S1

Mom and Dad’s Pizza Pad Pizza Adventure

Detailed Requirements and Management Plan

Team C - COMP 3663 X1
Scott Barnett 100123218, Jimmy Flemming 100116643,
Liam MacKinnon 100114393, Martin Main 100110854
February 8, 2016

Table of Contents

Glossary

Bazooka

This is a large weapon, used traditionally to fire rocket propelled explosives at vehicles and tanks during war. It is not traditionally used to shoot pepperoni.

GUI

Stands for “Graphical user interface” and refers to the type of user interaction system that appears as various buttons and graphics on a screen which users can interact with using their mouse or keyboard.

GUMP

This stands for “Giant Ultra Mutant Pepperoni” and is a part of the game *Mom and Dad's Pizza Pad's Pizza Adventure* described in part 3.7.1.6.1.1.1.

JRE

Stands for “Java Runtime Environment”, which is a program that needs to be installed on the computers running the system. This program is already installed on most modern computers. It allows programs written in the programming language “Java” to be run on the computer. This runtime environment has many versions, and a sufficiently late version of the runtime environment will need to be installed on the computers running the game.

Linux

This is an operating system that is an alternative to Windows and Mac OS

LORP

This stands for “Lump of radioactive pepperoni” and is a part of the game *Mom and Dad's Pizza Pad's Pizza Adventure* described in part 3.7.1.6.1.1.1.

man-hours/man-days

This is a standard measure of time commitment on a project, which measures the number of hours a single person would take to do the project. This number can be divided by the number of members of a team to determine the approximate amount of time each member will spend on the project. This term can also include people of other genders.

OS

This stands for “Operating System”, and refers to the base system installed on any computer from which all other programs are run.

UI

Stands for “User interface” and refers to any system that a user of the system would use to interact with the system.

1. Introduction

This is the primary specification document for the development of the software Mom and Dad’s Pizza Pad’s Pizza Adventure video game. This is a piece of software designed to be relevant to the current teen generation. It is a fun PRPG (Pizza Role Playing Game) It involves a character named Pizza boy Joe, and the rescue of pizza from the antagonists of the game, the Anger Crew. This game is designed to be accessible and easy to use from any common computer for a teen audience.

This document contains requirements and a management plan for this project. The first section is the customer requirements, which provides a simple to read requirements document similar to the one we have received from the customer. The next section is the detailed requirement, which goes into greater detail to the structure and architecture of the system, to aid designers in the production of the system. The last major section is the project management plan, which outlines the management of the project, including the roles and delegations of our team, a schedule of deliverables, and the management structure being implemented.

2. C-Requirements Definition

2.0 Introduction

The following section describes the customer requirements of the game. These requirements are meant to be non-technical descriptions of the basic structure of the game, without going into deep detail of specifics.

2.1 Basic Functional Requirements

Fun and replayable

Standard time:

- default is 5 min.
- can be set by player

Option to see current progress

Method to save

High score (employee of the month):

- affected by remaining slices of pizza

2.2 Plot Requirements

Pizza stolen by anger crew

Try to recover pizza before it's eaten

Deliver whatever was recovered before time runs out

2.3 Character Requirements

Can be created by user

- default male is Pizza Boy Joe
- default female is Garlic Finger Gina

Given unique stats upon creation

- endurance: total energy capacity
- speed: time frame to change move, after seeing opponent’s move
- intelligence: display opponents tendencies

100 starting energy

0 energy = pass out

Alert when energy is low, or attribute is changed

2.4 Item Requirements

Static inventory

In combat and out of combat items

- In combat
  - Soda can: restores small amount of energy
  - Baked beans: increases chance of successful attack
  - Energy drink: increase speed temporarily

- Out of combat
  - Soda can: restores small amount of energy
  - Mr. Michel's map: reveal's nearby tile contents
  - Angry pizza: restore's large amount of energy, but reduces total capacity
  - Pizza Slice: consume pizza to regain energy

2.5 Combat

Follows a Rock, Paper, Scissors format

2.6 Map Requirements

Grid style map

Starts on grid tile

Max one anger crew per tile

- random placement

Authentic street names

Each tile has three qualities

- Slipperiness
- Toxicity
- Pizza whiff

2.7 UI Requirements

Records and prints all relevant actions that happen

Keep the player up to date

2.8 Basic Non-functional Requirements

- Simple graphics
- Run on all desktop platforms
- Easy to open and run
- Downloadable from website

- Market Mom and Dad’s pizza throughout game
- Diss other pizza places represented in game

2.9 Undesired Requirements

- Offensive to any groups of people, except rival pizza shops
- Lengthy user manual. Rather have simple and concise document broken up with pictures and variances in fonts, and intuitively organized for ease of end-user's use.

3. Specific Requirements

3.0 Introduction

The following section describes the detailed requirements of the game. These requirements are meant as an aid to developers in the actual design and implementation of the game, and discuss the deeper architecture and specific parts of the game in detail.

3.1 External interface requirements

3.1.1 User interfaces

The UI will be made of multiple GUI windows. These windows are as follows:

