

Wolfen Blood



Ericka Zimmer

Final Report

December 8, 2009

## Table of Contents

Application Development .....	4
Concept: .....	4
<i>General</i> .....	4
<i>Personal Creation</i> .....	4
<i>Challenges</i> .....	5
Motivation: .....	6
<i>Personal</i> .....	6
<i>Professional</i> .....	6
<i>Audience</i> .....	6
Resources: .....	6
Preliminary Software Requirements Specifications:.....	8
Vision: .....	10
Scope: .....	10
System Design and Architecture .....	11
<i>Planned Project</i> .....	11
<i>Software Components</i> .....	11
Implementation:.....	12
Known Bugs and Errors .....	12
Test Plan.....	13
<i>Purpose</i> .....	13
<i>Features to be tested</i> .....	13
<i>Features not to be tested</i> .....	13
<i>Approach</i> .....	13
<i>Suspension criteria and resumption requirements</i> .....	13
<i>Environmental Needs</i> :.....	13
<i>Schedule</i> .....	14
<i>Acceptance criteria</i> .....	14

<i>Roles and responsibilities</i> .....	14
Test Cases.....	14
Unit and Integration Testing .....	16
<i>MudRoom</i> .....	16
<i>MudMap</i> .....	16
Software Demonstration.....	18
About the creator:.....	19
Executive Report.....	20
Week 1: .....	20
Week 2: .....	21
Week 3: .....	22
Week 4 & 5: .....	24
Week 6: .....	26
Week 7: .....	28
Week 8: .....	29
Week 9.....	31
Week 10: .....	33
Week 11 & 12: .....	34
Week 13: .....	35

## **Application Development**

### **Concept:**

- *(Revised September 22, 2009)*

- *(Revised December 7, 2009)*

### ***General***

The goal of this project is to complete a foundation for a text based role playing game. A role playing game (RPG) is a game where the player takes on the role of a fictional character and determines his or her actions based on that fictional character's personality. In the RPG there are some rules, but the player still has freedom in what choices to make in game. Text based RPGs are RPG where everything including environment and fighting is depicted through text. The user enters in commands via the keyboard to move, fight, and interact with non-player characters.

### ***Personal Creation***

I want to create a text based RPG foundation. That is to say, my creation will not be the game itself, but the ability for others to make their own game without having to worry about the coding themselves. Therefore I will make the basic needs for a text based RPG and a set of commands for creating the parts to that game. There would be limited access to normal players who start playing the game and only those with the permission may create additions to the game.

What is to be created by the end of the class is the ability to make new rooms and handle players correctly. If time permits, more design of the world will be created as well as items, equipment, etc. as well as possibly be expanded to an online multiplayer game. Once this game is complete enough to distribute to others, they will also have access to the source code so that they can make their own versions. All that is asked in return is the chance to play whatever they create as well as a citation to the original creation to give credit where credit is due.

## ***Challenges***

The challenges in this project are few but trying. The first challenge would be making something from scratch that I as the creator have never done before. I have created many small programs, but nothing with so many classes and as much code that will be done. Second is figuring out and researching all of the information I need. This will be the hardest part. It has always been weak of mine as I have a hard time leaning by reading and edgy to start programming from the get go. Last, if I get to network programming, learning it will take a large amount of time.

**Motivation:*****Personal***

- (revised December 7, 2009)

When I was in high school, I used to enjoy playing multi-user dungeons (MUD) which are text based games that more than one person can get on and their characters may interact with each other. The first time I ever played a MUD was in the 10<sup>th</sup> grade. My friend came over to hang out and she told me about the MUD called “Cities of Mdhoria” (CoM). She told me I should make a character and set me up to do so. Being the person she is, she played around with my new character until she left, leaving me in the middle of the main city and didn’t really teach me to play. However, I grew to enjoy the game play and the ability to imagine the way things look in the game. If I felt that it was within my reach by the end of this class, I would be creating a MUD which is what my game probably will evolve into in the future.

