

Blaze Brigade

- Software Requirements Specification -

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

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

October 11, 2016

Contents

1	Project Drivers	3
1.1	The Purpose of the Project	3
1.2	The Stakeholders	3
1.2.1	The Client	3
1.2.2	The Customers	3
1.2.3	Other Stakeholders	3
1.3	Mandated Constraints	4
1.3.1	Solution Constraints	4
1.3.2	Off-the-Shelf Software	4
1.3.3	Schedule Constraints	4
1.3.4	Budget Constraints	4
1.4	Naming Conventions and Terminology	5
1.5	Relevant Facts and Assumptions	5
1.5.1	Relevant Facts	5
1.5.2	Assumptions	5
2	Functional Requirements	7
2.1	The Scope of the Work and the Product	7
2.1.1	The Context of the Work	7
2.1.2	Work Partitioning	7
2.1.3	Individual Product Use Cases	8
2.2	Functional Requirements	8
2.2.1	GUI	8
2.2.2	Game Structure	8
2.2.3	Unit Movement	9
2.2.4	Unit Attacking	9
2.2.5	Unit Structure	10
2.3	Combat Damage Calculations	10
2.3.1	Hit Rate	10
2.3.2	Critical Rate	11
2.3.3	Physical Damage	11
2.3.4	Magical Damage	11
2.3.5	Number of Hits	11
2.3.6	Damage Dealt	11
3	Non-functional Requirements	12
3.1	Look and Feel Requirements	12
3.1.1	Appearance Requirements	12
3.1.2	Style Requirements	12
3.2	Usability and Humanity Requirements	12
3.2.1	Ease of Use Requirements	12
3.2.2	Learning Requirements	12
3.2.3	Understandability and Politeness Requirements	12
3.2.4	Accessibility	12

3.3	Performance Requirements	12
3.3.1	Speed and Latency Requirements	12
3.3.2	Safety-Critical Requirements	13
3.3.3	Precision Critical Requirements	13
3.3.4	Reliability and Availability Requirements	13
3.3.5	Robustness or Fault-Tolerance Requirements	13
3.3.6	Capacity Requirements	13
3.3.7	Longevity Requirements	13
3.4	Operational and Environmental Requirements	13
3.4.1	Expected Technological Environment	13
3.4.2	Productization Requirements	13
3.5	Maintainability and Support Requirements	13
3.5.1	Maintainability	13
3.5.2	Portability and Adaptability	13
3.6	Security Requirements	14
3.6.1	Access Requirements	14
3.6.2	Integrity Requirements	14
3.7	Cultural Requirements	14
3.8	Legal Requirements	14
3.9	Health and Safety Requirement	14
4	Project Issues	15
4.1	Open Issues	15
4.2	Off-the-Shelf Solutions	15
4.3	New Problems	15
4.4	Tasks	15
4.4.1	Project Planning	15
4.4.2	Planning of the Development Phases	16
4.5	Migration to the New Product	16
4.6	Risks	16
4.7	Costs	17
4.8	User Documentation and Training	17
4.8.1	Training Requirements	17
4.9	Waiting Room	17
4.10	Ideas for Solutions	17

List of Tables

1	Revision History	3
---	----------------------------	---

List of Figures

1 Project Drivers

1.1 The Purpose of the Project

Blaze Brigade serves as a tribute and adaptation of a popular genre: tactical simulation role-playing games. The game aims to provide a form of entertainment to the general public, presenting a fun strategical challenge to its players. The purpose of the project is to recreate an already existing tactical role-playing game, Tactics Heroes, and to ultimately further improve on the overall user experience by implementing additional features not already included in Tactics Heroes.

1.2 The Stakeholders

1.2.1 The Client

The client of the project is Dr. Spencer Smith in the Computing and Software Department at McMaster University.

1.2.2 The Customers

The customers of the game include members of the general public who are interested in the tactical role-playing game genre and are willing to play the game.