1) Title screen

2) Character Creation/Load Game

3) City Street Tiles

4) Fight Screen

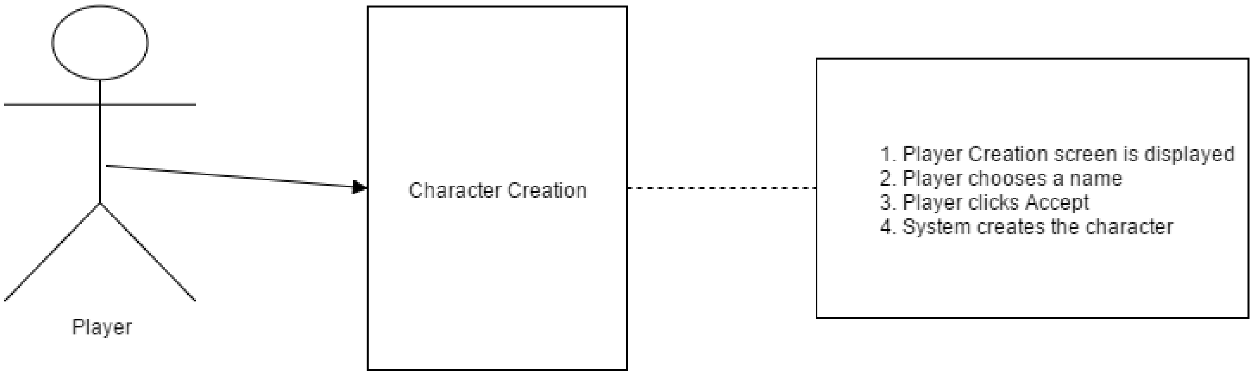
5) Inventory Screen

3.1.1.1 Title Screen

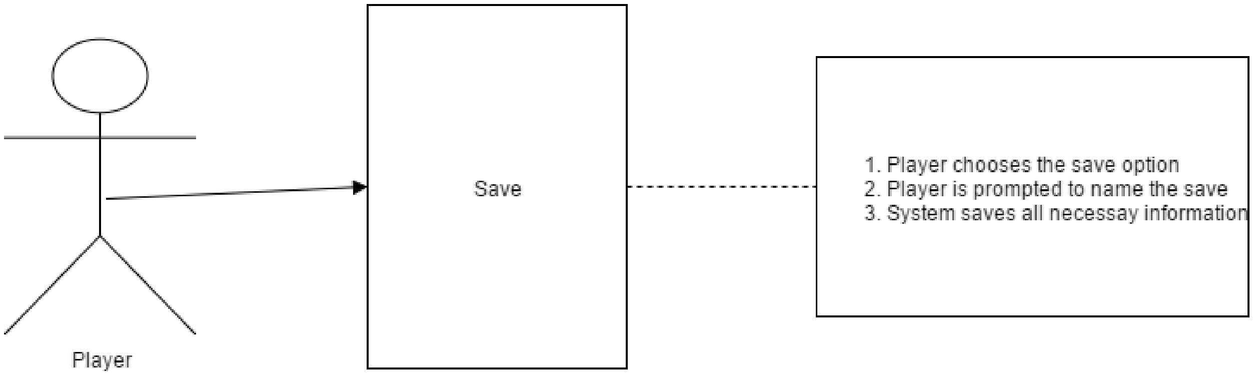
This screen displays the game's title, a list of high scores, and the text “press any button to start” pressing a button will bring you to the Character Creation/Load Game Screen.

3.1.1.2 Character Creation/Load Game

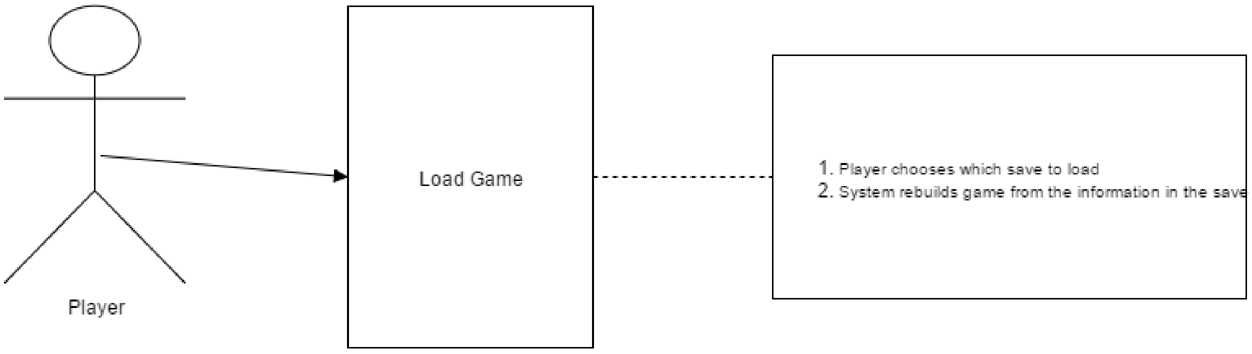
This screen gives the option for either creating a new character or loading a previously saved game. If character creation is chosen the screen changes to allow the user to select a character and give them a name (Default names are Pizza Boy Joe and Garlic Finger Girl Gina). Clicking create or loading a game will bring you to the Street Tile Screen.



1. Player Creation screen is displayed
2. Player chooses a name
3. Player clicks Accept
4. System creates the character



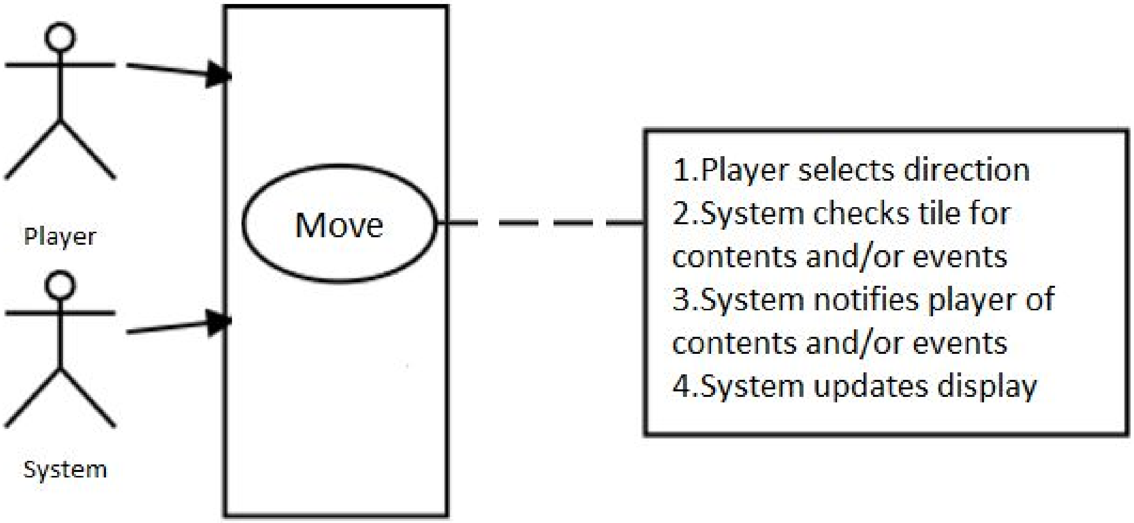
1. Player chooses the save option
2. Player is prompted to name the save
3. System saves all necessay information