***Professional***

One of the possible jobs I am thinking about is game design. I’m sure this would help me if I were to choose that path. My entire life I have thought of stories in my head that I believe would be nice to share, but I could never be any good at writing, so I find it best to make RPG games and get paid for it.

***Audience***

This project audience would be anyone who wants to use my program to make a text based game themselves, programmers who want to expand on my code for themselves or those who would want to play the games that could be created through this.

**Resources:**

- (revised September 22, 2009)

- (*revised October 13, 2009*) *Removed book*

- (*revised December 7, 2009*)

- Mud Game Programming by Robert Penton
- G++ compiler in Linux
- Emacs code editor

## **Preliminary Software Requirements Specifications:**

- *(Revised October 20, 2009) Changed to Linux from windows*

Number: 1

- Statement: Must run Linux
- Evaluation Method: Everything runs fine on a Linux environment
- Dependency: None
- Priority: High
- Requirement revision history: Changed from Windows to Linux (Oct 20. 09)

Number: 2

- Statement: Some foundation made for the text based RPG
- Evaluation Method: When the game runs there is enough to see that there is something  
Resembling a text based RPG
- Dependency: Runs on Linux and at least an environment creator and some commands
- Priority: High
- Requirement revision history: Added (Nov 4. 09)  
Changed from MUD to text-based RPG (Dec 7. 09)

Number: 3

- Statement: The program must be easy enough to understand so that anyone can build off  
of it, even with little to no programming skills. (plain text commands)
- Evaluation Method: Let others use it and see how they like it
- Dependency: Something is built
- Priority: High
- Requirement revision history: More detailed than before (Nov 10. 09)



Number: 4

- Statement: A small demo of the text-based RPG
- Evaluation Method: Let people log in and play around
- Dependency: A foundation for building has been made
- Priority: Medium
- Requirement revision history: Changed from MUD to text-based RPG (Dec.7 09)

Number: 5

- Statement: A web space is provided for any user to access the program and play as well as having access to see source codes
- Evaluation Method: There is a website that anyone can go to and the download is there
- Dependency: A full engine is built
- Priority: Low (only done as time permits and most likely completed when class is over)
- Requirement revision history: Changed to Req 5 from 6 (Dec 7, 09)

**Vision:**

- *(revised October 13, 2009)*

- *(revised December 7, 2009)*

At the end of this project, there will be a foundation that can be used for many text-based RPGs to come. Since the final product will be fully accessible to all those who wish for it, many more people will be inclined to use it and create their own worlds for people to enjoy. It will also be free source code for fellow programmers who wish to expand upon it. It is hard to find and free and open source code for these sort of games and some people want payment for it. Also, with its ease of use to create different games, more people will be encouraged to look at it and then spark interest in using it.

**Scope:**

- *(revised October 13, 2009)*

- *(revised December 7, 2009)*

The foundation for the text-based RPG will have coding set up for making areas, exploring, and handling new characters and saving said data. The main focus is on the ability to create one's own game. To be more specific, this will not be a fully implemented game at the end of this project. Players will not necessarily be able to have quests or really have fun creating a character or fight monsters. This will only be coded so that future creators may be able to make their own worlds.

## System Design and Architecture

### *Planned Project*

In this project there will be a working basis for creating a game. By logging in with the special permissions, anyone will be able to create their own environment. There will be a function in MudController.h that handles several different types of inputs that are intuitive commands. For example, if someone were to type the command “create room north” the function would properly call the correct functions in order to create a room north of the current location.

### *Software Components*

#### *- (Class List)*

- MudSocket.h //For dealing with telnet and network programming
- MudLogin.h //For handling Player Login and accessing database
- MudController.h //For handling user input
- MudRoom.h //For creating and dealing with rooms in a map
- MudMap.h //For creating and accessing map
- MudItem.h //For creating items
- MudNpc.h //For creating Non-Player Characters
- MudStats.h //For stats of the player
- MudSpells.h //For creation and use of spells
- MudSkills.h //For creation and use of skills

#### *- (Database List)*

- Users.db //Keeps track of users and their stats
- Items.db //Keep track of items created

## **Implementation:**

- README.txt
- MudLogin.h / MudLogin.cpp //Not currently working
- MudRoom.h/MudRoom.cpp //For use of creating a mud environment
- MudMap.h/MudMap.cpp //For use of creating a mud environment
- MudController.h/.cpp //For sorting out text input
- MudPosition.h //For character location
- Makefile //For compilation
- Main.cpp //For testing

## **Known Bugs and Errors**

- *(revised December 7, 2009)*

- When creating a long description for a room, there is a space at the beginning of the paragraph.