1.2.3 Other Stakeholders

Other stakeholders of the project include:

- The Blaze Brigade team

The project will serve as a means of entertainment and project development experience to its developers.

- Tactics Heroes developers

The game may serve as a guideline to the original developers of Tactics Heroes to further improve their game by incorporating our projects additional add-on features, as well as modern software development tools and concepts.

Table 1: **Revision History**

Date	Author	Notes
October 10, 2016	All	Revision 0 of Requirements Document
October 21, 2016	Thien Trandinh	Added changes to functional and Non-functional requirements

1.3 Mandated Constraints

1.3.1 Solution Constraints

Description: The game shall support Windows OS.

Rationale: The client will access the game using a device running Windows.

Fit Criterion: The game shall be developed to support these three operating systems, and testing will ensure that the game runs smoothly on all three operating systems.

1.3.2 Off-the-Shelf Software

The product will be utilizing XNA Game Studio for game development, as well as Adobe Photoshop for graphic editing. A device that is able to run XNA Game Studio is also required for project development. Customers must be in possession of an operating system to be able to play the game, and a pointing device to be able to select items on the screen.

1.3.3 Schedule Constraints

Full development of the game does not have any applicable schedule constraints. However, the basic functions of the game must be fully completed by December 2016.

1.3.4 Budget Constraints

The budget for the project is \$0 as all software required for development is provided for free. In addition, all hardware required is already owned by the project's respective developers.

1.4 Naming Conventions and Terminology

Term	Definition
RPG	Role-playing game, in which the player plays the game as a character.
AI	Artificial Intelligence.
Unit	A single character in the game.
Enemy	Units that are hostile to the player's units and may attack the player's units when in range.
Critical hit	Causes the user to deal more damage than expected from a normal hit.
HP	Health points; the amount of damage a unit may take before being defeated.
STR	Strength; Used in calculating physical damage dealt by the unit.
INT	Intelligence; Used in calculating magical damage dealt by the unit.
SKILL	Used to calculate unit's hit rate and critical hit rate.
SPEED	Used to calculate a unit's number of hits on the enemy.
DEFENSE	Used to calculate the melee damage dealt to the unit.
RESISTANCE	Used to calculate the magic damage dealt to the unit.

1.5 Relevant Facts and Assumptions

1.5.1 Relevant Facts

The game that Blaze Brigade will re-implement, Tactics Heroes, was initially named Fire Emblem Factory. The game is heavily influenced from a Japanese game series called "Fire Emblem", which was released in 1990 by Intelligent Systems and Nintendo. Fire Emblem revolutionized the genre by incorporating RPG elements, and as such, the success of the genre in modern day can largely be attributed to this series. In addition, Tactics Heroes is still in alpha stage, and only the basic gameplay mechanics have been fully implemented. Future content for additional features in Tactics Heroes is still under works by its developers.

1.5.2 Assumptions

It is assumed that upon project launch, the host site for the game's download will be free. It is also assumed that members of the general public will not use

any aspect of the project or its source code for malicious intent or for resale for monetary value.

2 Functional Requirements

2.1 The Scope of the Work and the Product

The developers of Blaze Brigade must create a tactical turn-based game, in which the objective is to eliminate all of the enemy units.

2.1.1 The Context of the Work

The game will be available to play on all desktop and laptop capable of running Windows (or Windows VM), and will be packaged and deployed Visual Studio. The developers will first develop a working game, and will then allow others to use and evaluate the prototype. Afterwards, the developers will use the feedback to refine the code further and repeat the evaluation process. These evaluation stages are crucial for developing software that meets the customers' desires, as the game should be fun and entertaining for those who play it.