1. Player chooses which save to load
2. System rebuilds game from the information in the save

3.1.1.3 City Street Tiles

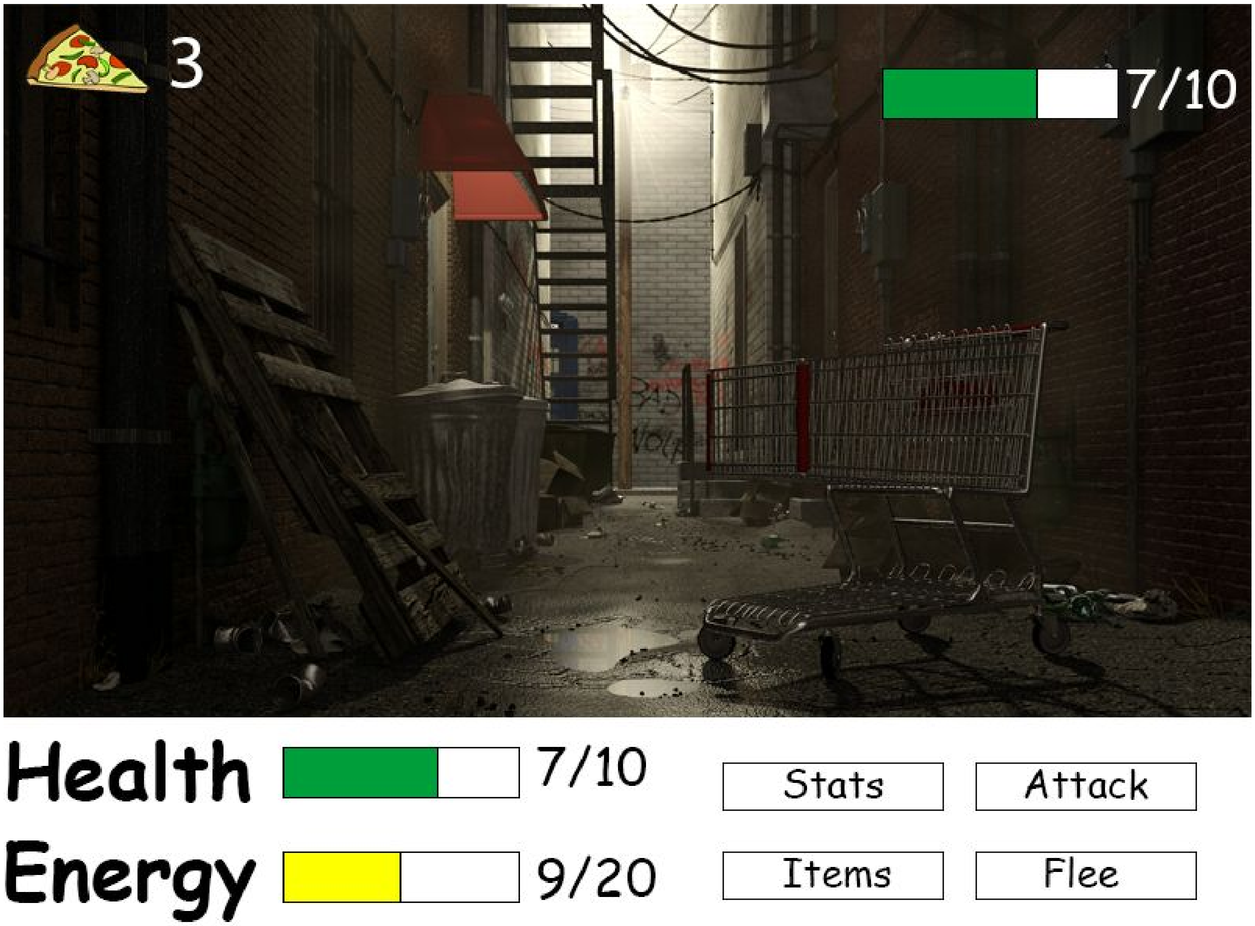
This screen shows different street tiles throughout the map. This is where the player can move around the map when in a tile with an enemy a fight will start



1.Player selects direction
2.System checks tile for contents and/or events
3.System notifies player of contents and/or events
4.System updates display

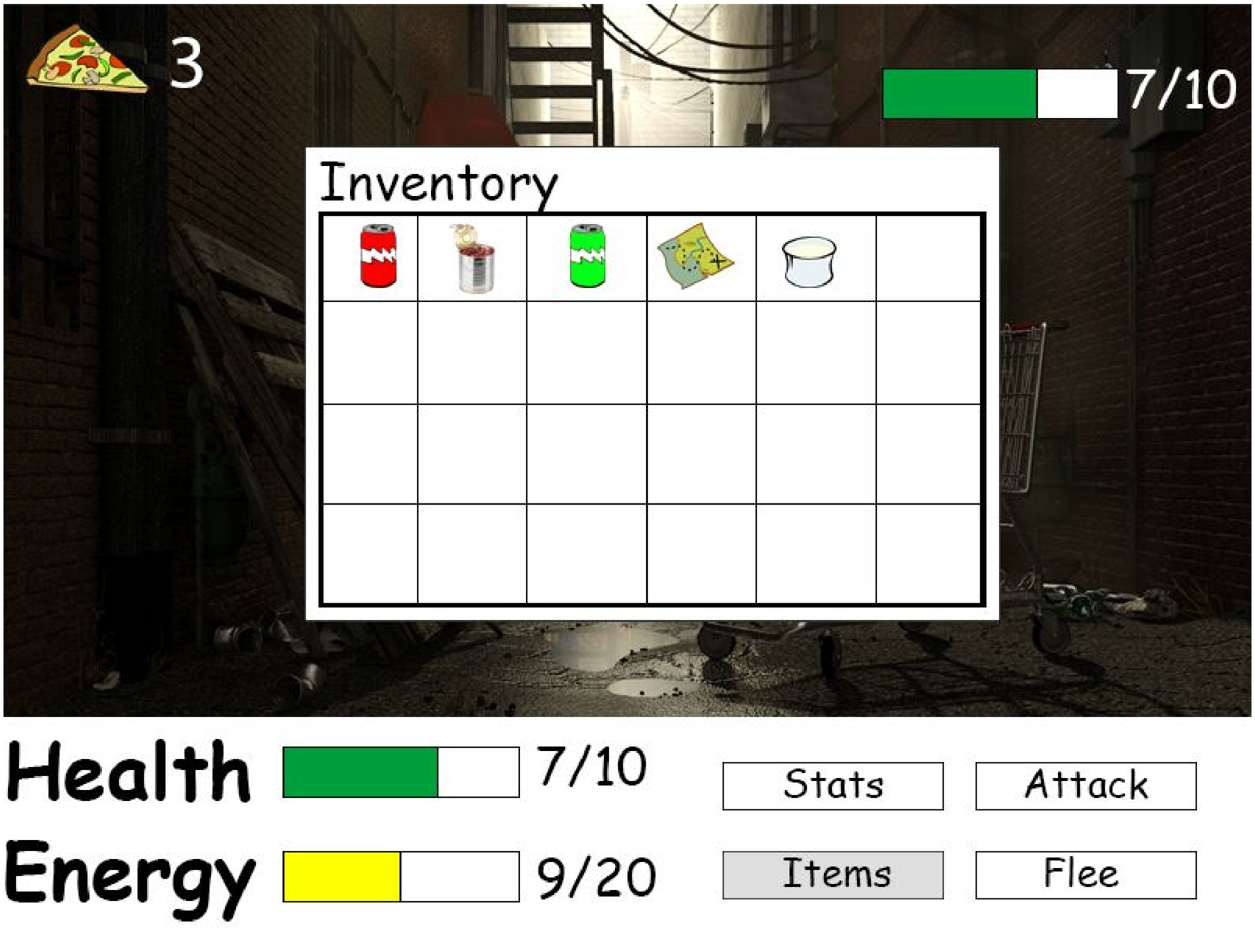
3.1.1.4 Fight Screen

This Screen shows the player and the enemy, their respective health and the options for the player (attack, run, etc..). A message window will also be in place to output messages. Upon victory the City Street Tile will be loaded defeat the player will lose a life token and the street will be loaded.



3.1.1.5 Inventory Screen

This screen shows the items the player currently has acquired, it also has a description of the items. The player can select to use an item from here.When exited the screen returns to the City Tile Screen.



