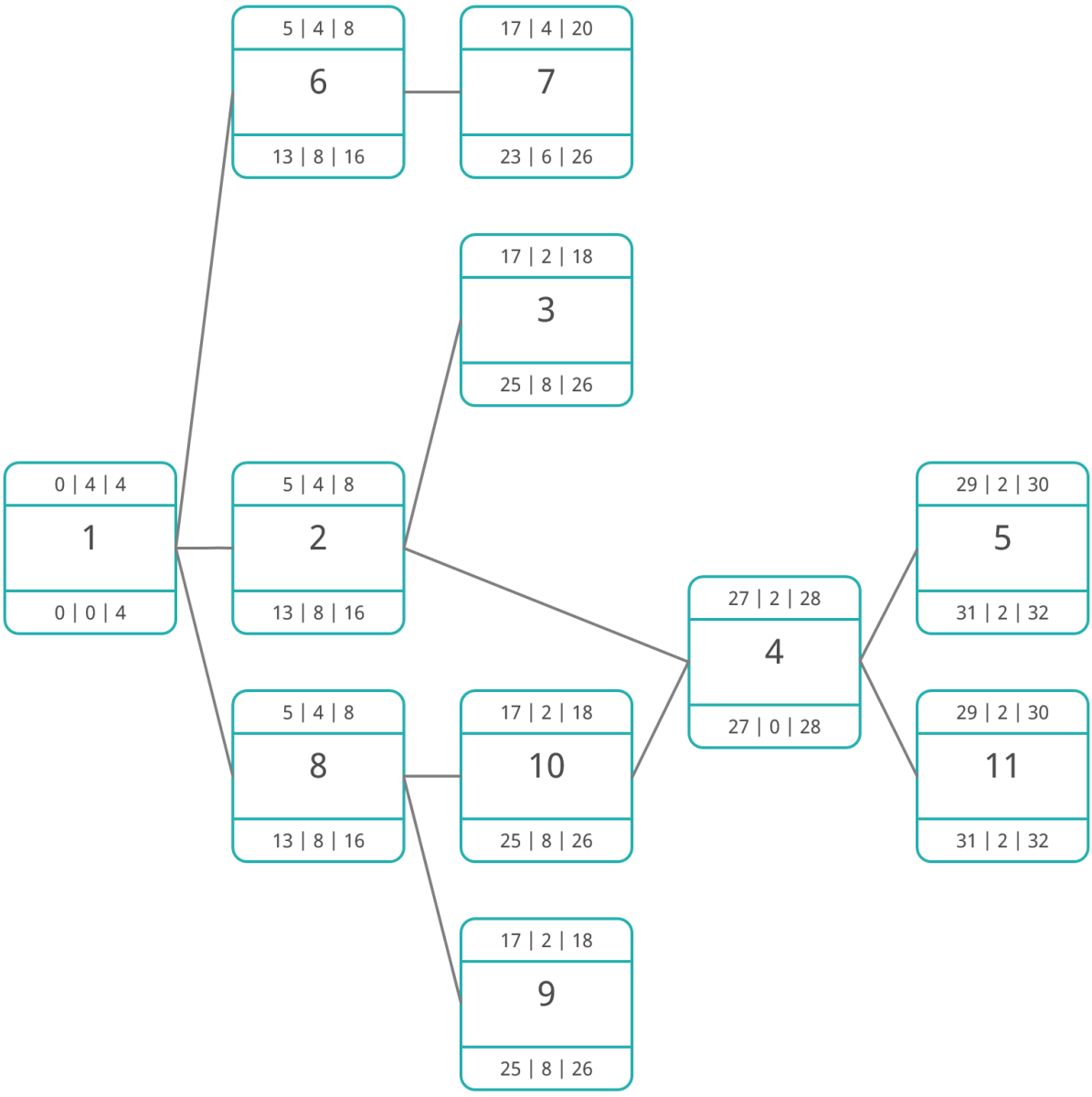
This will likely just be making up inputs to be used in the game, such as actual tiles, actual obstacles, actual monsters etc, and making sure they function properly within the class. Maybe automatically increasing the number of monsters called to make sure the tile doesn't store an impractical amount, or break when trying to. Likewise with doors and obstacles.

5. Timeline _____/10

Work items

Task	Duration (hours)	Predecessor Task(s)
1. Basic class	4	-
2. Linking Monsters	4	1
3. Monsters by Call	2	2
4. Plug in Combat	2	2, 10
5. Removing Monsters	2	4
6. Obstacles	4	1
7. Obstacles by Call	4	6
8. Doors	4	1
9. Doors by Call	2	8
10. Linking Heroes	2	8
11. Removing Heroes	2	4

Pert diagram



Gantt timeline