## **Test Plan**

- (revised November 17, 2009)

### ***Purpose***

The application being tested is the text-based RPG that has been created over the semester. There are still things that need to be checked for usability and correctness.

### ***Features to be tested***

- Requirement 1: Must run Linux
- Requirement 2: Some foundation made for the text-based RPG
- Requirement 3: Easy to use
- Requirement 4: A small demo of the text-based RPG

### ***Features not to be tested***

- Requirement 5: Web space

### ***Approach***

Have others install the needed content to run the game and have them play off of their computers, as long as they are running a Linux environment.

### ***Suspension criteria and resumption requirements***

Suspension criteria: are the conditions that, if satisfied, require that the test be halted.

- Something unexpected comes up

Resumption requirements: are the conditions that are required in order to restart a suspended test.

- Computer crashes

### ***Environmental Needs:***

A working computer is needed that can run in Linux. Also, mysql must be downloaded and installed as well as mysql++ from this website:

<http://tangentsoft.net/mysql++/>

### ***Schedule***

- Requirement 3: Between Nov 11 and Nov 16
- Requirement 2: Starts Nov 11, continuous testing as more code is made
- Requirement 1: Nov 17
- Requirement 4: Dec 8

### ***Acceptance criteria***

Any objective quality standards that the software must meet, in order to be considered ready for release. This may include things like stakeholder sign-off and consensus, requirements that the software must have been tested under certain environments, minimum defect counts at various priority and severity levels, minimum test coverage numbers, etc.

- All tests pass

### ***Roles and responsibilities***

- Ericka: Find people to try using program
- Ericka: Test current foundation
- Ericka: have other Linux users use program on own computer

### **Test Cases**

- (*revised November 17, 2009*)

Test1: Foundation test

- Requirement 2
- Preconditions: Code can be compiled
- Steps:
  - Run the program

- Expected results: The running program acts like an actual text-based RPG with basic commands

#### Test2: Linux Test

- Requirement 1
- Preconditions: Other users have downloaded the needed software and saved the project to a folder.
- Steps:
  - Compile the code for the program
  - Run the program as is
- Expected results: Full compilation and program is exactly like on creator computer

#### Test3: Ease of use

- Requirement 3
- Preconditions: The game is compiled and running
- Steps:
  - Person using program enters in sample commands
- Expected results: Person easily understands commands and could use them without much help.

#### Test4: Demo created

- Requirement 4
- Preconditions: The game is compiled and running
- Steps:
  - Start up the game
  - Explore the abilities of the program

- Expected results: Game runs smoothly and there is some environment

## **Unit and Integration Testing**

### ***MudRoom***

#### Unit Test 1: ReadRoom

1. Start up the program to test the function when it loads all saved rooms
  - If it works, previously saved rooms will show up

#### Unit Test 2: Room accessors and mutators

1. Type “edit room” while in the program to enter room editor
2. Use every option given in the edit mode to test most of the functions.
3. When exiting edit mode, the program writes to file, testing the SaveRoom function.
  - If everything works, the room will save the values entered and show the next time the program is run

#### Integration test 1: MudRoom

1. Type “look” to use the display function to test interaction for all accessors
2. Use the read room to test interaction between mutators at start up of program
  - If it works, the description of the room is displayed on start up.

### ***MudMap***

#### Unit Test 3: LookDir

1. Type “look” and then a direction (north, east, etc.) to test
  - If it works, the short description of the next room pops up



#### Unit Test 4: PrintArea

1. Type “look” to use the display function to test interaction for all accessors
  - If it works, the description of the room is displayed

#### Unit Test 5: Move

1. Type a direction or the first letter of a direction.
  - If this works, the user is moved to the new location and the description of the next room is displayed.