3.1.2 *Hardware Interfaces*

The game will use mouse and keyboard input, it will be able to run on most computers.

3.1.3 *Software Interfaces*

The game will require a recent version of JRE.

3.1.4 *Communication Interfaces*

No network connections are needed for this game it will be completely offline.

3.2 Classes/Objects

3.2.1 Character

3.2.1.1 Attribute Points

The character will gain points from beating enemies and leveling up, these points will be spent on increasing the player’s attributes.

3.2.1.2 Attributes

Attributes are the qualities a player possess. (endurance , speed and intelligence)

3.2.1.2.1 Endurance

Players total health in fights.

3.2.1.2.2 Intelligence

Displays the opponent's tendencies.

3.2.1.2.3 Speed

Time frame to change your move after you see an opponent's

3.2.1.3 Experience

When the character wins a fight they gain experience points that go towards increasing their overall level.

3.2.1.4 Level

The player’s level depends on the amount of experience points they have.

3.2.1.5 Energy

This decreases when the player moves and when they lose a fight, if it reaches zero the game ends. Players energy starts at 100.

3.2.2 Gameplay Area

3.2.2.1 Streets

Each street can hold max one enemy, a hidden item or nothing at all. They are all given unique names to help the player navigate.

3.2.2.1.1 Street Edges

At the edges of each street the player can click to progress to another adjacent street. They can go up, down left or right.

3.2.2.1.2 Street Qualities

Each street can have one or none of the following qualities, slipperiness, poison gas:, pizza whiff.

3.2.2.1.2.1 Slipperiness

Player has a low chance that their next move will not take them to the street they desired.

3.2.2.1.2.2 Poison Gas

Lowers the player's current energy slightly.

3.2.2.1.2.3 Pizza Whiff

Temporary endurance boost.

3.2.2.1.2.4 Thirsty Wizard

When encountered he asks the player for a drink. The player can then choose to give the wizard a can of soda if he has one. Upon giving the wizard soda he gives you a magic pizza potion that turns you into Angry Larry.

3.2.3 Fights

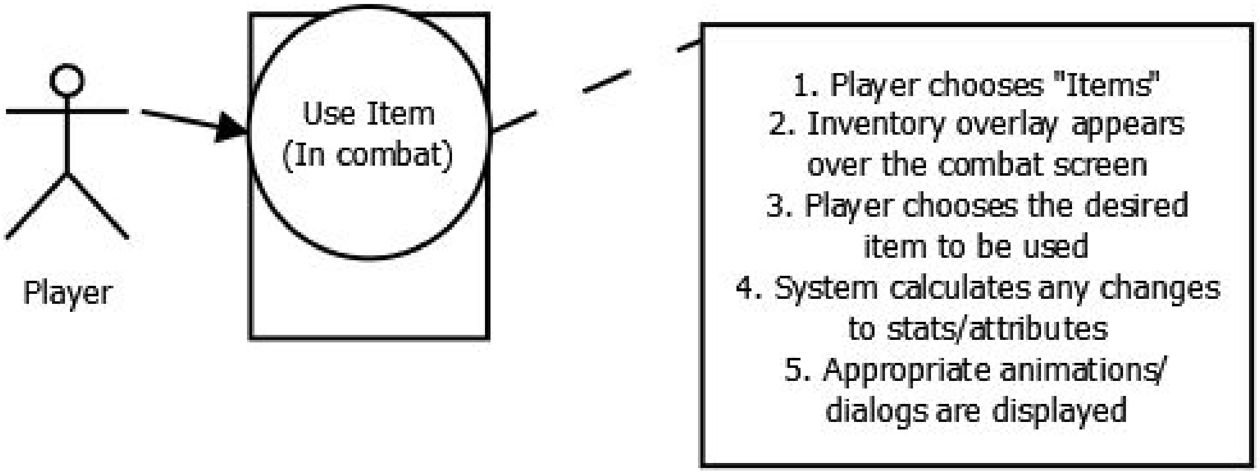
3.2.3.1 Enemy (Anger Crew)

A member of the Anger Crew the player must fight. They all have the same health and attack strength but differing appearance. They are randomly spread out throughout the map.

3.2.3.1.1 Boss (Angry Larry)

Leader of the Anger Crew, he has a higher health and attach strength attribute.

3.2.4 Items



1. Player chooses "Items"
2. Inventory overlay appears over the combat screen
3. Player chooses the desired item to be used
4. System calculates any changes to stats/attributes
5. Appropriate animations/ dialogs are displayed

3.2.4.1 Soda

Replenish the player’s energy.

3.2.4.2 Mom’s Baked Beans

Increases chance of a successful attack.

3.2.4.3 Extra Power Energy Drink

Temporarily increases speed attribute.

3.2.4.4 Mr. Michels Map

Reveals the location of nearby enemies and items.

3.2.4.5 Pizza Slice

Goal is to gather as many as possible, they can be eaten to increase the player's energy when it is running low. They are dropped by enemies when they are defeated.

3.2.4.6 Donair Sauce

Completely refills energy bar.

3.2.4.7 Pizza Potion

Turns the player into an Angry Larry clone. This effect will confuse the Anger Crew members and they will give the player their Pizza Slice without needing to fight them first. This effect wears off after the first Anger Crew encounter.

3.3 Performance Requirements

Must be easy to open and start a game. The game shouldn’t take more than 30 seconds to start after it is selected.

3.4 Design Constraints

The game must be able to be downloaded from a website and played. It must be compatible with current versions of Windows, Mac OS and Linux. Must have simple graphics.

3.5 Software System Attributes

3.5.1 Reliability

The game should fail less than twice per 100 plays. If this occurs an error message will display.

3.5.2 Availability

The game can run on all recent computers.

3.5.3 Security

Save files will not be password protected.

3.6 Other Requirements

The game must market Mom and Dad’s Pizza Pad from start to finish. At the same time it must diss other pizza places represented in game.

3.7 Combat System

- When a player encounters a member of the Anger Crew, they begin combat with them

3.7.1 Combat Screen

3.7.1.1 Player health

Usually, the player only uses energy to move around the map, health is visible only when in combat or in the menu system