2.1.2 Work Partitioning

Event #	Event Name	Input	Output
1	Mapping System	Developer Code	Executable File
2	Mechanics	Developer Code	Executable File
3	Units	Developer Code	Executable File
4	Terrain Types	Developer Code	Executable File
5	Sprites	Developer Code/Graphics	Executable File
6	Map Creation	Developer Code	Executable File
7	Audio	Microphone	Audio Output Device
8	Deployment	Developer Code	Executable File

Event #	Summary of BUC
1	A grid system that simulates the playing board.
2	Built-in combat system.
3	A class that all playable units are modelled by.
4	Properties of terrain types.
5	A collection of sprites used for terrain types, and playable units.
6	A playing board with all sprites added.
7	Sound effects and background music.
8	To make Blaze Brigade playable on all operating systems.

2.1.3 Individual Product Use Cases

Product Use Case	Description of Use Case
Playing the Game	The user plays the game and has a sense of enjoyment.
Stimulating the Brain	The user thinks about the game at a deeper level. This causes the player to exercise and train their brain.

2.2 Functional Requirements

The following contains the functional requirements of the project:

2.2.1 GUI

1. The GUI will be controlled with mouse input
2. The game will contain a main menu on screen upon launch
3. You will be able to select New Game from the main menu
4. You will be able to select Load Game from the main menu
5. You will be able to select How-To-Play from the main menu
6. You will be able to select exit Game Game from the main menu

2.2.2 Game Structure

1. The game shall be turn-based.
2. A player's turn shall end once all their units have performed available actions or if the game ends.

3. The game will consist of 2 players alternating turns
4. When one player's turn ends, the next player's turn begins.
5. Each player will receive the same amount of units, and be unable to gain anymore until the next match.
6. A unit shall only be able to move and attack once per turn.
7. During a unit's turn, clicking a unit will give a drop down menu with available actions
8. One side will be victorious when the other side has no playable units left.

2.2.3 Unit Movement

9. Units will be able to select move as an available option after clicking on a unit that has yet to perform its action
10. Units will only be able to move within their move range
11. Units shall not be able to pass through obstacles.

2.2.4 Unit Attacking

12. Units will be able select attack as an available option after clicking on a unit that has not attacked this turn, regardless of having moved or not
13. A unit will only be able to attack a unit within it's attack range.
14. All units shall be able to attack enemy units.
15. Units will be unable to move after attacking.
16. Attack information displaying damage, hit rate, and crit rate will be displayed to screen
17. Units will lose HP according to damage calculation.
18. Units that fall below 1 HP will be deceased and no longer be able to be used in the current battle.
19. Player shall be able to select which weapon each unit uses to perform an attack.

2.2.5 Unit Structure

20. All units shall have a corresponding Unit type class.
21. Classes shall include:
 - (a) Warriors, which will be physical oriented units with a focus on Strength and Defense.
 - (b) Archers, which will be a ranged physical oriented unit with a focus on Strength, Skill, and Speed.
 - (c) Mages, which will be a magical oriented unit with a focus on Magic and Resistance.
22. All units will have stat values corresponding to their class.
23. The stats of the game shall include:
 - (a) Strength stat will be used to calculate physical damage.
 - (b) Intelligence stat will be used to calculate magical damage.
 - (c) Defense stat will be used to calculate physical defense.
 - (d) Resistance stat will be used to calculate magical defense.
 - (e) Skill stat will be used to calculate hit, miss, and critical rates.
 - (f) Speed stat will be used to calculate if a unit can attack twice.
24. All units will have a unique corresponding HP bar.

2.3 Combat Damage Calculations

For unit x = attacking unit, and unit y = defending unit where:

- `unit.skill` gets the unit's skill
- `unit.str` gets the unit's strength
- `unit.int` gets the unit's int
- `unit.def` gets the unit's defense
- `unit.res` gets the unit's resistance
- `unit.spd` gets the unit's speed

2.3.1 Hit Rate

$\text{hitRate} = [(x.\text{skill}/10 - y.\text{skill}/10) + 1] * 0.8 * 100$,

where `hitRate` is 80% if both players have the same skill, and adds or subtracts 8% to `hitRate` in favour of the unit with the higher skill stat.

2.3.2 Critical Rate

$\text{critRate} = [(x.\text{skill}/3 - y.\text{skill}/3) + 1] * 0.81 * 100$,

where critRate is 10 if both players have the same skill, and adds or subtracts 3.33% to critRate in favour of the unit with the higher skill stat.

