*Prog3 AT3 Project*

Product Design Specification

Version *1.0*

*6/11/2021*

Name: Bradley Willcott

ID: M198449

Date: 6 November 2021

VERSION HISTORY

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version #** | **Implemented**  **By** | **Revision**  **Date** | **Approved**  **By** | **Approval**  **Date** | **Reason** |
| 1.0 | *Bradley Willcott* | *6/11/2021* | *<name>* | *<mm/dd/yy>* | Initial Design Definition draft |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**UP Template Version:** 12/31/07

TABLE OF CONTENTS

[1 Introduction 4](#_Toc87111262)

[1.1 Purpose of The Product Design Specification Document 4](#_Toc87111263)

[2 General Overview and Design Guidelines/Approach 4](#_Toc87111264)

[2.1 Assumptions / Constraints / Standards 4](#_Toc87111265)

[3 Architecture Design 5](#_Toc87111266)

[3.1 Logical View 5](#_Toc87111267)

[3.2 Hardware Architecture 9](#_Toc87111268)

[3.3 Software Architecture 9](#_Toc87111269)

[4 System Design 10](#_Toc87111270)

[4.1 Use-Cases 10](#_Toc87111271)

[4.2 User Interface Design 10](#_Toc87111272)

[4.3 Coding Compliance 10](#_Toc87111273)

Figures

[Figure 1 - Overall Specification UML 5](#_Toc87111257)

[Figure 2 - Common Library Specification UML 6](#_Toc87111258)

[Figure 3 - GUI Client Specification UML 7](#_Toc87111259)

[Figure 4 - RMI Server Specification UML 8](#_Toc87111260)

[Figure 5 - Socket Server Specification UML 8](#_Toc87111261)

# Introduction

## Purpose of The Product Design Specification Document

The Product Design Specification document documents and tracks the necessary information required to effectively define architecture and system design in order to give the development team guidance on architecture of the system to be developed. The Product Design Specification document is created during the Planning Phase of the project. Its intended audience is the project manager, project team, and development team. Some portions of this document such as the user interface (UI) may on occasion be shared with the client/user, and other stakeholder whose input/approval into the UI is needed.

# General Overview and Design Guidelines/Approach

This section describes the principles and strategies to be used as guidelines when designing and implementing the system.

## Assumptions / Constraints / Standards

The program will allow the user to create new account, login to an existing account, and run a chat session.

# Architecture Design

This section outlines the system and hardware architecture design of the system that is being built.