3.7.1.2 Energy level

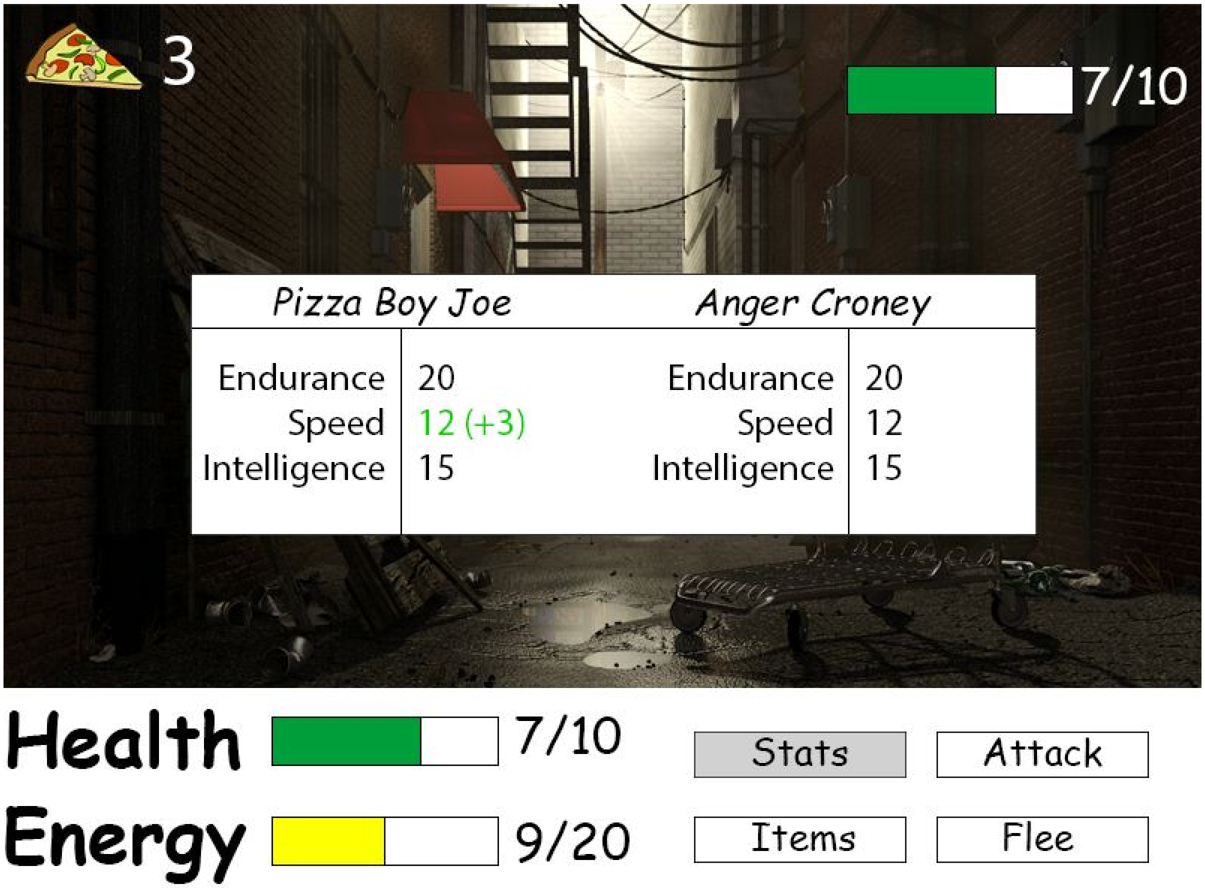
This is the same energy bar as used to do movements. It can also be used in combat to use attacks (Player is unable to attack when energy level reaches zero). The double use of this statistic allows for players to have multiple play styles.

Example:

- If a player finds a map, they can use it to their advantage to avoid fights if low on health, allowing them to find items to regain it
- Players can also try to fight as often as possible without wasting energy on movements
    - Players will be able to defeat enemies more quickly if they can have more energy saved for fighting, less used on movement

3.7.1.3 Stats

Selecting this option will show an overlay on the fight-screen GUI that shows the player their attributes



Pizza Boy Joe		Anger Croney	
Endurance	20	Endurance	20
Speed	12 (+3)	Speed	12
Intelligence	15	Intelligence	15

Health 7/10

Energy 9/20

3.7.1.3.1 Endurance

Gives the player a higher maximum energy possible, so that they can fight and move around the map for longer without needing to find an item to regain energy

