Revision History

Table 1: Revision History

Date	Developer	Change	Revision
September 29	Jiahao Li	Initial Draft	0
September 29	Pavithran Pathmarajah	Initial Draft	0
September 29	Viren Patel	Initial Draft	0

Team 6
JSTanks

SE 3XA3: Development Plan

Name	MacID
Jiahao Li	li577
Pavithran Pathmarajah	pathmap
Viren Patel	patelvh3

Meeting Plan:

The meeting plan we have come up with will ensure we are on schedule throughout this project. On top of the weekly labs on Mondays and Thursdays (ITB 236), we have decided to hold a meeting every Wednesday in Thode Library at 5:30pm. In addition, there will be a short online meeting on the team Facebook group every Sunday evening. This is to make sure that each team member is on track for completing their task. This online meeting will also allow for the planning of work in the coming week. These meetings are strategically set to ensure that consistent work is being done every week, since large gaps between two consecutive meetings are avoided. We have decided to rotate the facilitators with every meeting whom will be responsible for the meeting agenda, a tool that will be used to plan and reflect every meeting.

Team Communication Plan:

The primary means of communication our team will be using throughout this project is a Facebook group chat. The Facebook group will be used for sharing ideas, research, questions and issues that are discovered while working individually. However, the implementation (including programming issues) and documentation will be communicated through GitLab. We are also using google docs as a means to work together on the documents involved in this project.

Team Member roles:

Pavithran Pathmarajah:	Team Lead
Jiahao Li:	Scribe
Viren Patel:	JS-Specialist

Git workflow plan:

A good Git Workflow is key to the success of a project and must be determined before all else; our team has decided to go with a simple Git workflow where in we branch off of the master. Using this branch as a central branch we will be debugging and adding features merging with the master branch upon reaching milestones such as movement, graphics, etc. Merging conflicts with the central branch will be handled by the two members with conflicts working together to ensure neither of their features are affected. In short our workflow will consist of a central branch off of the master, with conflicts resolved as a team.

Proof of Concept Demonstration plan:

The proof of concept demonstration will show that we are capable of of completing what we set out to do, we will prove this by overcoming what we believe to be the most difficult task. Since we plan to learn JavaScript throughout the development process we believe that the most difficult task will be to take user input through the browser and into our game. The demo we plan to showcase will consist of a cube which moves around the browser window, with respect to user input, with bounds so that it does not go off screen. We are not deterred with the task of bringing our project to class in order to demonstrate for it is web-based and runs in the browser window. For our proof of concept demo we will open an html file which will run the script for our block with boundaries which will move based on keyboard input.

Technology:

Technology has always been an important part of software development. It is usually composed of four major parts which are the programming language, the integrated development environment, the testing framework and the document generation. For the part of programming language, the plan is to build the structure of the website with HTML and CSS, running JavaScript to complete the part of the game. As for IDE, we decided to use web- storm and notepad++. The reason of choosing web-storm is because of its smart and convenient features which are able to analyze the project to provide the best code auto-completion results with hundreds of built-in inspections. And notepad++ allows working with multiple open files in a single window which is better than the built-in Windows text editor. Pertaining to the testing framework, Qunit is a good choice for JavaScript which is similar to Junit which we have experience with for Java. As for documentation, YUIdoc was chosen because it can be used to document the code clearly and stably.

Coding Style:

Based on coding habits of all members in the group, a unified coding style has been set up. We decided to use K&R style for the placement of the braces in compound statement. For the part of naming variables, camelCase will be used throughout all the code which will make it look uniform, constants will be written in all capital letters. Every class will begin with a capital letter to distinguish them. Functions will end in 'Func' to distinguish them. We also agreed to follow commenting standards to ensure that when looking back at the code it will still be understandable. Indenting will be mandatory and be one tab for levels of logic to be easily recognizable, for example the code in a loop is indented to identify the code is running within the loop. All global variables will be declared in the top of the document, before any logic/code is implemented.

Project Schedule:

Refer to Gantt chart here

Project review (To be filled in: Revision 1):