## Logical View

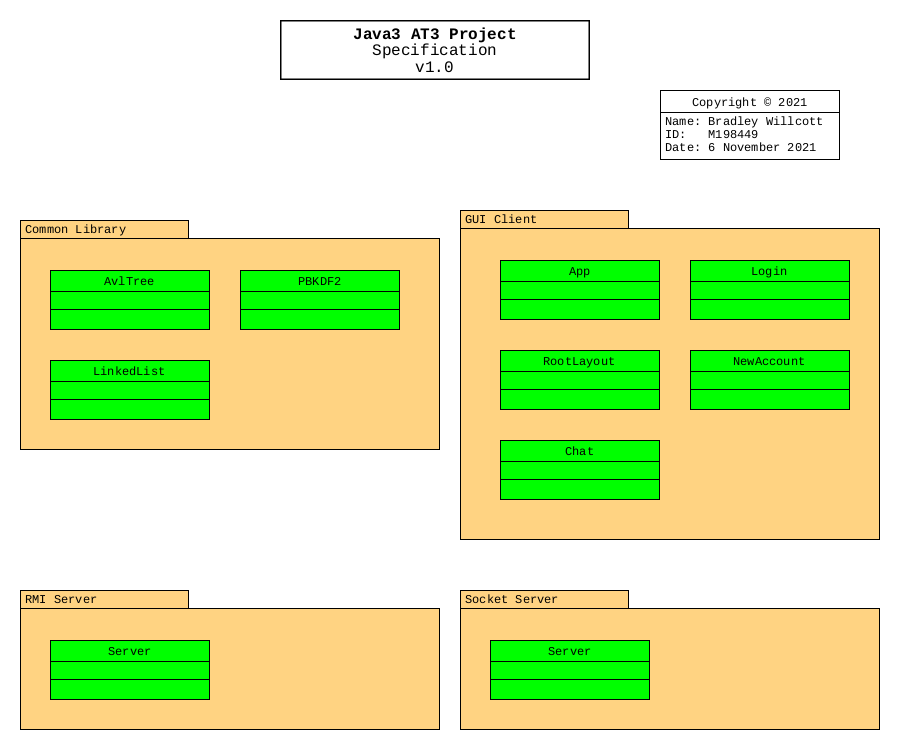


Figure - Overall Specification UML

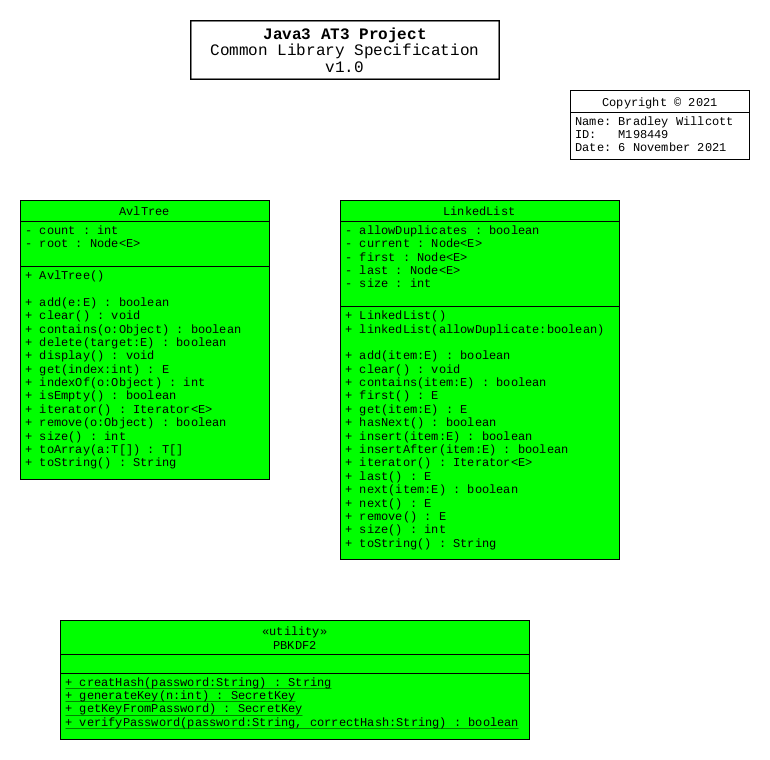


Figure - Common Library Specification UML

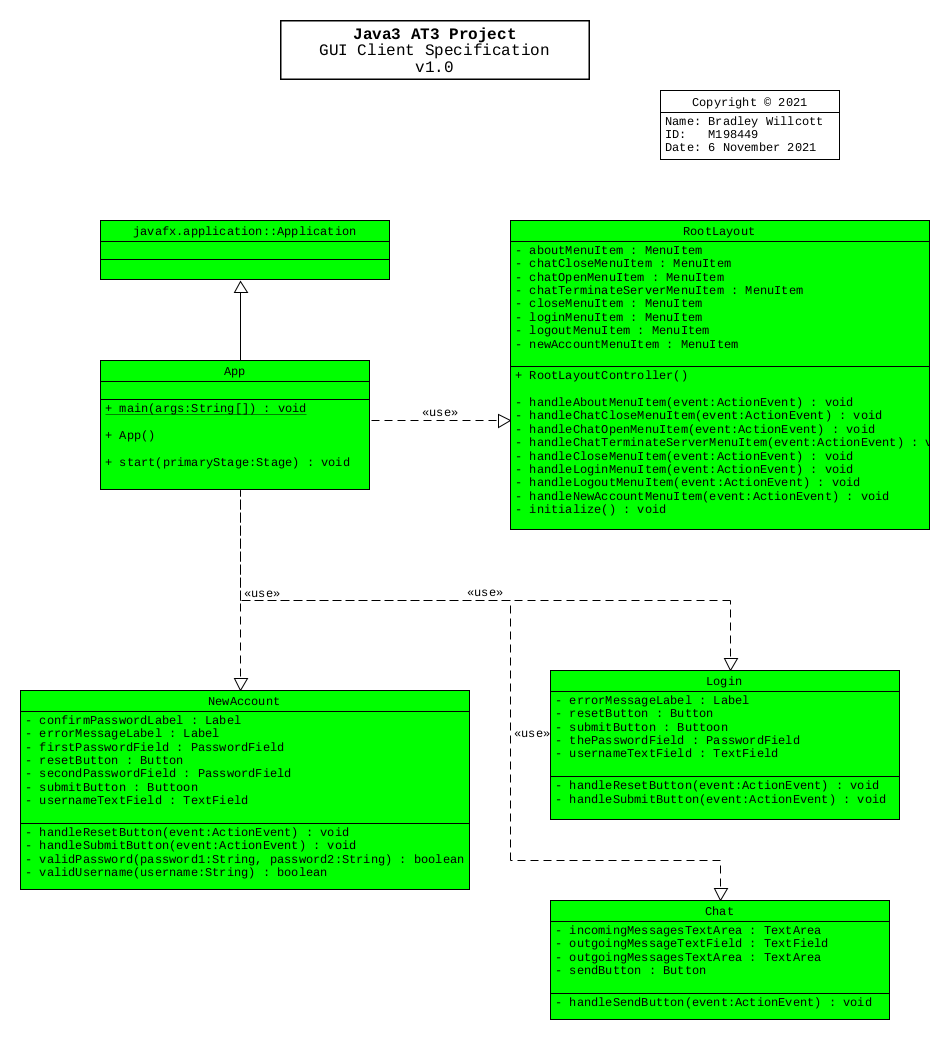


Figure - GUI Client Specification UML

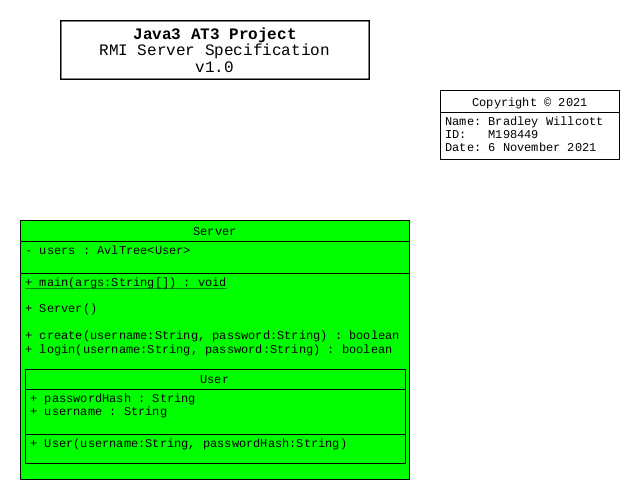


Figure - RMI Server Specification UML

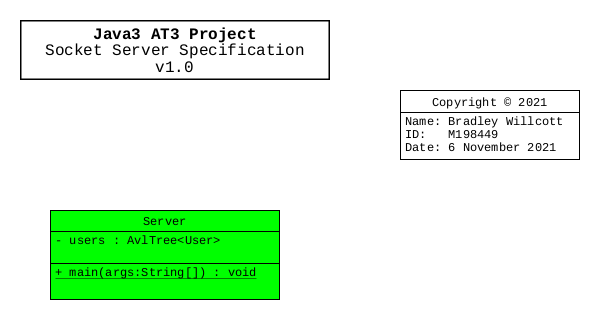


Figure - Socket Server Specification UML

## Hardware Architecture

The design only needs to allow for the program to be run on any Microsoft Windows 10 compatible desktop or portable hardware, be it Intel or AMD processor based.

## Software Architecture

The language to be used, need only be compatible with the Microsoft Windows 10 operating system.

### Source Control System

A source control system will be implemented to allow for version control of the project files. It is recommended that the GitHub site, which uses the git version control system and software, be used to provide this facility. It will be necessary that each team member obtain his/her own GitHub membership.

The project will be located at: <https://github.com/bewillcott/Java3AT3-Project>

# System Design

## Use-Cases

The project scenario is to develop a program that provides a chat facility, protected by username and password. This requires the user to login to the server prior to gaining this access.

There are to be two separate servers:

* An RMI server – User authentication functions
* A Socket serve – Echo Chat functions