3.7.1.3.2 Speed

The player has a chance to dodge attacks, based on their speed attribute

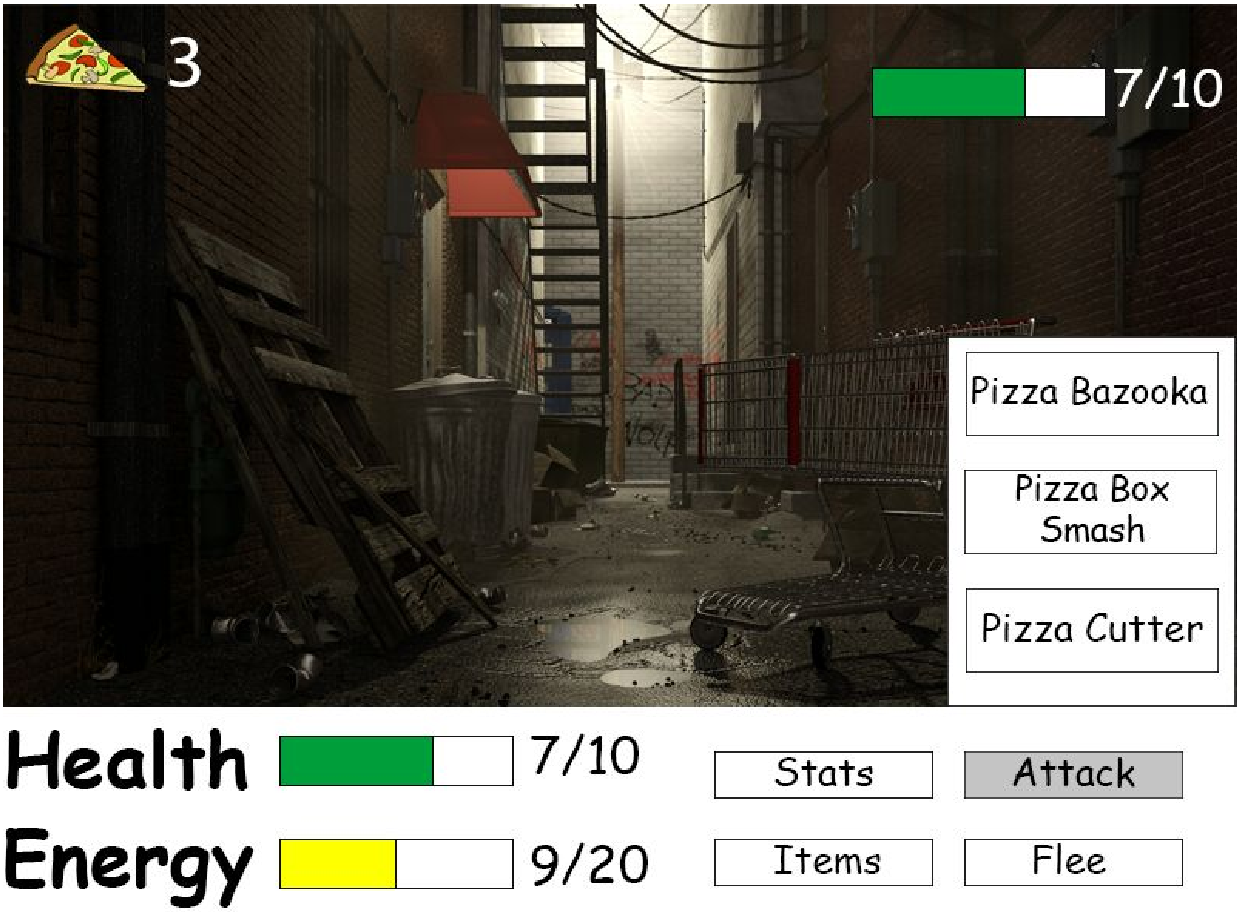
3.7.1.3.3 Intelligence

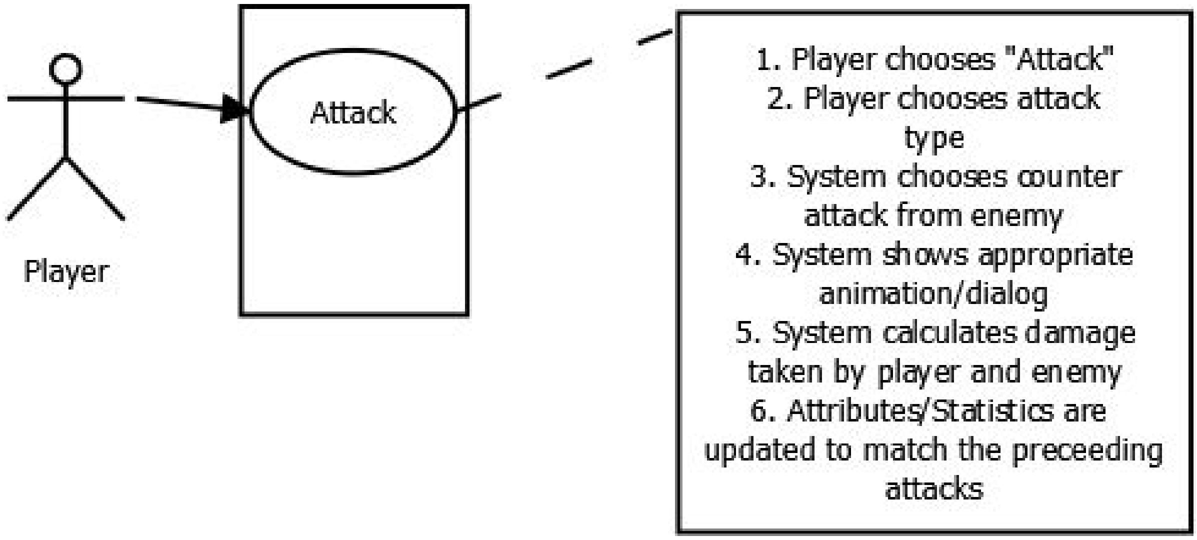
This allows a player to have a greater chance of the enemy choosing an attack that is weak against theirs (see attack types section)

3.7.1.4 Attack

This brings up a small menu overlay that gives a choice of attacks

- Pepperoni Bazooka
- Pizza box smash
- Pizza cutter



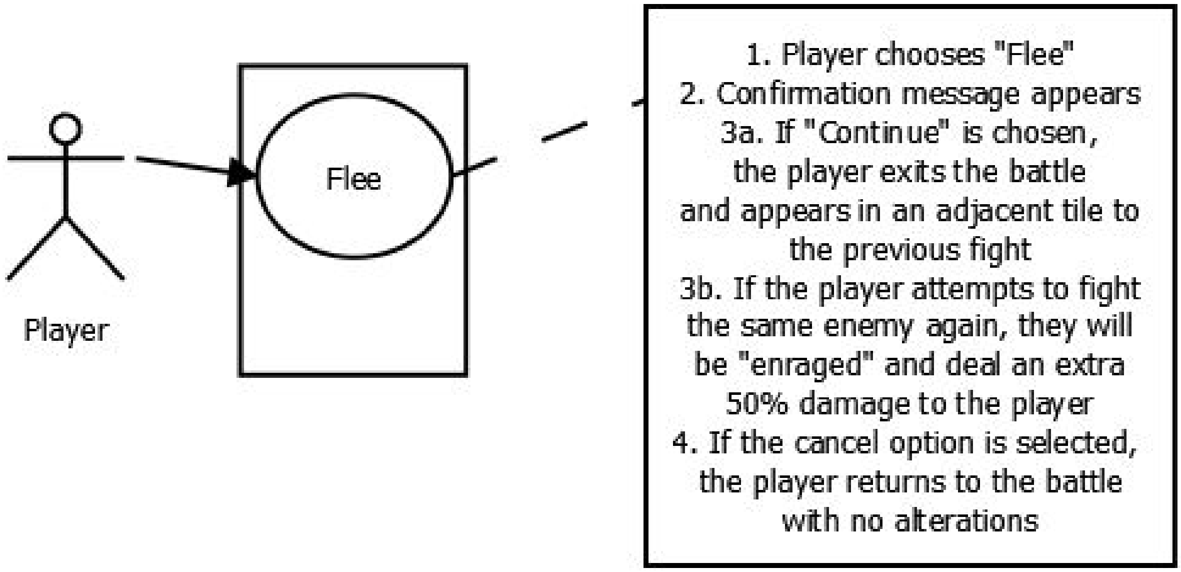


1. Player chooses "Attack"
2. Player chooses attack type
3. System chooses counter attack from enemy
4. System shows appropriate animation/dialog
5. System calculates damage taken by player and enemy
6. Attributes/Statistics are updated to match the preceeding attacks

3.7.1.5 *Flee*

- If a player is in danger of running out of health, it is more beneficial to flee the battle beforehand than to let the character lose the fight

- Fleeing from an Anger Crew member enrages them, and



1. Player chooses "Flee"
2. Confirmation message appears
3a. If "Continue" is chosen, the player exits the battle and appears in an adjacent tile to the previous fight
3b. If the player attempts to fight the same enemy again, they will be "enraged" and deal an extra 50% damage to the player
4. If the cancel option is selected, the player returns to the battle with no alterations

3.7.1.6 *Attack Types*

The player chooses an attack type and the enemy rolls for an attack out of the same grouping. The attacks work in a rock/paper/scissors type format. Damage advantage or disadvantage works in the following way.

Table 3.1

Player Attack	Enemy Attack	Pepperoni Bazooka	Pizza Box Smash	Pizza Cutter
Pepperoni Bazooka		-50% (of remaining health)	100%	300%
Pizza Box Smash		150%	100%	50%
Pizza Cutter		50%	150%	100%

3.7.1.6.1 Attack Bonus Attributes

Each attack also has a unique bonus attribute.

3.7.1.6.1.1 Pepperoni Bazooka

Successful attack does double damage, but halves your own intelligence for the remainder of the battle. If both players use the Pepperoni Bazooka, they both lose half of their remaining health but the player is given a lump of radioactive pepperoni.

