**1. Introduction**

**1.1 Project Overview:**

        Labyrinth is designed to be a 3D, multiplayer maze game where players will have to avoid obstacles and enemies to get to the destination before the other players.

**1.2 Project Scope:**

        Our game will allow multiple players from different desktops/PCs to compete against each other in a three-dimensional game environment. We will develop the game so that the final product consists of having the users spawn in a random location, and having them race to the finish line while avoiding enemies.

        Saving a player's progress in a game in the event of a network of server failure is out of the scope of the project and the player will have to join a new game

**1.3 Development Process:**

     We following the iterative model of software development. The product will be developed in the following two iterations.

*First Iteration:* This iteration will make the product into a fully networked 2D game that can handle multiple players with basic game functionality This iteration will include the following components:

|  |  |
| --- | --- |
| **Module** | **Purpose** |
| Backend Server | Simulates game play and manages user actions. |
| Basic 2D User Interface | Gives the user the ability to control avatar movements |
| Basic Maze Generator | Creates default 2D maze |
| Random Maze Generator (RMG) | Generates different maze configurations per each iterations of play |

*Second Iteration:* This iteration will bring the product to its full functionality as a networked 3D game. The following enhancements will be done in this iteration:

|  |  |
| --- | --- |
| **Module** | **Purpose** |
| Automated Enemy Bots | Enhances the backend server to control and manage multiple enemy bots to peruse the players |
| Advanced 3D User Interface | Enhancement will allows the user to interact with other users and enemy bots |
| Advanced Unity Gaming Server | Replaces the 2D game server to allow for the management of a 3D environment and players |
| Advanced Unity Gaming Clients | Replaces the basic 2D game clients to allow for greater player control |

**1.4 Effort, Schedule, and Team:**

        The team comprises of the following 5 persons:

        John Fouad

        Patrick Barron

        Christopher Wong

        Daniel Mathieu

Following is the schedule and effort for the three iterations:

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration #** | **Start Date**  **(mm/dd/yyyy)** | **End Date**  **(mm/dd/yyyy)** | **Total Effort**  **(person-hours)** |
| Iteration 1 | 09/25/2016 | 11/11/2016 | 100 |
| Iteration 2 | 11/12/2016 | 12/14/2016 | 160 |

Total Effort in man-hours: 260

Final Delivery Date: 12/14/2016

**1.5 Assumptions made:**

    No major assumptions beyond what is stated in the SRS

**2. Detailed Effort and Schedule**

We use the bottom up approach for estimations. In this we list the major modules and tasks, and then estimate their effort and schedule. Task assignments to project members is also specified.

**2.1 First Iteration:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Task** | **Estimated Effort**  **(man- hrs)** | **Start Date**  **(mm/dd/**  **yyyy)** | **End date**  **(mm/dd/**  **yyyy)** | **Person** | **Actual Effort (man-hrs)** |
| 1 | Backend Server | 34 | 9/25 | 11/11 | Mathieu |  |
| 2 | Multiple Player Handler | 10 | 10/27 | 11/3 | Mathieu |  |
| 3 | Create Multiple Playing Fields | 12 | 10/27 | 11/3 | Mathieu | 2 |
| 5 | Basic UI with Viewing and Movement Ability | 20 | 10/2 | 11/11 | Fouad |  |
| 5 | Create a Default Map | 10 | 10/8 | 11/11 | Barron & Wong |  |
| 6 | 2D Map Generator | 10 | 10/6 | 10/26 | Barron & Wong |  |
| 7 | Documentation | 4 | 10/25 | 10/26 | Fouad |  |

\**Note:* After the first iteration, Labyrinth will be a fully functional single player game with 2D maze rendering.

**2.2 Second Iteration**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Task** | **Estimated Effort**  **(man- hrs)** | **Start Date**  **(dd/mm/**  **yyyy)** | **End date**  **(dd/mm/**  **yyyy)** | **Person** | **Actual Effort**  **(man-hrs)** |
| 1 | Automated Enemy Bots | 10 | 11/17 | 12/6 | Mathieu |  |
| 2 | Advanced 3D User Interface | 30 | 11/17 | 12/6 | Mathieu & Fouad |  |
| 3 | Advanced Gaming Server | 60 | 11/17 | 12/6 | Wong |  |
| 4 | Advanced Gaming Client | 20 | 11/14 | 12/14 | Wong |  |
| 5 | Advanced 3D Game Environment | 20 | 11/17 | 12/6 | Barron & Fouad |  |
| 6 | Documentation | 10 | 12/2 | 12/6 | Barron & Mathieu |  |

\**Note:* After the second iteration, Labyrinth will be a finalized product, with an advanced user interface for player to player interactions and automated enemies to peruse players in a 3D environment.

**3. Team Organization**

We have a small team, so we will use a flat team structure of peers, with one person having an additional role as project manager and one person in charge of maintaining accurate documentation. Following table gives the organization:

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | **Description** |
| Daniel Mathieu | Team Leader | Head of organization and facilitating team decisions |
| John Fouad | Developer/Secretary | Head of User Interface |
| Pat Barron | Developer | Head of Gaming Environments |
| Chris Wong | Developer | Head of Multiplier 3D Gaming |

**4.**    **Hardware and Software Resources Required**

    For clients:

Hardware:       A workstation that can connect to the internet.

           Software:        A web browser.

    For developers:

             Hardware:       A workstation that can connect to the internet.

                                      A server that can network.

             Software:        C/C++ compiler for first iteration.

                                      Unity IDE for 3Ddevelopment.

                    Software repository for shared code

**5.    Quality Plan**

* *SRS/Architecture Review:* This will be reviewed by the team and may be influenced by suggestions from outside resources.
* *Design Review:*  This will be reviewed by the project team.
* *Unit Testing:*  Each team member will be responsible for testing his own code, but further testing may be performed by the rest of the team in order to meet specifications.
* *System Testing:*  This will be done in accordance with the system test plan, which will be reviewed by the project team.

**6.    Risk Management Plan**

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Probability** | **Impact** | **Mitigation Plan** |
| Not finishing in time | Moderate | Very High | Reduce certain features that would take too much time |
| Cannot implement 3D rendering | Moderate | Low | To make a 2D representation instead |
| Noticeable lag time due to insufficient server power | Moderate | Moderate | Reduce functions running at the same time, Loading more into the client’s browser, and/or using more servers |

**7.    Project Tracking**

We will primarily communicate through email and meetings inside and outside of class.  Status of tasks will be checked as predetermined checkpoint dates by the group and contingency plans will be created if necessary.