2.3.3 Physical Damage

$\text{physicalDamage} = x.\text{strength} - y.\text{defense}$,

where physical damage is the damage calculated from a physical attack.

2.3.4 Magical Damage

$\text{magicalDamage} = x.\text{int} - y.\text{res}$,

where physical damage is the damage calculated from a magical attack.

2.3.5 Number of Hits

$\text{numHits} = 2$ if $x.\text{spd}$ is greater than $y.\text{spd} + 4$, else $\text{numHits} = 1$.

2.3.6 Damage Dealt

$\text{damageDealt} = (\text{physicalDamage or magicalDamage}) * \text{numHits} * (\text{hitRate, which will be 1 if the attack hits or 0 if the attack misses}) * (1.5 \text{ if it is a critical hit or } 1 \text{ if it is not a critical hit})$.

3 Non-functional Requirements

3.1 Look and Feel Requirements

3.1.1 Appearance Requirements

The product will display the '007' logo upon start up for a reasonable amount of time, then will display the menu page afterwards. The menu page will consist of a nature-like background (such as trees), with clouds floating across the top. The actual game will consist of maps pertaining to varying locations and themes, such as a volcano or a tundra. The characters to be controlled will be drawn and coloured in such a way that they do not blend into the background. The resolution of the game will be preset and unchangeable from a 640x480 resolution.

3.1.2 Style Requirements

The style of the game will be a pixelated tactical RPG, similar to classics like Fire Emblem, and will have a nostalgic appeal to those who are familiar with this genre of old school RPGs.

3.2 Usability and Humanity Requirements

3.2.1 Ease of Use Requirements

The game shall be playable by persons who are capable of thought and are able to operate a pointing device (ie. mouse or touchpad).

3.2.2 Learning Requirements

The user shall be able to operate a computer or pointing device.

3.2.3 Understandability and Politeness Requirements

The user shall be able to operate a computer or pointing device.

3.2.4 Accessibility

The game shall be playable on Computers running Windows, with a screen size greater than 1600x960.

3.3 Performance Requirements

3.3.1 Speed and Latency Requirements

The response between the game and user input should be near instant.

3.3.2 Safety-Critical Requirements

The game must meet standard video game regulations.

3.3.3 Precision Critical Requirements

The gameplay combat statistics shall accurately reflect what is displayed on the screen. Changes in these statistics shall also be accurately reflected upon proper user input.

3.3.4 Reliability and Availability Requirements

The game shall be available for as long as the user has access to a Windows machine.

3.3.5 Robustness or Fault-Tolerance Requirements

The game shall run for as long as the machine is on, or until the user exits the game.

3.3.6 Capacity Requirements

The game shall be able to support gameplay for up to 2 players.

3.3.7 Longevity Requirements

The game shall be playable for as long as Windows OS supports the game file type.

3.4 Operational and Environmental Requirements

3.4.1 Expected Technological Environment

The user shall be accessing and navigating the game from a computer running Windows, Mac, or Linux OS.

3.4.2 Productization Requirements

Before the product is playable, it must be deployed through Visual Studio 2015 and XNA Game Studio.

3.5 Maintainability and Support Requirements

3.5.1 Maintainability

The game should be ready for launch without need for further maintenance.

3.5.2 Portability and Adaptability

The game shall be playable on OS.

3.6 Security Requirements

3.6.1 Access Requirements

The game shall be made available to the general public through a free online download link hosted by a remote hosting site. The public shall be able to access the game given that they have previously acquired internet access and access to a machine capable of running the game.

3.6.2 Integrity Requirements

The game shall not accept invalid user input, nor have the source code modified by any persons or apparatus who are not permitted to do so.

3.7 Cultural Requirements

The game shall not contain any form of content that may be found offensive to any religious or ethnic group. The game shall state that any resemblance in game to real people or events is purely coincidental.

