Work Breakdown Structure

Gaming Web Application

1.0 Requirements:

2.0 Design:

3.0 Build:

4.0 Q/A

5.0 Deliver:

1.1 - Refine
A. Ask for Specification
B. Meet with Client
C. Discuss possible outcomes

2.1 - Specify
A. Research Processes and Tools
B. Discuss Design of Application
A. Create UI Mockups
B. Design System Processes

3.1 - 3.1 -Initial Setup
A. Create schema/ database
B. Create barebone app
C. Docker containers

4.1: Deliver and Deploy Prototype

1.1 - Documentation
A. Discuss user cases
B. Create documentation for application

3.2 - Refactoring
A. Making app adjustments

3.2 - Build Features
A. Plug-ins/APIs
B. Create features for Requirements

4.2: Deliver Final Product
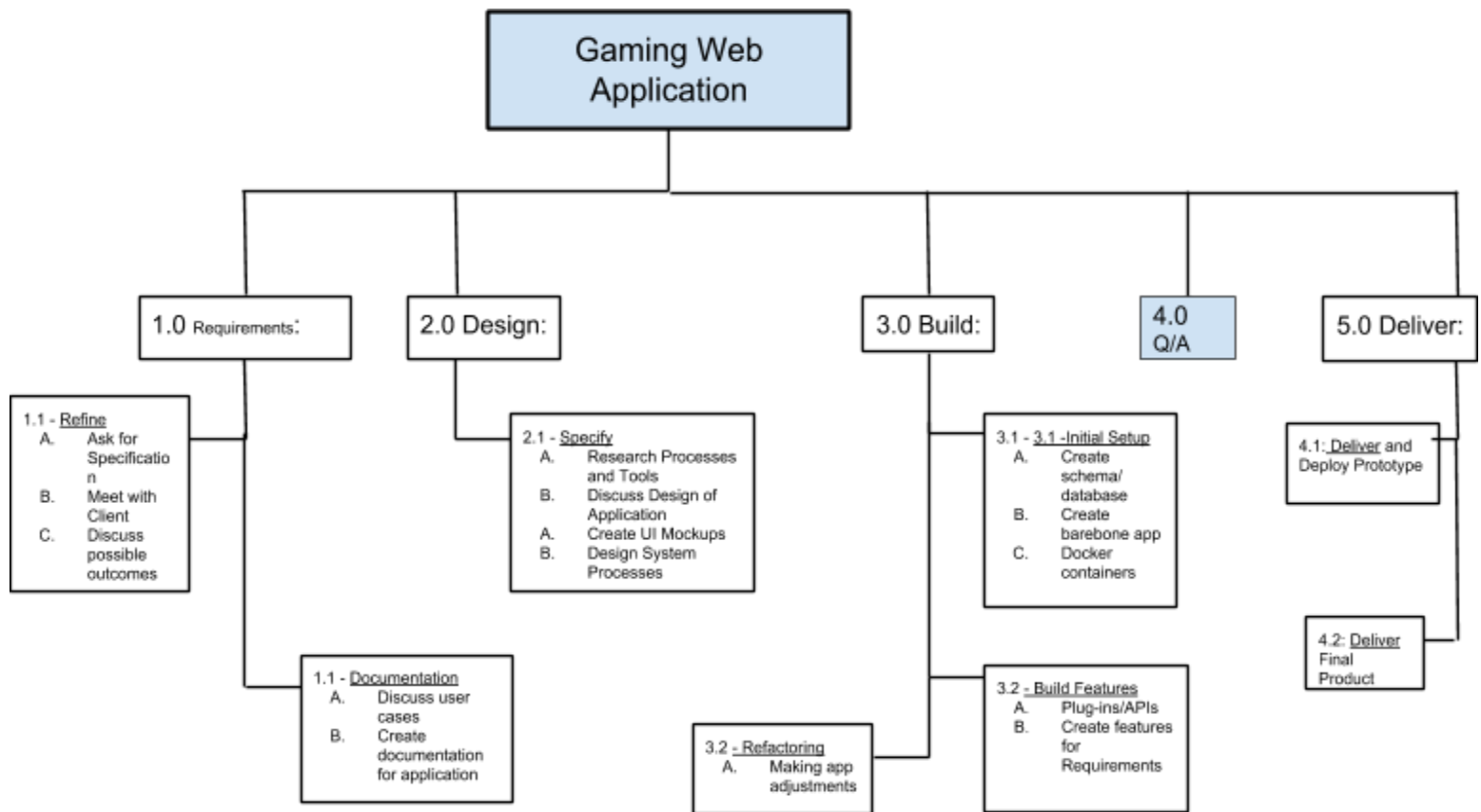
Work Breakdown Diagram:

Definition and Explanation of our WBS:

As stated above, our team has decided to work iteratively rather than linearly. We will work iteratively in terms of feature development. These features will be defined by the requirements listed above. During the development of a single feature, our development team will be responsible for analyzing, designing, constructing, and testing that feature.

The requirements step will entail gathering all of the necessary requirements needed to develop the feature. In short, the requirements analysis engineer will need to meet with stakeholders at the start of every feature development lifecycle.

During the design step, developers assigned to the feature will meet separately and determine a method for implementing the feature in the system. This plan will include low-level technical detail. The team should divide up the work amongst themselves and set a deadline during this step. This deadline should account for the time needed to test the feature.

Work Breakdown Structure

During the build, the development team will work independently on the tasks assigned to them during the design phase. The goal of the construction phase is to actually implement the feature. Developers should be testing as they develop during this phase.

Finally, all developers will come together during the Q/A phase and test the finished feature one last time before deployment. The deployment of the feature triggers the start of the next feature development lifecycle.

Tools to be used for feature planning/ development:

Waffle.io will be used to keep track of feature development. Waffle.io is a skin for Github issues that allows developers to easily break down large tasks, assign them to each other, and connect them to tasks that are acting as roadblocks.

During the design phase, developers may optionally choose to diagram their proposed implementation. For diagramming, we will use lucidchart.com.

Source:
https://expertprogrammanagement.com/2010/03/wbs-work-breakdown-structures-everything-you-need-to-know/

Selected Software Process:

We've decided to use Agile for our software process. Agile will allow us the flexibility to revise our processes while giving us the opportunity to refactor our code as we go. Additionally, the Agile process will allow us to:

1. Deliver software frequently
2. Reassess best practices
3. Constant communication with team and client on progress and requirements
4. Deliver a working prototype midway throughout the project

Sources:

https://linchpinseo.com/the-agile-method/

Possible Project Roles:
- Database Designer
- Database Developer
- Front-end Developer
- Back-end Developer
- UI Designer
- Project Manager
- Team Leads

Work Breakdown Structure