Battleship  
Configuration Management Plan

Chris Hoorn  
Cody Plungis  
Tiffany Pohl

Table of Contents

[Software Configuration Management Plan 3](#_Toc322110632)

[Executive Summary 3](#_Toc322110633)

[Reference Documents 3](#_Toc322110634)

[Management 3](#_Toc322110635)

[Organization and Responsibilities 3](#_Toc322110636)

[Policies, Directives, and Procedures 3](#_Toc322110637)

[Activities 3](#_Toc322110638)

[Configuration Identification 3](#_Toc322110639)

[Configuration Control 4](#_Toc322110640)

[Resources 4](#_Toc322110641)

[Schedules 4](#_Toc322110642)

[Resources 4](#_Toc322110643)

[References 5](#_Toc322110644)

[Appendix 6](#_Toc322110645)

[Appendix A 6](#_Toc322110646)

[Appendix B 7](#_Toc322110647)

[Glossary 8](#_Toc322110648)

# Software Configuration Management Plan

## Executive Summary

This handy little document covers everything we need in milestone 2. Our tentative schedule is in Appendix A. Our class diagram is in Appendix B.

## Reference Documents

We are using the code conventions for Java as defined by Oracle. [1]

## Management

### Organization and Responsibilities

We are using a github repository to share and organize documents as well as source code. This account can be accessed at <https://github.com/pohltm/Battle>.

The team is responsible for monitoring and maintaining the above repository. Each individual will need to update his or her copy of the repository before any changes are made. Individuals will also need to commit and push any modifications or additions to the master branch of the repository.

### Policies, Directives, and Procedures

## Activities

### Configuration Identification

#### Identifying Configuration Items

All documentation for the project will be controlled by configuration management. This includes technical documents needed for milestones and progress reports. Alongside documentation, any source code for the project and test cases will be under configuration management control. Any requests for changes will also be organized by the configuration management.

#### Naming Conventions

All documentation in the repository will have a unique name pertaining to its contents. Title Case should be used for the document name and no revision states should be included in the name (e.g. Configuration Plan.docx).

All source code will be named following Oracle’s Java code conventions. [1]

#### Acquiring Configuration Items

All items should be placed in a specific folder in the repository depending on their type. Inside each folder, addition directories may exist that are not under the following guidelines:

* The “Code” folder will contain all source code for the project
* The “Documents” folder will contain all documents that are to be delivered
  + Within this folder, individual folders will exist for each milestone

### Configuration Control

#### Requesting Changes

Any changes will be talked about as a team. The team should create a ticket for each change that includes a description, type, and priority level.

#### Evaluating Requested Changes

Any bug that has been found should be tested and evaluated according to severity. Enhancements should be discussed as a group before the change is made.

#### Approving or Disapproving Changes

After evaluating a ticket, it can be approved by accepting it. Disapproving a ticket involves marking it as “unchanged.”

#### Implementing Changes

Once a change has been approved, it should be implemented. Following implementation, it should be submitted to the repository. The included commit message should be accurate and descriptive.

## Resources

### Schedules

A tentative schedule for this project is outlined in Appendix A. The given outline is relative to project milestones and other class activities.

### Resources

The tools used for the configuration management will be github.

# References

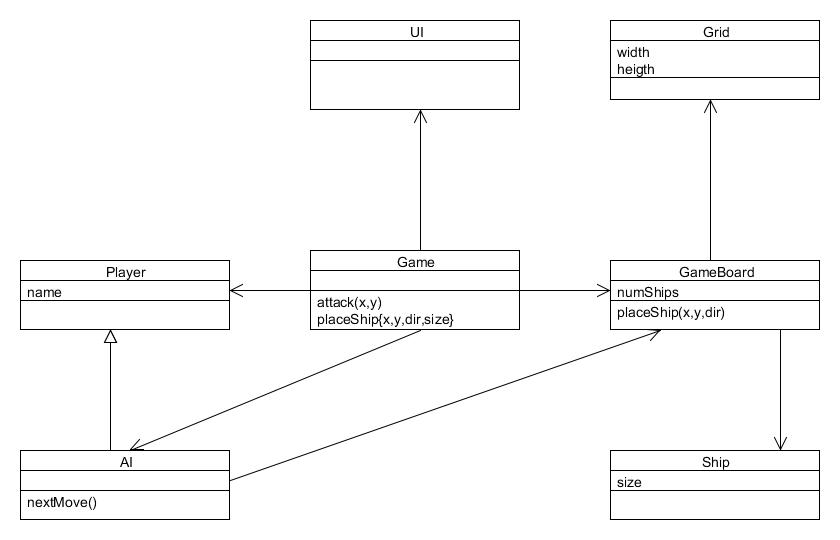
|  |  |
| --- | --- |
| [1] | Oracle, "Code Conventions for the Java Programming Language," Sun Microsystems, Inc., 20 April 1999. [Online]. Available: http://www.oracle.com/technetwork/java/javase/documentation/codeconvtoc-136057.html. [Accessed 13 April 2012]. |

# Appendix

## Appendix A

|  |  |
| --- | --- |
| **Week** | **Task(s)/Feature(s) – Implementer(s)** |
| 0 | Configuration Management Plan  Initial Class Diagram  Milestone 2  Week 1 Test Cases |
| 1 | Game Board and Ships  Research GUI  Week 2 Test Cases |
| 2 | Menu System  Basic GUI  Simple AI  Milestone 3  Week 3 Test Cases |
| 3 | Localization(German/English)  Connect Game Functions and GUI  Improved AI  Week 4 Test Cases |
| 4 | Full Game Functionality  Week 5 Test Cases  Milestone 4 |
| 5 | Finalize Project  Add Extra Features  Milestone 5 |

## Appendix B



# Glossary