#### Unit Test 6: CreateRoom

1. Type “create room” and then a direction
  - If this works, the user is entered into the edit mode for the room in that direction and the changes have been made to possible exits.

#### Unit Test 7: Edit room

1. Type “edit room”
  - If working, the program enters edit mode for current room.

#### Integration Test 2: MudMap

1. Start up the program and test different commands that will be sent through the MudController class to test interactions between all functions of MudRoom and MudMap.

## **Software Demonstration**

In the first demo, it goes through and shows what the game can do as of right now as far as logging in and using the room editor and creator.

<http://www.youtube.com/watch?v=LekP5X9N1Ww>

## About the creator:



The creator, Ericka Zimmer, has been passionate about this project for a long time. Before this class started, there was little time for her to do such a project, but the desire was still there. Having a strong background in C++ programming, Ericka has coded most of the project in that language. She is currently employed at Berea College as a teaching assistant for the math and computer science department, and is also the Math/CS lab coordinator.

## **Executive Report**

### **Week 1:**

This first week was focused on researching information on MUDs and how to make them. I was given an electronic book to help me titled MUD Game Programming by Ron Penton. I spent a total of about 5 hours reading through it and other internet sites. I also looked at the differences between MUCKs and MUSHs compared to a MUD and what, in my opinion, are the pros and cons of each. I only looked a little into programming languages for MUDs, but I only found personal opinions and not so many facts about them.

Along with research, I spent some time trying to come up with a good logo and name. I started by looking at other logos for different things. I finally decided I wanted a wolf, so I drew a picture and scanned it in. Afterwards, I used Inkscape to clean it up and I was really pleased with it. Off of the logo, I came up with a name. I figured the name should have wolf in it as the logo is a wolf head, and thus the name Wolfen Blood.

I plan in this upcoming week to soon decide on the programming language and then I will read more about the first steps in building a MUD and then hopefully start coding.

**Week 2:**

This week I spent more time looking up information. I read through more of the book MUD Game Programming and many internet sites and decided my coding would be done in C++, then proceeded to download Code::Block for future programming to take place. I also looked for more reading and bought a book online to help me as well. Many of the internet sites proved to be inadequate. One site said it had the complete guide to creating a MUD and basically told me I should get an internship to teach me. I did find one site that was interesting. It was a site to help you create your own MUD without the programming. It was interesting and I played around with it for a few hours, but I would have had to pay to get it working.

I hope by next week I can have started an outline of my program and know more of the steps I have to do to get it working. I wanted to start programming, but then I thought I should know what I have to do before I code anything that is inefficient.

### **Week 3:**

Wolfen Blood



Ericka Zimmer

Report 3 of Week 3

September 29, 2009

This past week I have done much research on how other people have made similar projects to mine. I found a website < <http://www.rdxgames.net/projects/wrathlands/index.html>> that has screen videos of a person creating a text based RPG. This process deals with a single user and in the end there is little freedom to do what you want. It was like games where you are given a specific character to play and you cannot choose how you wish to play. This can be useful when looking at leveling systems along with battling. I knew right away it is not what I want, but I spent a few hours looking at what might be useful.

My book I ordered came in only yesterday and I have spent only two or three hours looking through it. It explains MUD creating in LPC (Lars Pensjö C) which is quite similar to C code. MUDs programmed in this code are LPMUDs. This book tells a lot about how to program once you have become a “wizard” on an already existing LPMUD and nothing so far on creating a MUD from scratch. Most likely I will not be using this book.

On a brighter note, I have found some source code to look through and I will be doing that later this week. Also, I will focus more on the other book I have, as it seems to be more

useful. I hope that this upcoming week, I can get a design of the project down and actually start coding.

