USER'S MANUAL

Warboats

Created by The Hasbros: Keller Chambers, Stephen Haberle, Christian Ouellette, Peyton Rumachik

CSCI 205: Software Engineering & Design

May 1, 2017

Revision Sheet

Release No.	Date	Revision Description
Rev. 0	5/1/17	User's Manual and Checklist

USER'S MANUAL

TABLE OF CONTENTS

Page #

1.0 GENERAL INFORMATION 1-1

1.1 System Overview 1-1
1.2 Project References 1-2
1.3 Authorized Use Permission 1-2
1.4 Points of Contact 1-2
1.4.1 Project Manager 1-2
1.4.2 Scrum Master 1-2
1.4.3 Coding Staff 1-2
1.5 Acronyms and Abbreviations 1-2

2.0 SYSTEM SUMMARY 2-1

2.1 System Configuration 2-1
2.2 Data Flows 2-1
2.3 User Access Levels 2-1

3.0 GETTING STARTED & USING THE SYSTEM 3-1

3.1 Getting to the GUI 3-1
3.2 System Menu 3-1
3.3 Using the GUI 3-1
3.4 Exit System 3-2

4.0 SYSTEM OUTPUT 4-1

4.1 Console Output and Interpretation 4-1

1.0 GENERAL INFORMATION

1.0 GENERAL INFORMATION

1.1 System Overview

- Major functions performed by the system
    - LAN server connectivity
    - Challenge opponents to a game of Warboats and upon completion can re-challenge opponents
    - Drag and drop ship placement
    - Turn based game
- This application runs under an automatic client/server architecture
- Users interact with the software through a sleek graphical user interface (GUI) implemented through JavaFX
- Created by The Hasbros
- Operational status: Operational
- Warboats is a recreation of the Battleship board game. Players can place their fleet on the game board and battle their opponents for control of the sea through a turn based shooting style. Players can connect to one another through a LAN network simply by running the software.

1.2 Project References

https://github.com/EsotericSoftware/kryonet
https://docs.oracle.com/javase/8/docs/api/overview-summary.html

1.3 Authorized Use Permission

Permission to make copies of the code is not allowed. It is free to use but not to steal. Contact Stephen Haberle <sth007@bucknell.edu> with questions or concerns about legal use of materials.

1.4 Points of Contact

1.4.1 Product Owner

Chris Ouellete <clo006@bucknell.edu>

1.4.2 Scrum Master

Stephen Haberle <sth007@bucknell.edu>

1.4.3 Coding Staff

Keller Chambers <khc009@bucknell.edu>
Peyton Rumachik <pcr008@bucknell.edu>

1.5 Acronyms and Abbreviations

Graphical User Interface – GUI
Local Area Network - LAN
Unified Modeling Language - UML

2.0 SYSTEM SUMMARY

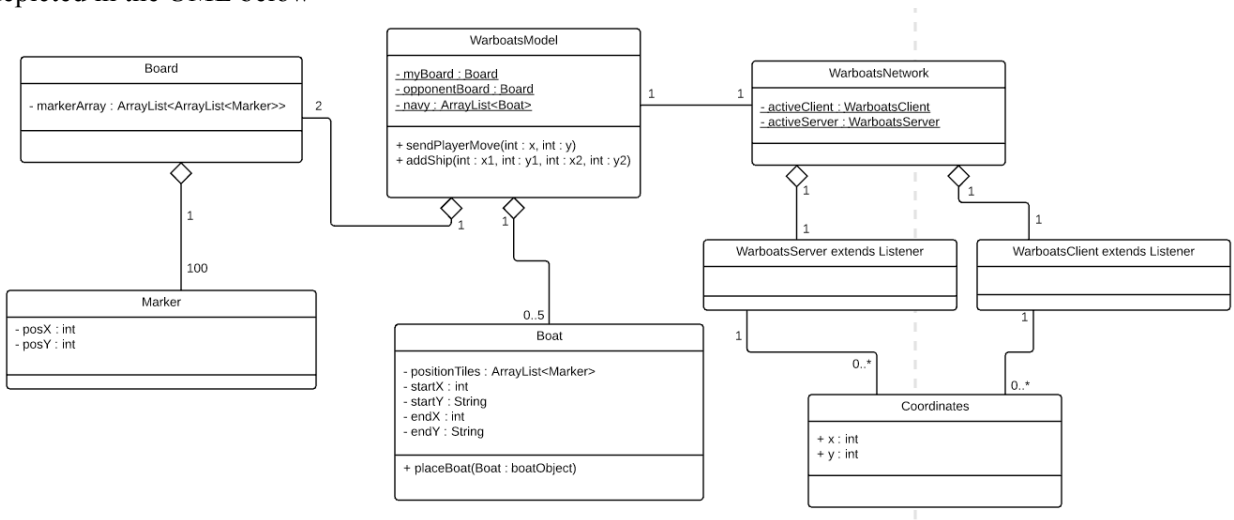
2.0 SYSTEM SUMMARY

2.1 System Configuration

The software currently works for LINUX computers that are connected to the same LAN. Users must have at the minimum a mouse to interact with the software; a keyboard is not needed.

2.2 Data Flows

Methods to maintain and store data are internally handled within the program. An overview of this is depicted in the UML below



2.3 User Access Levels

One user will act as the client and the other as a server. This is determined automatically. Both users should have the same experience within the game and should have a restricted view of their opponent's board. Only the shots they have fired will appear on the opponent's board and the type of shot (hit or miss) is indicated.

