Yahtzee game

made in C#

By Mehlian

12/28/2017

Table of contents

[1 Introduction 2](#_Toc508199929)

[1.1 Overview 2](#_Toc508199930)

[1.2 Goals and Objectives 2](#_Toc508199931)

[1.3 Scope 2](#_Toc508199932)

[1.4 Definitions 2](#_Toc508199933)

[2 General Design Constraints 2](#_Toc508199934)

[2.1 Yahtzee Application Environment 2](#_Toc508199935)

[2.2 User Characteristics 2](#_Toc508199936)

[3 Nonfunctional Requirements 2](#_Toc508199937)

[3.1 User Interface 2](#_Toc508199938)

[3.1.1 General information 2](#_Toc508199939)

[3.1.2 UI sketches 3](#_Toc508199940)

[4 Functional Requirements 5](#_Toc508199941)

[4.1 Game rules 5](#_Toc508199942)

[4.2 Required Features 7](#_Toc508199943)

[4.3 Use case 1 7](#_Toc508199944)

[4.4 Use case 2 8](#_Toc508199945)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Name** | **Description** |
| 1 | 12/28/2017 | Mehlian | Initial document |
| 2 | 03/07/2018 | Mehlian | Document polish |
|  |  |  |  |

# Introduction

## Overview

Yahtzee is a game for 2-4 players. Each player rolls set of 5 dice each turn and based on roll results receives points. Yahtzee application will be a desktop app for Windows OS. The app will allow players to play Yahtzee game and save their score.

This document provides information on requirements for the Yahtzee software application.

## Goals and Objectives

Project’s goal is to provide users ability to play Yahtzee game in virtual environment with ability to store and retrieve game scores.

## Scope

Application is designed to run on Windows 10 with .NET Framework.

## Definitions

**Yahtzee Application/ App/ Software** – the product that is being described in this document.

**User** – the person or persons who will interact with the Yahtzee application.

**Use Case** – description of goal – oriented interaction between the system and an actor. A use case may define several variants called scenarios that result in different paths through the use case and usually different outcomes.

**Scenario** – one path through a user case.

**Actor** – user or other software system that receives value from a user case.

**Developer** – the person or organization developing the system, also sometimes called the supplier.

# General Design Constraints

## Yahtzee Application Environment

Application is designed to work on a Windows 10 and .Net Framework 4.6.1.

## User Characteristics

User in anyone that gain access to the application.

# Nonfunctional Requirements

## User Interface

### General information

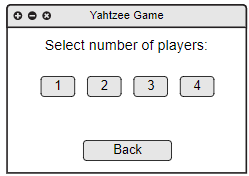
Application is designed to work as WPF app. UI should inform users with game’s status.

### UI sketches

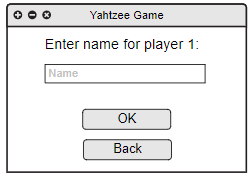
**Main window** – welcome screen:



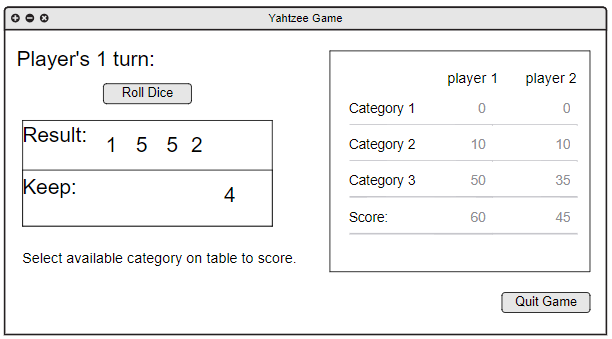
**Main Window** – number of players selection:



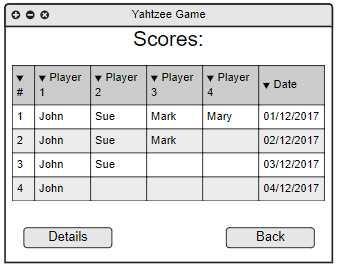
**Main Window** – player’s name input:



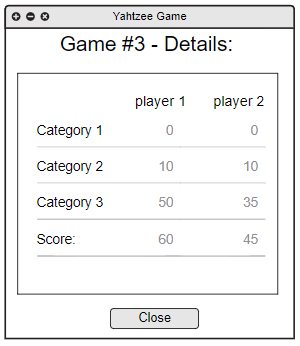
**Game Window:**



**Scores Window:**



**Scores Window** – details panel:



# Functional Requirements

## Game rules

**General rules:**

Game consists of thirteen turns, where every player can roll set of dice three times. Every turn starts with roll of 5 dice, then Player can choose if he wants to roll again with selected dice or assign points for given category. Every player on his turn can roll dice up to three times. By choosing category, player ends his turn and makes chosen category unavailable for him in his next turn.