## **Week 4 & 5:**

Wolfen Blood



Ericka Zimmer

Report 4 of Week 4&5

October 13, 2009

These past few weeks I have really started to feel like I am further behind than I should be. I can find no internet sources to help me with this. One site said it was the complete guide to MUD programming, but it actually stated that I should play MUDs and learn some programming from the creators. From what I have seen, creators keep refusing to let people see their code. The one electronic version of a book I was given was supposed to come with a CD, but I do not have access to what is on it and that is supposed to have a lot of what I need. The only way I could really get it is by buying the actual book, and I do not have the money to do that, and will not have it until this Friday afternoon. Then by the time I get the book, I am supposed to have been programming for over a week. I really want to start programming and I feel like I am never going to get there.

I did get some source code, but I can't make sense of a lot of it. There are several different headers and I don't know which one is to start with as it did not come with documentation. I believe that once I do start coding, I am going to write down each and every step I do in a document.



I also spent some time talking to some friends online who want to create a MUD and they need a coder to help them with their creation, so I feel more like I am actually doing this for someone like I had hoped.

I spent 2 hours talking to people online about the project, 5 hours trying to look stuff up online and 4 hours reading through the book I have. Total time spent on project: 32 hours

## **Week 6:**

Wolfen Blood



Ericka Zimmer

Report 5 of Week 6

October 20, 2009

Last week I was ill and did not finish typing up important sections in my paper due to the constant sleeping, so this week I went back and edited the Application Development Section, So now there is actual information there.

I also found it crucial to look up network programming and telnet, as I will need to use it later. I was reading a tutorial from this website:

<http://beej.us/guide/bgnet/output/html/singlepage/bgnet.html>

Also I started working on the MudLogin. This needs to deal with a database; therefore I was reading a lot on SQL. I was looking into using SQLite3, but this does not seem to be working well, so I may have to switch over to mySQL.

After banging my head around network programming and databases, I decided to work on something that I know I could do. I began coding for the MudRoom, so that I could test it and then start commands in MudController later. Currently I am having trouble with getting my most recent version of MudRoom to compile. There is a strange error saying that the class is not used properly.

I have been talking with some people who want to create a Mud, and they are looking into getting a host for their mud, so I will have a place to upload code online. I will at some point

in my life get my own server for my completed code and my own Mud, but I do not have the money to do it now.

- 2 hours on editing paper
- 3 hours looking up network programming
- 3 hours working on MudLogin
- 4 hours working on MudRoom

Total hours: 42

For next week, I hope to have MudRoom done and working as well as MudMap. If I can, the login will also be working and I will get more research done on sockets.

## Week 7:

Wolfen Blood



Ericka Zimmer

Report 6 of Week 7

October 27, 2009

This past week I spent a lot of time getting my MudRoom to work correctly. There were several errors with the reading in and out from a file, but I managed to get it working well. Then I began work on a MudMap class. There were several issues when trying to save and then load all of the rooms. After a long time I realized my problem was dealing with the stringstream properly. The MudMap is not fully implemented yet, but what is implemented seems to be currently working.

- 3 hours editing and fixing MudRoom.cpp
- 4 hours creating and editing MudMap.cpp

Total week hours: 7

Total Semester hours: 49

This upcoming week I hope to do all that I wanted to accomplish last week, including creating a MudController class and getting login to work with mySQL. If I can get that done, I will start work on socket programming along with creating an item class MudItem.

## Week 8:

Wolfen Blood



Ericka Zimmer

Report 7 of Week 8

November 3, 2009

After fixing up a few issues with the MudRoom and MudMap, I began work on a MudController to handle input from the user. There were a lot of errors that kept coming up, mostly one that had a lot of nonsense and then said there was a problem with `_start`, which isn't any of my work. I found that the problem was with the linking to a main. So all I had to do was include it in the main and then compile everything together.

Once I get the socket for telnet working, I will have to change this and probably edit a lot of my other code as well. For this reason, I want to get it and the login working properly before I continue, so that I will not have a ton of files to edit in the future. I need to make sure people can log in and keep all of their stats. I have spent a lot of time this week trying to find the correct SQL download I need, and possible a wrapper to make the sockets easier to work with. At least then I might have some hope of getting this part done on time. I have been getting a lot of help from Ryan, but I still feel like I'm not getting anywhere with this part of my project. Part of this is because I keep getting sick this semester. I don't know why, but I keep catching everything that comes around. This upcoming week I need to focus all of my free time on getting the previously mentioned part done and hopefully my body does not decide to get the flu next...