3.7.1.6.1.1.1 Lump of Radioactive Pepperoni

While the player has a Lump of Radioactive Pepperoni (LORP) in their inventory, it decays the players health by 1 per turn. Once the player uses the item, it transforms them into a Giant Ultra Mutant Pepperoni (GUMP). While in GUMP state, the player takes 3 damage per turn, but also attacks for 200% base damage. This effect ends after the current fight.

3.7.1.6.1.2 Pizza Box Smash

Successful attack grants a small buff to the player wherein their speed is increased for the next turn

- This creates a strategy where the player can consecutively use Pizza Box Smash and continue to receive the speed buff as long as each hit is successful, which is more likely with the speed buff

3.7.1.6.1.3 Pizza Cutter

Successful attack leaves the enemy bleeding so that they take a small extra amount of damage for the next 3 turns, regardless of if proceeding attacks land

3.7.2 Victory

If the player defeats the enemy:

- there is a random item dropped from the enemy
  - Stronger enemies have a higher % chance of dropping higher tier items
- The pizza slice counter in the top left is incremented by 1
- The area in which the enemy was encountered no longer triggers a fight when the player moves to that area
- The player is given 1 attribute point, which can be put into any of the 3 supported attributes

3.7.3 Defeat

If the player is defeated by the enemy:

- They respawn at the adjacent tile to the fight
- They lose 10% of their maximum energy
- They drop all of their earned items
- They drop 1 pizza slice

3.7.4 Scoring

- The player is scored primarily based on the number of Pizza Slices they were able to retrieve from Anger Crew members
- Each member defeated gives 1 slice. Defeating Angry Larry gives the player 3 slices
- To differentiate high scores, the player with the highest health at the end of the round is given priority on the leaderboard when players have identical scores

4. Management Plan

4.1 Introduction

We are a software development firm based out of Wolfville NS, focused on building the highest quality software for all of our clients. Our team is made up of four computer science students at Acadia University.

4.2 Team Organization

Our team consists of student members Scott Barnett, Jimmy Flemming, Liam MacKinnon and Martin Main. Our team is organized as follows:

Team Member	Role	Responsibilities
Scott Barnett	-Customer side documentation -Non-combat system design -Quality Control	Customer side documentation: -Responsible for providing documentation and product guides that will be easily readable for non-technical readers Non-combat system design: -Responsible for the design components of the non-combat systems of the game, including the user interfaces, map, and menu (co-leading with Liam MacKinnon) Quality Control: -Ensure all software and documentation components are of a high standard of quality, and inform team accordingly of required improvements
Jimmy Flemming	-Combat systems -Documentation Diagrams	Design of Combat systems: -Responsible for the design component of the combat system of the game, including the rock-paper-scissors style fighting system, and determining the types of attacks and any other specifics of the combat system Documentation Diagrams: -Leader in diagrams for all documentation; responsible for consistency between all graphs and diagrams

Liam MacKinnon	-Communications with Client -Meeting Minutes -Non-combat system design	Communications with Client: -Responsible for all client communications, including clarification of design and requirement details Meeting Minutes: -Responsible for taking minutes at all the team meetings, and making them available on the project website Non-combat system design: -Responsible for the design components of the non-combat systems of the game, including the user interfaces, map, and menu (co-leading with Scott Barnett)
Martin Main	-Project Lead -Documentation editor -Webmaster	Project Lead: -Responsible for bringing together the various pieces involved in each deliverable, and making sure the deliverable is complete and satisfies all requirements. Responsible for leading team meetings and producing meeting agendas Documentation editor: -Responsible for editing and formatting all documentation deliverables Webmaster: -Responsible for all updating and maintenance of the project website, including posting all relevant documentation and deliverables

4.3 Managerial Process

The project team meets weekly at a minimum to discuss updates on the current deliverable being developed, and planning for the future deliverables. The deliverable with the next immediate deadline will be the top priority for the team. More meetings and work sessions will be planned by Project Lead as required.

4.4 Project design model

The project design will follow the iterative process model, in which the development will follow a cycle including requirements, design, implementation, and testing. The primary goals will be implemented first, followed by secondary and optional features.

Project Schedule

Deliverable	Description	Deadline Date
S1	Requirements Specification and Management Plan	February 8, 2016
S2	System Overview Presentation	March 2, 2016
S3a	System Design document	March 9, 2016
S3b	Test Plan Document	March 9, 2016
S4	User Manual	March 21, 2016
S5	System and System Demonstration	April 6, 2016
S6	Supplier Project Evaluation	April 11, 2016

4.5 Risk Management

Some of the possible risks involved in this project include:

4.5.1 Failure to complete all requirements of deliverable on time

A risk management retirement plan is to add emergency team meetings before the dates of the deadline, and prioritize requirements to make sure the most crucial components are completed first.

4.5.2 Meeting conflicts with various academic-related events

A risk management retirement plan is to either:

1. Reschedule the meeting to a time that works for everyone (Martin Main is responsible for organizing this)
2. Take good notes during the meeting to give missing member(s) all required information

4.5.3 Lack of technical knowledge of certain programming languages, software patterns, or related technologies

A risk management retirement plan is to be forward thinking in our design stage, and have relevant team members read over required technical documentation before the time of implementation.

4.6 Design Phase Model

The design phase will consist of first creating a design document, and then delegating team members to various parts of the game system to design, implement, and test for the first iteration. Please refer to the team organization document for various design role delegations.

4.7 Costs

4.7.1 Time

The time costs for this project will not exceed 35 man-days, at 8 hours / day, or 280 man-hours.

4.7.2 Materials

The materials required for this project will include:

- Computer systems available for all team members
- Operating systems and development environments that meet the needs of the project specifications
- Internet access for group collaboration efforts including concurrency control, access to the project website, as well as client communications.

4.8 Client Communication Expectations

During the course of the development, there will be certain expectations for the client (Mom and Dad's Pizza Pad) in communication with the development team.

4.8.1 Review of relevant documentation

We request that all relevant documentation, including the current document, be reviewed by the client. We request that all questions and comments be recorded and reported to the development team as soon as possible, and that any and all changes in the requirements be communicated to the development team as soon as possible. All client communications can be directed to our client communications manager Liam MacKinnon through email at 114393m@acadiau.ca

4.8.2 Response to queries posed by the development team

We request that all questions posed to the client from the development team be answered in a timely fashion, appropriate to the questions being asked