The RMI server will use cryptographic level hashing of the passwords which are to be stored in a CSV file, local to that server.

## User Interface Design

A Graphical User Interface is to be provided to allow the user to create new accounts, login under an existing account, and to participate in Chat sessions.

## Coding Compliance

The following standards must be adhered to for software coding:

<https://www.oracle.com/technetwork/java/codeconventions-150003.pdf>

# Test plan

There will be two levels of testing implemented.

* Unit tests (Class)
* Manual tests (GUI)

## Unit testing

Where appropriate, unit testing will be used to show that required functionality has been achieved, and to provide for regression testing.

## Manual testing

When unit testing is not practical, such as test the Graphical User Interface, manual testing will be used, following a set of testing scenarios as set out below:

1. Start-up RMI and Socket Servers
2. Start-up GUI Client
3. Create a New Account
4. Show Logout menu item now enabled
5. Open Chat session
6. Have a Chat
7. Close Chat session
8. Logout
9. Display About Dialog
10. Display the Help Window
11. Test Login with wrong password
12. Show validation of new Username & Password
13. Test GUI Application with both servers down
14. Restart RMI Server and retry Login
15. Try to open Chat session
16. Restart Socket Server and reopen Chat session
17. Logout with Chat session open
18. Final look at server consoles with log entries
19. Close the application