All thirteen categories are shown in table below:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **Player 1** | **Player 2** | **Player 3** | **Player 4** |
| **First** **Half** | Aces |  |  |  |  |
| Twos |  |  |  |  |
| Threes |  |  |  |  |
| Fours |  |  |  |  |
| Fives |  |  |  |  |
| Sixes |  |  |  |  |
|  | **SUM (Bonus)** |  |  |  |  |
| **Second Half** | 3 of a kind |  |  |  |  |
| 4 of a kind |  |  |  |  |
| Full house |  |  |  |  |
| Small straight |  |  |  |  |
| Large straight |  |  |  |  |
| Yahtzee |  |  |  |  |
| Chance |  |  |  |  |
|  | **SUM (Bonus)** |  |  |  |  |
| **RESULTS** |  |  |  |  |

**Score rules:**

The table consists of two parts – simple categories (First Half) and complex categories (Second Half). In first half of the table (available categories: Aces, Twos, Threes, Fours, Fives, Sixes) score is determined by summing points of the similar dices.

If sum of points for first half of the table gives 63 points or more, given player gets bonus 35 points.

In second half of the table, score is determined based on requirements for given category:

* 3 of a kind – there are at least 3 dice of same kind; score is based on sum of all dice. Example: 2,3,4,4,4 gives 17 points.
* 4 of a kind – there are at least 4 dice of same kind; score is based on sum of all dice. Example: 4,5,5,5,5 gives 24 points.
* Full house – combination of three of one number and two of another; gives 25 points. Example: 2,2,5,5,5 gives 25 points.
* Small straight – four sequential dice; gives 30 points. Examples: 1,3,4,5,6 or 1,2,3,4,6 gives 30 points.
* Large straight – five sequential dice; gives 40 points. Examples: 1,2,3,4,5 or 2,3,4,5,6 gives 40 points.
* Yahtzee – all five dices are the same; gives 40 points. Example: 1,1,1,1,1 or 5,5,5,5,5 gives 50 points.
* Chance – score is based on sum of all dice. Example: 1,1,3,3,5 gives 13 points.

Every player can choose categories freely after every dice roll, but if conditions are not met he scores 0 points for given category.

Example: Player 1 rolls 5 dice and gets this result: 2, 2, 3, 6, 6.

Here are his options to score:

|  |  |  |
| --- | --- | --- |
| **Category** | **Possible Score for given category** | **Description** |
| Twos | 4 | two dices with 2 points (2x2) |
| Threes | 3 | one dice with 3 points (1x3) |
| Sixes | 12 | two dices with 6 points (2x6) |
| Aces, Fours, Fives | 0 | zero dices for given category |
| 3 of a kind, 4 of a kind | 0 | no requirements are meet |
| Full house | 0 | no requirements are meet |
| Small, Large Straight | 0 | no requirements are meet |
| Yahtzee | 0 | no requirements are meet |
| Chance | 19 | sum of all sides |

Player 1 can choose any category from first or second half of the table. If he can’t meet with requirements for given category or no more categories left that meet requirements, then he must choose category to assign 0 points.

**Extra rules:**

If player rolls “Yahtzee” more than one time, and in first half of the table suitable category is already taken, player can choose freely other categories from second half of the table and on top of that scores 100 bonus points.

Example: Player 1 has already taken “Threes” category and “Yahtzee” category, then rolls 3,3,3,3,3. Now he gets 100 bonus points, can select freely category from second half of the table and score points even if requirements are not meet. But if there is no more available categories in second half, he must to choose category in first half of the table that will be scored as 0 points.

**Ending:**

End of the game occurs when all categories are taken for every player. Player with highest score wins.

## Required Features

**General Features:**

1. Ability to play Yahtzee game on PC with Windows 10 OS.
2. Support for 1 to 4 players.
3. Application provides table with game categories.
4. Application displays players names .
5. Application displays scores for each category.
6. Application allows to roll set of 5 dice.
7. Application allows to keep selected dice.
8. Application displays roll results.
9. Application informs about game’s state.
10. Application saves scores when game is over.

**Additional Features:**

1. Player order is determined by order of name input.
2. Game scores are saved as xml files.

## Use case 1

**Description**

1. User starts application to play the game.

**Actors**

1. One or more users.

Steps of Execution (basic path)

1. User starts Yahtzee Application.
2. System displays welcome screen.
3. User clicks “New Game” button.
4. System asks for number of players.
5. User selects number of players by clicking on proper button.
6. System asks for players names.
7. User provides players names.
8. System provides clear table with categories to fill and asks player one to roll dice.
9. Active player rolls the dice.
10. System provides result of roll and displays available options for given player.
    1. Player selects dice to keep and then roll remaining dice again (return to step 10; max 2 times).
    2. Player selects category to assign points based on roll results.
11. System assigns points and asks next player to roll the dices (return to step 9).
12. If all categories are taken for all players, System compares points and announce the winner.
13. System saves Game results.
14. System asks if players wants to play again.
    1. User selects Yes button (return to step 8).
    2. User selects No button (return to step 2).

Steps of Execution (alternate path)

1. User starts Yahtzee Application.
2. System displays welcome screen.
3. User clicks “Exit Game” button.
4. System exits.

## Use case 2

**Description**

1. User starts application to view score of past games.

**Actors**

1. One or more users.

Steps of Execution (basic path)

1. User starts Yahtzee Application.
2. System displays welcome screen.
3. User clicks “Scores” button.
4. System displays table with scores an user names.
5. User selects record from score table and clicks “Details” button.
   1. System displays selected game details providing game’s table and scores for each category.
   2. User clicks “Close” button (return to step 4).

Steps of Execution (alternate path)

1. User starts application.
2. System displays welcome screen.
3. User clicks “Scores” button.
4. System displays table with scores an user names.
5. User clicks “Back” button (return to step 2).