<Monopoly Game>

Supplementary Specification

Version <2.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 2020/04/21 | 1.0 | First draft. | LianJiawei Group |
| 2020/06/20 | 2.0 | Update after Elaboration iteration. | LianJiawei Group |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Definitions, Acronyms, and Abbreviations 4

1.4 References 4

1.5 Overview 4

2. Functionality 4

3. Usability 5

4. Reliability 5

5. Performance 5

6. Supportability 6

7. Design Constraints 6

8. Online User Documentation and Help System Requirements 6

9. Purchased Components 6

10. Interfaces 6

10.1 User Interfaces 6

10.2 Hardware Interfaces 6

10.3 Software Interfaces 6

10.4 Communications Interfaces 6

11. Licensing Requirements 7

12. Legal, Copyright, and Other Notices 7

13. Applicable Standards 7

Supplementary Specification

# Introduction

## Purpose

This document describes all the requirements of Monopoly game that are not described in the use cases.

## Scope

This Supplementary Specification applies to the Monopoly Game which will be developed.

## Definitions, Acronyms, and Abbreviations

See Glossary

## References

Glossary.

## Overview

Supplementary Specification is organized by introduction, functionality, usability, reliability, performance, supportability, design constraints and so on.

# Functionality

## <Functional Requirement One>

Functional requirements are captured via the defined use cases

# Usability

## <Ease of use>

The game system will not require user training beyond that of using an android mobile phone or any other android device. This will be verified by usability tests during the beta period.

# Reliability

## <Save Load>

When the player exits the game abnormally, the system automatically saves it to provide the recovery of the game situation for the next time the player enters the game。

# Performance

## <Performance Time Requirement>

The player wants to get a quick response from game system when the player is moving, using a prop card, etc. This delay should not exceed 0.1s.

# Supportability

## <Supportability Requirement One>

No custom software will be required to reside on the user’s android device.

# Design Constraints

## <Develop Standards>

This game system shall conform with existing android app design standards. This game is going to be developed with Java and android API. Game system must follow the game rules.

## <Game Rules>

This Monopoly game is a 2-4 player simulation game. The Monopoly board has 28 squares, 20 of which represent property that players can buy. Players occupy property and build houses to let opponents pay tolls and increase their own money. The more property they occupy, the higher the profit. The last player who is not bankrupt became the winner of the game.

1. Set the parameters such as the number of players when starting the game, and different colors are used to mark different player.

2. Turn-based game. Each player rolls the dice and the pips of dice determine the number of squares that the player can move.

3. Players can use prop cards before rolling the dice to benefit themselves or frame other players.

4. After the player moves to a square represent property, if it is not occupied, the player can choose to buy the property; if the square is already occupied by the player, the player can choose to build a house or upgrade the house level; if the square is occupied by others, the player pay tolls.

5. The price of each property has its own difference. The house has level up to 4. The higher the house level is, the more tolls need to be paid when passing it.

6. The player moves to the prison and enters a state of imprisonment. Within 2 rounds, he cannot move and he cannot charge a toll.

7. The player moves to the designated square to trigger a random event, and the random event occurs during the game randomly.

8. When the player's money is reduced to 0, he enters a bankruptcy state, and all occupied property is forcibly recovered.

9. The last player who did not go bankrupt becomes the winner of the game.

# Online User Documentation and Help System Requirements

None.

# Purchased Components

None.

# Interfaces

## User Interfaces

## Hardware Interfaces

No custom hardware interface is required.

## Software Interfaces

A custom android OS is needed.

## Communications Interfaces

No custom communication interface is required.

# Licensing Requirements

None.

# Legal, Copyright, and Other Notices

Copyright statements indicating content ownership shall be included in content as required by policy

# Applicable Standards

To be defined in subsequent phases.