Team PB-J

CODENAMES GAME

Team Members

Name	ID Number
Ashesh Patel	???
Christophe Savard	40017812
Benjamin Thérien	40034572
Daniel Thibault-Shea	40073133
Shereece Victor	40105094
Michael Wilgus	29206388
Rezza-Zairan Zaharin	40003377
Steven Zanga	40000797
Mottel Zirkind	27206151

1 Project Analysis and Development Plan

1.1 Introduction

The purpose of this document is to detail the high-level requirements and features of the Codenames Game developed by team PB-J. The Codenames Game is a game of 2 to 8 human and AI players. This project is an adaptation of the Codenames Game designed by Vlada Chvátil and published by Czech games.

The specifics of how the Codenames Game fulfills these needs will be detailed in the use cases which more will be detailed in the upcoming design phase

1.2 Purpose

This document will describe the specifications entailed by the development of Codenames in compliance with the requirements of COMP 354. It will outline the high-level requirements encompassing user interfaces, product functions, user descriptions, assumptions and dependancies, constraints, specific requirements and an analysis model. The analysis model will hold use case diagrams, class diagrams, sequence diagrams and state transition diagrams.

1.3 Scope

This document only addresses the high level requirements of Codenames that the design phase.

1.4 Definitions and Abbreviations

1.4.1 Definitions

Board

The main playing area will be composed of a 5 by 5 grid of words. Each cell of the grid are representative of codenames of agents. Each cell in the grid will be defined in this document as a **Card**.

Card

The card will hold two values indicative of it's state: unrevealed and revealed. In it's unrevealed state, it will present a word taken from the game. In it's revealed state, it can either be a blue or red team's **Spy** card, a **Civillian** card or the **Assassin** card.

Assassin
The Assassin is a card type that when revealed, it would cause the team that revealed it to lose.

Spy

The Spy is a card type that belongs to either the red or blue team. Once all spy cards of a team are revealed, the team wins.

Civillian

The Civillian is a card type that occupies leftover space on the board that is not occupied by **Spy** cards or the **Assassin** card. When revealed, it only skips the revealing team's turn.

1.4.2 Abbreviations

To be updated.

1.5 Reference

To be updated.

1.6 Overview

The rest of this document outlines the problem description and the development plan.

The problem description will describe the game user's interfaces, product functions, user descriptions, assumptions and dependencies, constraints, specification requirements, and the analysis model.

2 Problem Description

2.1 Project Purpose, Scope, and Objectives

The objective of this project is to simulate a multi-user tabletop game named Codenames game by by Vlada Chvátil and published by Czech games. The rules of the game will intimately follow Vlada Chvátil with slight variations to accomodate the digital conversion. This project will be a multi-user game of up to four players composed of currently only computer players.

2.1.1 User Interfaces

Game Board

Card

Winner Interface

2.2 Product Functions

Every function below has to support system functions, such as a click of a button or revealing images when necessary. The following functions will be a part of the Codenames game.

2.2.1 Introduction

To be updated.

2.2.2 Board

To be updated.

2.2.3 Game

There will be two teams of **red** and **blue**. Each team possesses a pair of players each playing either the role of Spymaster of Operative. Since each team follows the same path:

- Team **Spymaster** reveals clue.
- Turn passes to Team's **Operative**
- Team **Operative** makes guesses based on clue given.
- Turn passes to opposing team's **Spymaster**

Input:
To be updated.

Action: To be updated.

Output:
To be updated.

Validity Check:
Sequence of the players is to be followed according to the order set earlier established. When a team finishes their turn, the next team's turn becomes active.

2.3 User Description

2.3.1 User Environment

2.3.2 User Profiles

2.4 Assumptions and Dependencies

2.5 Constraints

Assuming that a majority of player's PCs will run by the Windows OS, the project must be written in this platform that supports GUI(Graphical User Interface). Thus, the decision has been made to use :

- JAVA as the programming language.
- SQLite as chosen for data storage.
- JUnit for unit testing.
- Eclipse as the integrated development environment(IDE).

2.6 Specific Requirements

2.7 Analysis Models

2.7.1 Use Case Diagrams

The following diagarams will help provide an overview of the functions in the game. They describe the action that a player can perform, as well as the interaction between some of the system functions, ... To be updated.

2.8 Use Cases?

2.8.1 Use Case Details

Use Case 1: Start Game

Description	The user commences the game
Actors	User
Pre-Conditions	None
Basic Path	• The user clicks "Start Game".
Alternative Paths	None
Post-Conditions	• The Board is initialized. • The first Spymaster can reveal a clue.
Related Use Cases	
Used Use Cases	None
Extending Use Cases	None

Use Case 2: Reveal Clue

Description	The Spymaster issues a clue
Actors	Spymaster
Pre-Conditions	• The Board is initialized. • It's the Spymaster's turn to play
Basic Path	1. The word which comprises the clue is displayed 2. The number of cards related to the clue is revealed 3. 3. The system checks to see if the clue is valid
Alternative Paths	Alternative 1: • If the clue is not valid, a card belonging to the opposing team is revealed. • The turn is passed to the opposing spymaster. Alternative 2: • The clue is valid • Game play continues
Post-Conditions	• A clue has been revealed • The Spymaster's turn has ended
Related Use Cases	
Used Use Cases	None
Extending Use Cases	None

Use Case 3: Card Reveal

Description	The operative picks cards to be revealed
Actors	Operative
Pre-Conditions	• The Spymaster has given a valid clue and number of guesses • It is the operatives turn to play
Basic Path	1. The operative picks a card on the board based on the clue 2. The system reveals the contents of the card 3. If the card chosen belongs to the operative's team. The operative's reveal count increments. 4. The operative gets to reveal another card
Alternative Paths	Alternative 1: • If the operative has depleted their chances to guess, they cannot reveal another card. Alternative 2: • If the operative reveals the opposing team's card; the opposing teams reveal count is incremented. • The operative's turn ends. Alternative 3: • If the operative reveals a civilian card: • The operative's turn ends Alternative 4: • If the operative reveals the assassin, the game ends. • The operative's team loses
Post-Conditions	• A clue has been revealed • The Spymaster's turn has ended
Related Use Cases	
Used Use Cases	None
Extending Use Cases	None

Use Case 4: End Game

Description	The game is ended
Actors	System
Pre-Conditions	• The assassin card has been revealed • One of the teams has revealed all of their cards
Basic Path	1. The game board is hidden 2. A results screen is displayed
Alternative Paths	
Post-Conditions	• The game is done
Related Use Cases	
Used Use Cases	None
Extending Use Cases	None

2.8.2 Class Diagrams

Full Class Diagram

Simplified View

Hierarchical View

2.8.3 Sequence Diagram

3 Development Plan

This section of the document contains the estimated cost and schedule for the project in the form of a phase plan and project schedule. To be updated.

3.1 Project Estimates

3.2 Project Plan

3.2.1 Phase plan

3.2.2 Project Schedule

3.2.3 Project Resourcing