3.8 Legal Requirements

The game shall comply with the law.

3.9 Health and Safety Requirement

The patterns of lights from the game screen can result in epilepsy seizures and, or blackouts to a very small percentage of certain individuals. This may occur in individuals with no past history of seizures or blackout.

4 Project Issues

4.1 Open Issues

The developers of Blaze Brigade must become familiarized with the C++ programming language and investigate how to efficiently adapt to the specific coding standard and testing methodologies. With the use of new technologies, such as a XNA Game Studio, additional time would need to be allocated to learn this tool in depth. In addition, a feasibility study is yet to be completed to indicate whether a mouse-only control would be a feasible for all the possible scenerios present throughout the game play.

4.2 Off-the-Shelf Solutions

Blaze Brigade makes use of Tactics Heroes, an open source program that motivates the creation of a tactical simulation role-playing game. The engine streamlines the process of implementing the framework, and the developers of Blaze Brigade would simply need to implement maps, units, characters and the storyline. Rather than re-inventing the framework, we can use the pre-existing model of Tactics Heroes, and simply polish it in order to fulfil the enhancement requirement of the project. In addition, the open-source program presents a software license agreement and encourages developers to contribute to the program by getting involved within the forums.

4.3 New Problems

No new problems are tracked at this stage of the project.

4.4 Tasks

4.4.1 Project Planning

The game shall be completed using the prospective timeline:

Task	Resource	Start Date
Problem Statement	Developers	Sept 20
Development Plan	Developers	Sept 23
Requirements Document - Revision 0	Developers	Oct 3
Test Plan - Revision 0	Developers	Oct 21
Design Document - Revision 0	Developers	Sept 23
Development of Blaze Brigade	Developers	Oct 31
Testing of Game	Developers	Dec 1
Debugging of Game	Developers	Dec 1
Survey Round 1	Client	Dec 12
Implement changes from survey feedback	Developers	Dec 13
Survey Round 2	Client	Dec 20
Implement final changes from survey feedback	Developers	Dec 21

4.4.2 Planning of the Development Phases

The tasks listed in the table above will serve as the main guide for the developers to ensure that deliverables are completed in correct succession. In that regard, the output will serve the purpose to ensure that all of the requirements have been met before starting the next iteration of development in the next phase. In addition to the development phases, documents such as Problem Statement, Development Plan, Software Requirements Specification, Design Document, and Test Plan would need to be created and updated for the stakeholders as the project proceeds.

4.5 Migration to the New Product

Not applicable at this stage of the project.

4.6 Risks

The risk associated with the project comes with the nature of developing a game. For instance, accommodating testing into the game play would be a time-consuming and trivial task. Hence, the investigation on how to proceed with a specific testing standard and whether to introduce an automated testing framework is still underway at this stage of the project. In relation to the design aspect, a feasibility study is yet to be completed to indicate whether a mouse-only control would be feasible throughout all the scene of the game play. In addition, research will have to be conducted to ensure that the product is compatible on multiple operating systems. In addition, extra time would need

to be allocated in the process of learning and adapting to new technologies to ensure all of the objectives are met in the constrained time frame.

4.7 Costs

There is no cost associated in the development of the product. The developers of Blaze Brigade will make best use of their personal computers for the main development environment, with the addition of already-owned tools such as XNA Game Studio and Adobe Photoshop.

4.8 User Documentation and Training

4.8.1 Training Requirements

Ideally, an on-screen tutorial will be shown to first time users, and will guide them on how to interact with the game to meet its objectives. As the tutorial completes, the user will be directed to the story of the game. If time does not allow for this implementation, readable instructions will be added as a menu feature presented to the user upon game launch. In addition, instructions can be referenced later at any point of the game.

4.9 Waiting Room

No backlogged requirements are indicated at this stage of the project. This section will be updated as we proceed through the project planning phases.

4.10 Ideas for Solutions

No ideas for solution are indicated at this stage of the project. This section will be updated as we proceed through project planning phases.