
Software Requirements Specification

for

Roulette Game

Version 1.0 approved

Prepared by Mohamed Sultan

9-November-2015

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction.....	1
1.1 Purpose	1
1.2 Product Scope	1
1.3 References	1
2. Overall Description.....	1
2.1 Product Functions	1
2.2 User Classes and Characteristics	1
2.3 Design and Implementation Constraints.....	2
3. External Interface Requirements	2
3.1 User Interfaces	2
3.2 Software Interfaces	2
3.3 Communications Interfaces	2
4. Functional Requirements	2
4.1 System Feature 1	2
4.2 System Feature 2	3
4.3 System Feature 3	3
4.4 System Feature 4	3
4.5 System Feature 5	4
4.6 System Feature 6	4
4.7 System Feature 7	4
4.8 System Feature 8	5
4.9 System Feature 9	5
4.10 System Feature 10	5
5. Nonfunctional Requirements	6
5.1 Environmental Requirements	6
5.2 Social Requirements	6

Revision History

Name	Date	Reason For Changes	Version
Mohamed Sultan	9-Nov-15		1.0

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Web Roulette game. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate.

The intended audience of this document is system designers, software developers, and other possible concerned parties within implementation and deployment phases.

1.2 Product Scope

Roulette game is developed to produce results based on user bets and their bet type selection. It uses random generator to simulate spinning the Roulette and hence compares the user bet versus the generated one.

1.3 References

1. Wikipedia Roulette web page, <https://en.wikipedia.org/wiki/Roulette>. Last accessed 9-Nov-2015.
2. Eclipse IDE for Java EE, <https://eclipse.org/downloads> last accessed 9-Nov-2015.

2. Overall Description

2.1 Product Functions

The software application has the following main functions

1. Simulate Roulette game and winning bets.
2. Analyze user bets
3. Accumulates user winnings along every bet
4. Notifies the user of latest total winning amounts

2.2 User Classes and Characteristics

There is one main class of user who is anticipated to use this application. This main user class is referred to in this document as the player (gambler). The player user should have technical expertise in using computer games to play/run this application.

2.3 Design and Implementation Constraints

In the current phase (phase1) no specific constraints. However, some constraints might be needed.

3. External Interface Requirements

3.1 User Interfaces

Currently there are no restrictions on screen layout as it is working as backend only. Only the user has to select bet amount more than zero and select a pocket within pocket ranges.

3.2 Software Interfaces

No external interfaces for the system. However in future database will be needed to host the user logins, transactions and current amount he has.

3.3 Communications Interfaces

The system is using HTTP as protocol between HTML page and the backend system. In next phases future, will use database interface from the user servlet(s) and the hosting database, in phase 2.

4. Functional Requirements

4.1 System Feature 1

4.1.1 Description and Priority

In case of straight bets, the user places a bet on a single pocket. If the user has a successful bet, he wins 36 times his bet, else he get nothing.

4.1.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects the expected pocket number (which is **not** equal to the generated randomly by the system).
3. The user submit his request
4. The user gets £0 winnings.

4.2 System Feature 2

4.2.1 Description and Priority

In case of straight bets, the user places a bet on a single pocket. If the user has a successful bet, he wins 36 times his bet, else he get nothing.

4.2.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects the expected pocket number (which is equal the generated randomly by the system).
3. The user submit his request
4. The user gets £360 winnings.

4.3 System Feature 3

4.3.1 Description and Priority

Without any condition, the user bets with an amount less than or equal to zero

4.3.2 Stimulus/Response Sequences

1. The user places a bet of invalid amount.
2. The user submit his request
3. The user gets a user-friendly error message informing him that he has to enter a valid amount more than zero

4.4 System Feature 4

4.4.1 Description and Priority

Without any condition, the user bets with a pocket number not in range. The pockets of the roulette wheel are numbered from 1 to 36.in addition to the 0/00 pockets.

4.4.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user submit his request
3. The user gets a user-friendly error message informing him that he has to enter a valid pocket number.

4.5 System Feature 5

4.5.1 Description and Priority

In case of even/odd bets, the user places a bet on pocket 0, he gets nothing.

4.5.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects pocket number 0
3. The user submit his request
4. The user gets £0 winnings.

4.6 System Feature 6

4.6.1 Description and Priority

In case of even/odd bets, the user places a bet on an even pocket. If the user has a successful bet, he wins 2 times his bet, else he get nothing.

4.6.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects the expected even pocket number (which is equal to the generated randomly by the system).
3. The user submit his request
4. The user gets £20 winnings.

4.7 System Feature 7

4.7.1 Description and Priority

In case of even/odd bets, the user places a bet on an even pocket. If the user has a successful bet, he wins 2 times his bet, else he get nothing.

4.7.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects the expected even pocket number (which is **not** equal the generated randomly by the system).
3. The user submit his request
4. The user gets £0 winnings.

4.8 System Feature 8

4.8.1 Description and Priority

In case of even/odd bets, the user places a bet on an even pocket. If the user has a successful bet, he wins 2 times his bet, else he get nothing.

4.8.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects the expected odd pocket number (which is **not** equal the generated randomly by the system).
3. The user submit his request
4. The user gets £0 winnings.

4.9 System Feature 9

4.9.1 Description and Priority

In case of even/odd bets, the user places a bet on an even pocket. If the user has a successful bet, he wins 2 times his bet, else he get nothing.

4.9.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects the expected pocket odd number (which is **not** the generated randomly by the system).
3. The user submit his request
4. The user gets £0 winnings.

4.10 System Feature 10

4.10.1 Description and Priority

In case of straight bets, the user places a bet on a single pocket. If the user has a successful bet, he wins 36 times his bet, else he gets nothing.

4.10.2 Stimulus/Response Sequences

1. The user places a bet of £10.
2. The user selects the 00 pocket number (which is **not** equal the generated randomly by the system).
3. The user submit his request
4. The user gets £360 winnings.

5. Nonfunctional Requirements

5.1 Environmental Requirements

This software is a desktop application that does not require internet connection and thus does not need any specific environmental requirements in this phase. Phase 2 should require internet connection if an e-payment is integrated to allow actual bets.

5.2 Social Requirements

In the current phase (phase1) no specific social requirements are needed other than that the user has to have experience with computer games. However, in later phases the system has to check the user age and ensure that he is an adult +18.