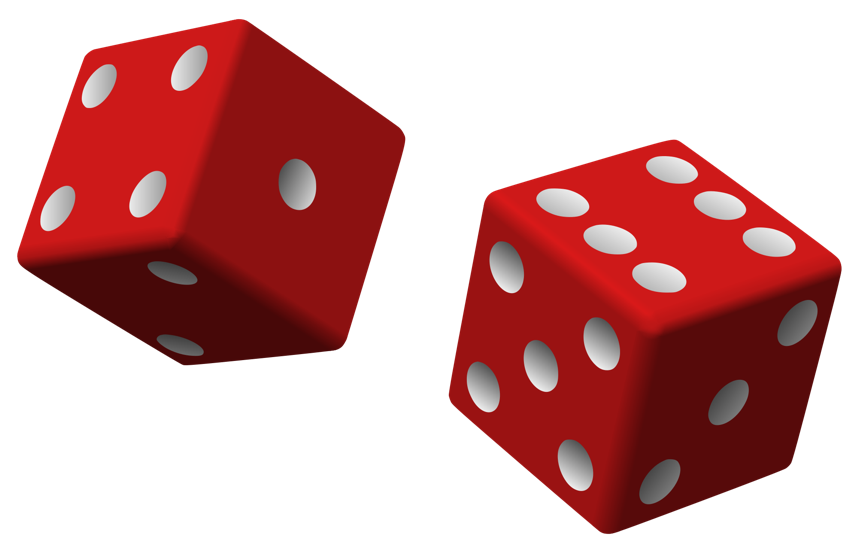
Chance-It

Module Guide

Version 1.0

**CMMS Systems**

Chris Wong

Matthew Casiro

Melissa Page

Sheryll Tabamo

Table of Contents

[Module Breakdown 1](#_Toc451203441)

[Module Interfaces 1](#_Toc451203442)

[Game 1](#_Toc451203443)

[Input/Output 1](#_Toc451203444)

[Local Turn 1](#_Toc451203445)

[Network Turn 1](#_Toc451203446)

[Dice 1](#_Toc451203447)

[Random 1](#_Toc451203448)

[Probability 1](#_Toc451203449)

[Network Protocol 1](#_Toc451203450)

[Computer Player 1](#_Toc451203451)

[High Score 1](#_Toc451203452)

[Uses Hierarchy 1](#_Toc451203453)

[Appendix A 2](#_Toc451203454)

# Module Breakdown

|  |  |  |  |
| --- | --- | --- | --- |
| Module Name | Owner | Tester | Intended Abstraction |
| App Driver | Chris W | Sheryll T | Hides how the program is initialized |
| Game | Matthew C | Melissa P | Hides how game moves between states based on user selections and game progress |
| Input/Output | Sheryll T | Melissa P | Hides all screen output logic as well as user input collection and validation |
| Local Turn | Melissa P | Chris W | Hides logic for walking through a turn and determining the turn score |
| Network Turn | Melissa P | Sheryll T | Hides network communications for interacting with a server for online play |
| Dice | Chris W | Matthew C | Hides dice attributes and dice roll functionality |
| Random | Matthew C | Sheryll T | Hides how random numbers are generated |
| Probability | Sheryll T | Matthew C | Hides probability calculations for chance to re-roll the first turn sum |
| Network Protocol | Chris W | Matthew C | Hides connection protocols to create connection with the server |
| Computer Player | Matthew C  Melissa P | Chris W  Sheryll T | Hides decision making algorithms for the computer player |
| High Score | Sheryll T | Chris W | Hides storage and retrieval method for high score data as well as score comparisons |

# Module Interfaces

Game:

Pre: randomInit() has been called once

Post: N/A

Clean-Up: N/A

Param: player is a pointer to an unsigned variable

Return: the winning score of the game, or zero if a computer or network player won

**unsigned gameInit();**

Input/Output:

// Pre: N/A

// Post: The screen was-Updated to display new information

// Clean-Up: N/A

// Param name is the player's name

// Param userSelection is the user's choice

// Returns N/A

**void displayGamePlay(char\* name, int userSelection);**

// Pre: A highscore file exists

// Post: The highscore was displayed on the screen

// Clean-Up: N/A

// Param N/A

// Returns N/A

**void displayHighScore();**

// Pre: N/A

// Post: The main menu was displayed on the screen

// Clean-Up: N/A

// Param N/A

// Returns N/A

**void displayMainMenu();**

// Pre: N/A

// Post: The help menu was displayed on the screen

// Clean-Up: N/A

// Param N/A

// Returns N/A

**void displayHelpMenu();**

// Pre: N/A

// Post: The Network Play Mode was displayed on screen

// Clean-Up: N/A

// Param N/A

// Returns N/A

**void displayNetworkPlayMode();**

// Pre: N/A

// Post: The prompt for network information was displayed on screen

// Clean-Up: N/A

// Param N/A

// Returns N/A

**void displayNetWorkPlayInput();**

// Pre: N/A

// Post: The Local Play mode was displayed on screen

// Clean-Up: N/A

// Param N/A

// Returns N/A

**void displayLocalPlayMode();**

// Pre: N/A

// Post: The menu for in-game help was displayed on screen

// Clean-Up: N/A

// Param N/A

// Returns N/A

**void displayInGameHelpMenu();**

Local Turn:

// Pre: N/A

// Post: N/A

// Clean-Up: N/A

// Returns: the final turn score

**unsigned localTurn();**

Network Turn:

// Recieves data from the network player

// Pre: N/A

// Post: N/A

// Cleanup N/A

// Return an int of the turn score

**int networkTurn();**

Dice:

Random:

Probability:

Network Protocol:

Computer Player:

High Score:

# Uses Hierarchy

# Appendix A