Blaze Brigade

- Development Plan -

SFWR ENG 3XA3 - Section L02
007 (Group 7)

Jeremy Klotz - klotzjj
Asad Mansoor - mansoa2
Thien Trandinh - trandit
Susan Yuen - yuens2

December 7, 2016

Date	Developer(s)	Change
Sept 28, 2016	All	Created the Development Plan - Rev 0
Oct 5, 2016	Susan Yuen	Edited Development Plan to reflect decision to use Unreal Engine 4.
Oct 21, 2016	Susan Yuen	Edited Development Plan to reflect decision to use XNA Game Studio and Visual Studio for development.
Dec 6, 2016	Asad Mansoor	Modified Team Meeting Plan to include Location, Deliverables and a detailed Agenda. Included a new risk in the Proof of Concept section. Added content to the Project Review section - Rev 1

Table 1: Revision History

1 Team Meeting Plan

Table 2: Team meeting plan with their corresponding time and location.

Meeting ID	Week	Meeting Dates	Location
M1	Sept 12 - Sept 18	Sept 14: 10:30am-12:30pm Sept 15: 6:30pm-7:30pm Sept 16: 8:30am-10:30am	ITB-236 Lab, Online
M2	Sept 19 - Sept 25	Sept 21: 10:30am-12:30pm Sept 22: 6:30pm-7:30pm Sept 23: 8:30am-10:30am	ITB-236 Lab, Online
M3	Sept 26 - Oct 2	Sept 28: 10:30am-12:30pm Sept 29: 6:30pm-7:30pm Sept 30: 8:30am-10:30am	ITB-236 Lab, Online
M4	Oct 3 - Oct 9	Oct 5: 10:30am-12:30pm Oct 6: 6:30pm-7:30pm Oct 7: 8:30am-10:30am	ITB-236 Lab, Online
M5	Oct 10 - Oct 16	Oct 12: 10:30am-12:30pm Oct 13: 6:30pm-7:30pm Oct 14: 8:30am-10:30am	ITB-236 Lab, Online
M6	Oct 17 - Oct 23	Oct 19: 10:30am-12:30pm Oct 20: 6:30pm-7:30pm Oct 21: 8:30am-10:30am	ITB-236 Lab, Online
M7	Oct 24 - Oct 30	Oct 26: 10:30am-12:30pm Oct 27: 6:30pm-7:30pm Oct 28: 8:30am-10:30am	ITB-236 Lab, Online
M8	Oct 31 - Nov 6	Nov 2: 10:30am-12:30pm Nov 3: 6:30pm-7:30pm Nov 4: 8:30am-10:30am	ITB-236 Lab, Online
M9	Nov 7 - Nov 13	Nov 9: 10:30am-12:30pm Nov 10: 6:30pm-7:30pm Nov 11: 8:30am-10:30am	ITB-236 Lab, Online
M10	Nov 14 - Nov 20	Nov 16: 10:30am-12:30pm Nov 17: 6:30pm-7:30pm Nov 18: 8:30am-10:30am	ITB-236 Lab, Online
M11	Nov 21 - Nov 27	Nov 23: 10:30am-12:30pm Nov 24: 6:30pm-7:30pm Nov 25: 8:30am-10:30am	ITB-236 Lab, Online
M12	Nov 28 - Dec 4	Nov 30: 10:30am-12:30pm Dec 1: 6:30pm-7:30pm Dec 2: 8:30am-10:30am	ITB-236 Lab, Online
M13	Dec 4 - Dec 7	Dec 6: 6:30pm-7:30pm Dec 7: 10:30am-12:30am	ITB-236 Lab, Online

Table 3: Team meeting agenda with specified roles and deliverables

Meeting ID	Meeting Agenda	Roles	Deliverables
M1	Form the Blaze Brigade team and introduce team members	All - Engage in the formation and discussion	N/A
M2	Finalize on the project and develop problem statement	All - Engage in the discussion and share ideas for solution	Problem Statement
M3	Create development plan and finalize on programming methodologies	All - Assign issues and discuss plan	Development Plan
M4	Create Software Requirements Specification	All - Assign issues and discuss scope of requirements	Requirements Rev 0
M5	Implement Proof of Concept.	All - Download tools and code specific parts	N/A
M6	Structure Proof of Concept into a presentation	All - Prepare to demo and answer potential questions	Proof of Concepts Demo
M7	Discuss test methodologies and test cases	All - Assign issues and discuss scope of tests	Test Plan Rev 0
M8	Discuss any issues within system. Plan to integrate test cases	All - Share feedback of current build	N/A
M9	Discuss design pattern and decompose system by creating MIS and MG	All - Assign MIS and MG to various members	Design Document Rev 0
M10	Structure Rev 0 into a presentation	All - Prepare to demo and answer potential questions	Rev 0 Demonstration
M11	Discuss any issues within system. Plan to integrate changes	All - Provide feedback from Rev 0	N/A
M12	Formulate and present the final demonstration of Blaze Brigade	All - Develop and present portion of presentation	Final Demonstration Rev 1
M13	Create Test Report and make changes to Rev 0 documents	All - Assign to review documentation	Final Documentation Rev 1

2 Team Communication Plan

The team will use Skype for communication when outside of the agreed upon meeting times. These lines of communication are available when team members require assistance with their assigned work or require input from other team members on a topic of question. The team will also be using Slack to organize announcements regarding project development and deliverable deadlines.