3.0 GETTING STARTED & USING THE SYSTEM

3.0 GETTING STARTED & USING THE SYSTEM

3.1 Getting to the GUI

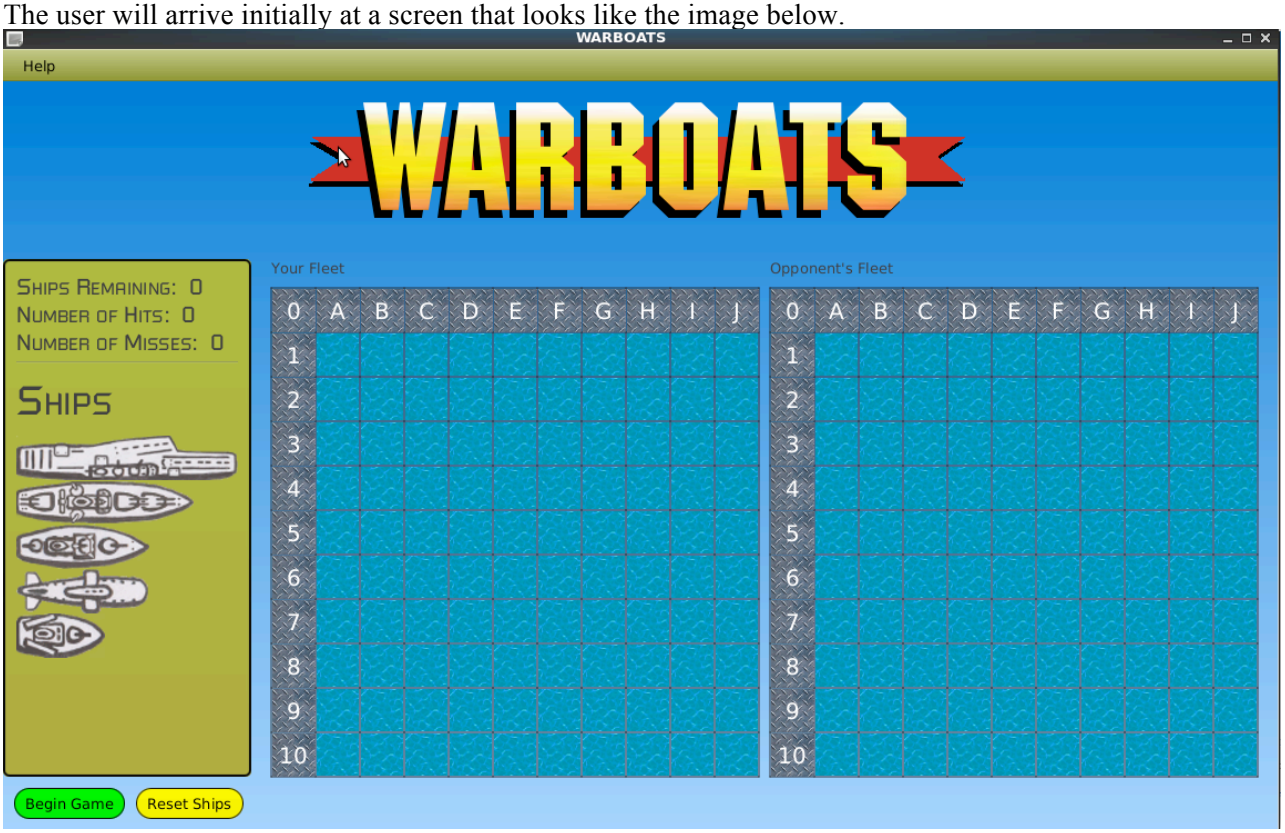
The software is downloaded as a .jar file. The files located in this compressed file must be extracted into a local folder on the user's system. To run the game, the user must select and run the executable

3.2 System Menu

The "Help" option at the top of the screen opens up a README file, which allows the user to learn how to use the software and complete the game.

3.3 Using the GUI

The user will arrive initially at a screen that looks like the image below.



From here, the user can place their boats in the "Your Fleet" section of the GUI. This is the user's board. By double clicking the ships once placed in the grid, the user can orient the ships along the opposite axis. If the user is not happy with the placement they have chosen, the option to reset ships resides in a button at the bottom of the screen.

Once both users have placed all their ships appropriately, selecting the "Begin Game" button will signal to the other user that you are ready to play. Appropriate alert messages will be shown if either user has failed to place their ships and hits the "Begin Game" button.

Once the game has begun, one user will be able to place a shot on their opponent's board by clicking on any tile in said board. This will be shown in the user's and opponent's GUI as either a hit or a miss. After shooting, the user loses control of the game until their opponent has placed their respective shot. The game continues until all of one player's ships have been destroyed. The number of ships remaining is indicated by the statistical panel on the left side of the GUI.

Once the game has been completed, the user will have the option to challenge their opponent to another game. If both users agree, a new game will be started. If not, the game will close.

3.4 Exit System

There are two options for exiting the system: 1) Upon completion of the game, the user's will be asked if they want to play their opponent again. If they choose the NO option, the game will automatically close. 2) The user can exit out of the window using the basic window control feature at the top right corner of the GUI.

4.0 SYSTEM OUTPUT

4.0 SYSTEM OUTPUT

4.1 Console Output and Interpretation

If the user runs the program from a terminal window, certain output will be generated. During the setup of the server/client relationship, the console will inform the user as to whether a server is already online (someone is ready to play) or whether a server needs to be built. If the server needs to be built, a success response is generated based on whether the server is created correctly. All other output should be disregarded.

```
Checking if server is online
Connecting to the server...
Connection failed. No existing server. Building server.
Creating the server...
Server is operational
```