**Software Design Document**

**For Target Finder**

By: Devin Taniguchi

Rudy Diaz

Zijun Zhao

Contents

[**1.** **Introduction** 3](#_Toc451604165)

[**2.** **System Overview** 3](#_Toc451604166)

[**3.** **System Architecture** 3](#_Toc451604168)

[**3.1** **Assumptions** 3](#_Toc451604169)

[**3.2** **Constrains** 3](#_Toc451604170)

[**3.3** **System environment** 3](#_Toc451604171)

[**3.4** **Architecture** 3](#_Toc451604172)

[**4.** **Data Design** 3](#_Toc451604173)

[**5.** **Component Design** 3](#_Toc451604174)

[**6.** **Human Interface Design** 3](#_Toc451604175)

[**7.** **Requirements** 3](#_Toc451604176)

1. **Introduction**
2. **System Overview**

The Target Finder system provides a simulation for the path-finding of a mouse reaching the destination in the shortest steps. The system has four layers, including user interface, user interface management (Authentication and Authorization), application functionality, system support (database). The user interface is used by clients to interact with the data system. User authentication is used to verify the users. The application functionality provides the functionality for the system. The database system manages a lot of different maze information.

1. **System Architecture**
   1. **Assumptions**

The user of the Target Finder system should have a knowledge of the basic operations of a computer and the system.

* 1. **Constrains**

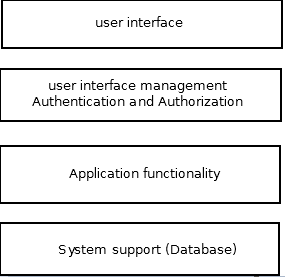
The Target Finder system is designed to create only for the robot company. The system is designed to implement by Java.

* 1. **System environment**

The Target finder system is designed to work perfectly in any operating system such as Windows, Linux, and OSX. It is also designed to work through any computer and laptop.

* 1. **Architecture**

The layered architecture demonstrates the architecture of Target Finder system.



The Target Finder system has four systems, including user interface, user interface management (Authentication and Authorization), application functionality, system support (database).

1. The top layer is responsible for the user interface. User could interact with the interface. The interface will give the user a guideline to guide user how to edit the inputs and get expected output. The user interface will implement using Java Gui.
2. The second layer is responsible for user to log in to the system. When customers want to access the system, authentication is utilized to identify exactly who is accessing the program, and authorization is utilized to identify if the user has permission to use this resource.
3. The third layer provides the application functionality of the system. It interact with user’s input and generate the expected output.
4. The last layer is responsible for database system Support. The data system is designed to deliver maze information to the user interface when user select a maze. User interface will generate the maze according to the maze information. Also it provides database management system.
5. **Data Design**
6. **Component Design**
7. **Human Interface Design**
8. **Requirements**