3 Team Member Roles

- **Jeremy Klotz**: Algorithms Specialist, Developer
- **Asad Mansoor**: Tester, Developer
- **Thien Trandinh**: Gameplay Mechanic, Developer
- **Susan Yuen**: Git Master, Product Architect, Developer

4 Git Workflow Plan

After considering the different types of workflows, we concluded that **centralized workflow** best fits the requirements of this project. This is due to the fact that the project is relatively small - only spanning 12 weeks, and the team will see the project through from start to finish over this time period. As such, a release branch separate from a development branch is not necessary. In addition, team members will be working on aspects of the game pertaining to the same feature or features that rely on each other, so creating any additional feature branches are also unnecessary. Due to these reasons, feature-branch and gitflow are excluded. As a result, we decided on maintaining only one branch, and are thus implementing the centralized workflow for our project. Labels will be used to label any commits containing documents that are graded.

5 Proof of Concept Demonstration Plan

The proof of concept demonstration shall consist of the layout of the software architecture, including the skeleton of the majority of classes, functions, and implementation of the Model-View-Controller software design. The program shall have a working grid implemented, as well click detection and mouse functionality. The game shall also have one unit, which the player shall be able to move by first selecting the unit, then selecting another position on the grid to move the unit to. The player shall also be able to select and deselect the unit by clicking on the unit repeatedly.

Will a part of the implementation be difficult?

There is no significant risk other than implementing all minor details within the given time constraint. An unforeseen minor risk can be losing the license to the free software such as Visual Studio and XNA Game Studio provided by the McMaster Engineering Faculty. If that does occur, a new copy may need to be purchased or there could be a change of the technologies being used to deliver the solution.

Will testing be difficult?

Testing will not be difficult as the team members have experience with unit testing in frameworks such as JUnit. Although our project will be coded in C#, the unit tests will share similar concepts and ideas to our previous experience with JUnit. As such, our automated unit tests will be able to cover whitebox testing of single functions and state variables.

Is a required library difficult to install?

The programs that will be used to develop, which include Visual Studio 2015 and XNA Game Studio, are available for free download to students in McMaster's Software and Computing Department, and therefore not be difficult. The downloads are available online through McMaster's CAS department.

Will portability be a concern?

Portability will be a concern as the platform that the game supports only includes Windows, due to the restriction of the software the game is developed with. Visual Studio and XNA Game Studio are supported by Microsoft, therefore the game will only be able to support Windows OS.

6 Technology

- **Programming Language**: C#
- **IDE**: Visual Studio 2015, XNA Game Studio
- **Testing Framework**: Visual Studio Unit Testing Framework
- **Documentation**: LaTeX
- **Other**: Git, Photoshop

7 Coding Style

The coding style for the project will follow Microsoft's C# Programming Guide (https://msdn.microsoft.com/en-us/library/ff926074.aspx).

8 Project Schedule

Please refer to the .gan file for access to the project Gantt Chart within the DevelopmentPlan folder.

9 Project Review

Upon reaching Revision 1 of the Blaze Brigade project, a few changes have been made to various documentation either as an update to outdated information or implementation on the feedback provided. Considering its timeframe, the project has been a success as the team has completed all of the objectives and goals that were initially planned. Although this was a first time for many team members being involved in a large-scale software solution, the feedback and personal experience throughout this learning process has shown what has worked well, and what did not work out as well will be improved on for future projects.

The consistency in delivering the right amount of content of each documentation has been a main contributor to the success of the project. Normally, all tasks would be assigned with the deadline set earlier than the actual due date to provide a buffer for edits or updates. The correct use of technology has worked well in the implementation of the project and provided no limitation in delivering the assigned solution. Each team member's role were assigned and everyone was well aware on where they stand in delivering their assigned contribution. The team would continue using the concepts presented within this development plan as a reference to future projects or even Blaze Brigade if it were to continue to Revision 2.

Areas of improvements can be seen in the planning of team communication and git workflow. Team communication was active during the labs, but the times during the evenings or weekend would be minimal. This can be changed if everyone were to agree to have a companion communication application on their desktop as well as their mobile devices. Moreover, a scope of switching communication application is also a possibility to combine the chatting aspect with the organization of tasks. Learning how to use git was a learning curve for some team members and prompted for a better guide during the initial planning phase of the project. There also has to be a method such to check each person's commits and level of contribution to ensure the right metric on what gets achieved for each issue.

In regards to a future project, the team would better modify the development plan to be a little more detailed and updated in terms of its correctness, as the development plan was essential on how the team should work. The meeting plan should contain more related agenda information, to ensure all the following points get covered in the meeting. In addition, there is a scope of incorporating meeting minutes into future meetings, such that everyone is well aware of the decisions taken place.

Overall, the quality of the team meetings have been productive and clear. Changes within the team communication and time management should be prioritized in the next revision to meet the team's full potential. One particular change is switch the central communication application to include both the chatting aspect as well as organizing tasks. This could mean to fully transfer communications onto Slack or Basecamp. Furthermore, a larger scope should be placed on how tasks should get assigned to avoid unaligned balance of tasks given to a specific team member.