DHPoly

For playing field surface

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 22/10/2017 | 1.0 | / | Alexander Busch |
| 03/11/2017 | 1.1 | Added link to Use-Cases | Sven Köhler |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms, and Abbreviations 3

1.4 References 3

1.5 Overview 3

2. Overall Description 3

3. Specific Requirements 3

3.1 Functionality 3

3.1.1 <Functional Requirement One> 3

3.2 Usability 3

3.2.1 <Usability Requirement One> 3

3.3 Reliability 3

3.3.1 <Reliability Requirement One> 3

3.4 Performance 3

3.4.1 <Performance Requirement One> 3

3.5 Supportability 3

3.5.1 <Supportability Requirement One> 3

3.6 Design Constraints 3

3.6.1 <Design Constraint One> 3

3.7 On-line User Documentation and Help System Requirements 3

3.8 Purchased Components 3

3.9 Interfaces 3

3.9.1 User Interfaces 3

3.9.2 Hardware Interfaces 3

3.9.3 Software Interfaces 3

3.9.4 Communications Interfaces 3

3.10 Licensing Requirements 3

3.11 Legal, Copyright, and Other Notices 3

3.12 Applicable Standards 3

4. Supporting Information 3

# Introduction

The only time that is wasted is the time when you think about time.  
To not let that happen, we will provide you with the best entertainment there will be.

That is what our Project "DHPoly" is for.

# Purpose

This SRS describes all specifications of DHPoly. Our game will be based on the usual Monopoly rules. To ensure that the game will hold completely new experiences, we will add dynamically occurring weather events.

# Scope

There are five major sectors we will be focussing on:  
**Playing field:**

The playing field will be different from all the other Monpolys´ playing fields, but it will contain all the mayor features one connects to a common Monpoly.  
**Surface:**

A nice looking surface design invites to have a game or two.  
Events: Interesting and crazy events and dynamic game changes provide a   completely new game experience.  
**Trading and making Business:**

When playing Monopoly playing and making the smartest decision for your Business Play a very important role.  
**AI (artifical intelligence):**

We want to make it possible, to Play Monopoly even when your friends are not around. Thatfor there will be the opportunity to Play against programmed robots.

## Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| SRS | Software Requirements Specification |
| UCD | Use Case Diagram |
|  |  |
|  |  |

## Overview

The following chapters will give you an overview about our project.

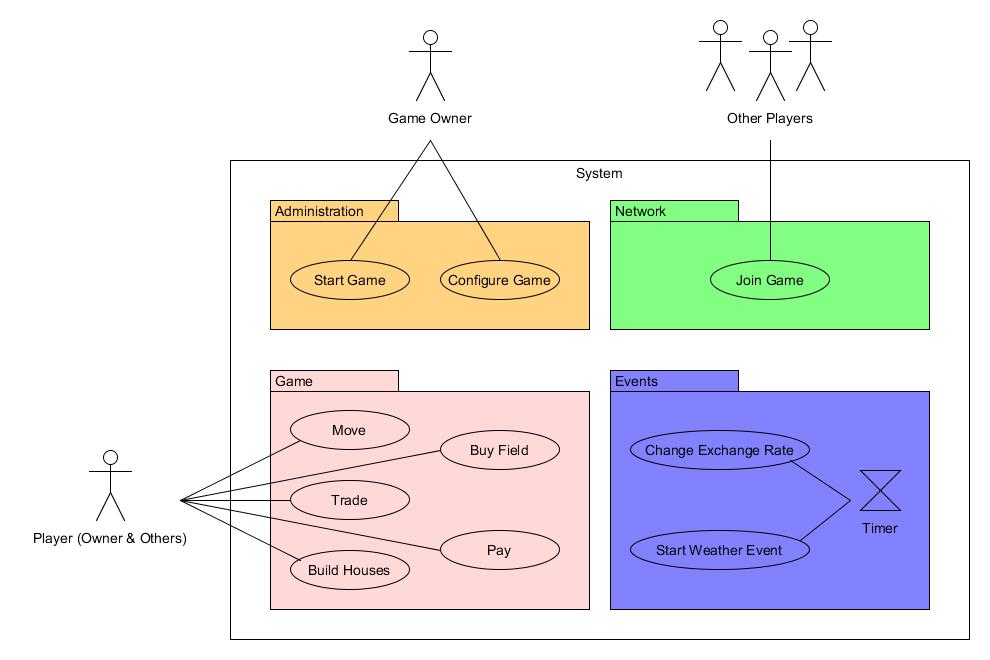
## References

|  |  |
| --- | --- |
| DHPoly Blog | <https://dhpoly.wordpress.com/> |
| DHPoly GitHub | <https://github.com/koehler1000/DHpoly> |
| DHPoly Use-Cases | <https://dhpoly.wordpress.com/2017/10/30/use-cases-and-prototype/> |

# Overall Description

As described in the introduction, we want to create a special version of Monopoly.

Our Use Case Diagram (UCD) can be seen below:



# Specific Requirements

In this chapter there are the requirements for each individual function in our project.

## Functionality

### Functionality Frontend: Game-window

In the game window you are able to play DHPoly solo or with other people. Before the game starts, you have to decide about the number of additional opponents. As a player you are able to move, trade with others, build houses, buy fields and trade with other players.

### Functionality Backend: Events

There are randomly occurring weather and exchange rate events.

### Functionality Backend: Administration

The game owner can start and configure the game.

### Functionality Backend: Network

Players can join a game through the network.

## Usability

Our game will have an easy usable graphical interface. Instructions and configurations are kept very trivial, so that it is very simple to start a game without the need to analyze the settings or anything else for minutes. However, there will be an amount of hints to make it even more accessible for new players.

## Reliability

The multiplayer function of the game is hosted on the computer of the game owner. Because of that the reliability depends on the host’s computer and network.

Whenever there is a bug, players can use our contact form to send us further information about the error, so we can analyze and solve it as fast as possible.

## Performance

Our game will react instantly to the inputs of the player. Our goal is to shorten the loading times as good as we can.

## Supportability

To be done.

### <Supportability Requirement One>

## Design Constraints

To be done.

### <Design Constraint One>

## On-line User Documentation and Help System Requirements

To be done.

## Purchased Components

To be done.

## Interfaces

To be done.

### User Interfaces

### Hardware Interfaces

### Software Interfaces

### Communications Interfaces

## Licensing Requirements

To be done.

## Legal, Copyright, and Other Notices

To be done.

## Applicable Standards

To be done.

# Supporting Information

To be done.