- 1 hour fixing previous Mud classes

- 2 hours working with MudController
- 3 hours looking up and not understanding network stuff

Total hours: 55

## Week 9:

Wolfen Blood



Ericka Zimmer

Report 8 of Week 9

November 10, 2009

I downloaded a mysql++ for making a database in C++. I was getting a lot of help from Ryan on how to use and do all of the work I needed for a simple login, but I keep having difficulty with the syntax. I think I understand it well enough now that I can make the Login work soon.

Also, I spent some time adding in the test section of the paper and editing other parts of the paper, such as the requirements section as to make them clearer to understand. The test section was confusing to me in some of the sections. I cannot figure out what exactly to put in those places. I understand the type of information to put in, but I don't know what to put in for my project. The logo started looking plain to me, so I decided to fix it up a bit as well. Along with improving the paper, I improved a lot of my code. Now there are no warnings when I compile due to converting string to char\* and when running the game it looks a lot nicer.

- 3 hours working on login system
- 2 hours adding section to paper and editing other sections
- 1 hour editing logo
- 1 hour fixing up code

Total hours: 62

Next week I intend to have the login completely working and then have some code done for the socket programming. Then I will have an actual Mud instead of a Sud (Single User Dungeon).



## Week 10:

Wolfen Blood



Ericka Zimmer

Report 9 of Week 10

November 17, 2009

When starting to work on the unit testing, my initial thought was to start creating test files in the code to test out the different classes. It wasn't until after creating the first one that I realized I could just use my main program to test everything by typing in different commands. Therefore, I began creating the unit test part of the paper and edited a bit of my original test cases.

Also, I spent some time commenting my code so that is someone was to have the desire to edit it, they can see in the header file what everything is supposed to do. Then I worked some more on the Login. I think I just about got the hang of it, but I cannot figure out the proper syntax for closing a connection in mySQL++

- 1 hour coding a test file that was not used
- 2 hours writing unit tests and editing test cases
- 2 hours commenting code and adding to login

Total hours: 67

To do by the end of thanksgiving: Everything... I plan to have the login and networking complete by the end of thanksgiving break. I will have plenty of time to work on it then.

## Week 11 & 12:

Wolfen Blood



Ericka Zimmer

Report 10 of Week 11 & 12

December 1, 2009

These past two weeks have been dreadful. Let me start by saying what I have accomplished and is working. After hours of working with the login system, I finally have it so that logging in to the game works and if the user enters in a name that is not stored, than the game asks for information and then inserts it into the database so that the next time one enters the game, the player is realized as there and just asks for a password. Yay! Now it is time for the more depressing part. I have read and tried out many sites, books, and source code to help me with network programming. I still have only a yoctogram of understanding for it as of now. I have spent many hours on this and still have nothing to show for it.

- 6 hours looking up and trying out network programming
- 5 hours completing MudLogin.cpp
- .5 hours organizing code in folder
- 1 hour trying to make demo before class starts

Total hours: 79.5

Next week, I hope to have some sort of networking done, and a much better demo than I'll have today.

## Week 13:

### Wolfen Blood



Ericka Zimmer

Final Report

December 8, 2009

After spending more time trying and failing at sockets, I figured it was time to give up on it for the sake of this project. There was no way I was going to get it done on time. If I had known how difficult this would be, I would have spent more time on making an RPG without the multiplayer. Also I may have decided to take someone else's socket code and wrote a clear and concise manual for using it, which nobody seems to have. They show you how to make it, but no one clearly states how to use it in different applications.

As far as code editing, it took me some time, but I can now save the location of where a character was so that when logging in, one starts out where they left off instead of the same place every time.

- 4 hours trying to make working socket programming and failing
- 3 hours revising and updating paper
- 3.5 hours editing code
- 2.5 hours making video
- 1.5 hours on poster board

Total hours spent: 94