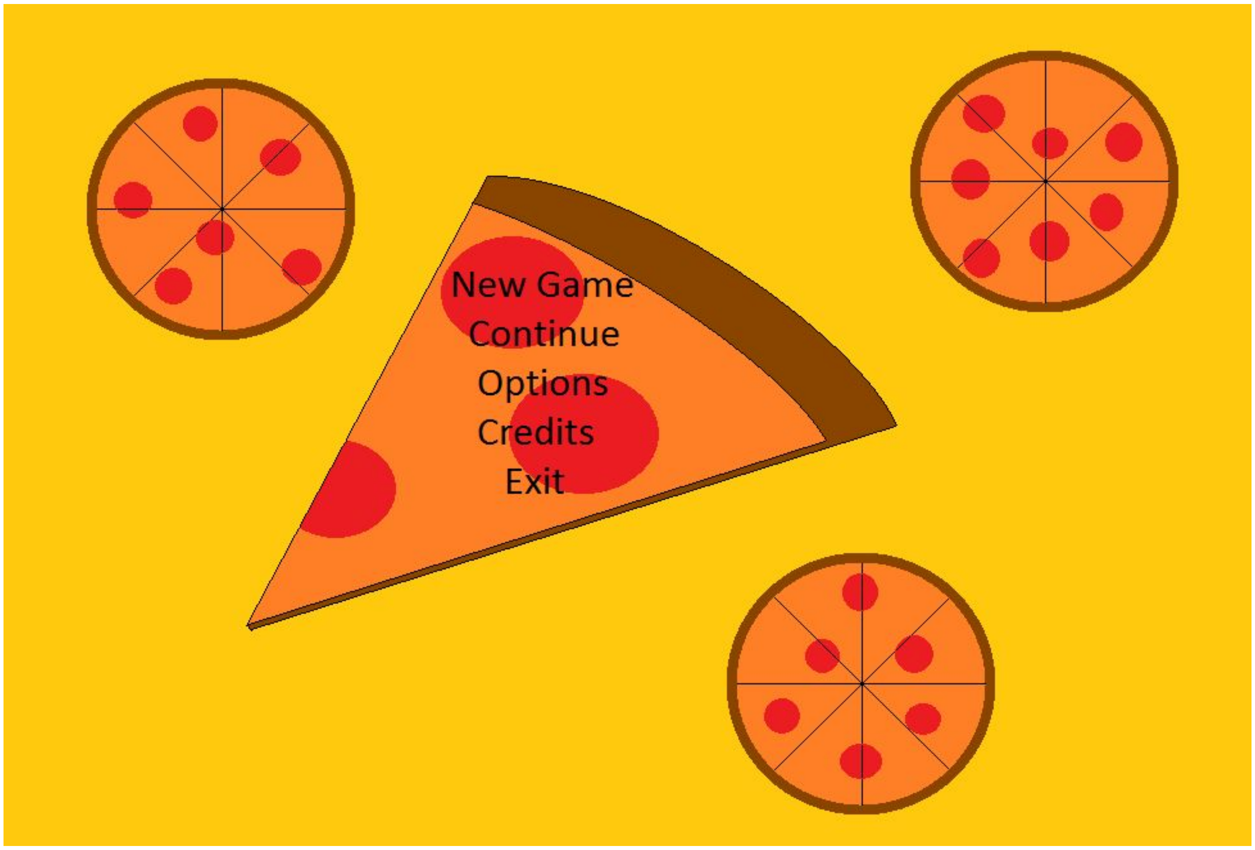
4.8.3 Attendance to required meetings

We request that at least one representative from the client is present at all meetings to discuss the project at the time and location as agreed upon with the client.

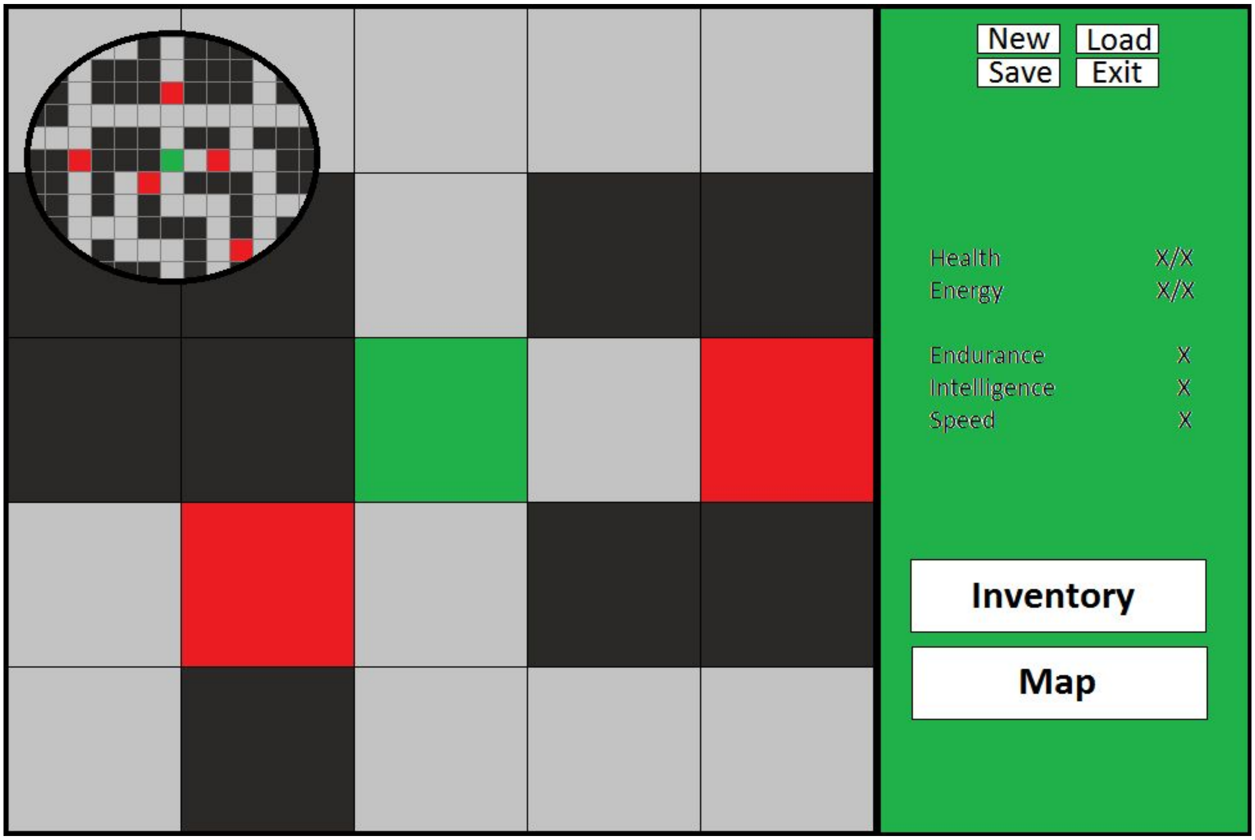
Summary

This document was designed in order to help the reader to understand in detail the workings of the game *Mom and Dad's Pizza Pad's Pizza Adventure* and was written and edited by the development group *Team C* as a part of a software engineering class at Acadia University in 2016. This document is meant to be read by the client, Mom and Dad's pizza pad, the development team, and by any future developers that may work on this project. If there are any mistakes, misinformation, typos, or offensive content appearing in this document, please feel welcome to contact our quality control expert Scott Barnett by email at 123218b@acadiau.ca

5. Appendix A



5.1 An example of the GUI for the main menu



5.2 An example for the map screen in the game