Software Requirements Specification: Code Quest - An Educative Coding Game

Mohamed Amine El Bacha and Younes Menfalouti

UM6P - Cyber Security Department

October 6, 2025

Contents

Chapter 1

Introduction

Code Quest is an educative coding game designed to make programming accessible and interactive for beginners. It combines a visual game environment with code-based commands to teach logical thinking, problem-solving, and algorithmic skills. Players use simple code syntax to move characters, solve challenges, and progress through levels — all while learning fundamental programming concepts.

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to outline the functional and non-functional requirements of Code Quest. It provides a detailed overview of the system's behavior, constraints, and objectives to ensure consistent understanding between the development team and stakeholders.

1.2 Scope

Code Quest focuses on providing an engaging learning platform for coding through a game-based interface. The project aims to target students, educators, and individuals new to programming who want an intuitive, interactive, and motivating way to practice coding.

1.3 Target

The target audience includes:

- Students learning programming fundamentals.
- Educational institutions integrating gamified learning.
- Self-learners looking to enhance logical thinking.

The target platform is a cross-platform application, deployable as a web app and desktop software.

Chapter 2

Requirements Specification

2.1 Gameplay Engine

The gameplay engine ensures smooth control, collision detection, and responsive interactions within the environment. It is responsible for maintaining player movement, level validation, and feedback mechanisms.

Table 2.1: Gameplay Engine Requirements

Type	**Description**	**State**	**Priority**	**Target**
Functional	The system shall run the main game loop	Proposed	Must have	V1
Functional	The system shall detect collisions	Proposed	Must have	V1
Functional	The system shall verify level completion	Proposed	Shall have	V1
Functional	The system shall display outcome feedback	Proposed	Could have	V1
Non-Functional	The system shall manage invalid input safely	Proposed	must have	V1

2.2 Command System and Code Execution

This module is the core of the learning experience. It allows players to input, validate, and execute commands that translate into actions within the game. It ensures code safety, scalability, and visual feedback for each executed command.

Table 2.2: Command System Requirements

Type	Description	State	Priority	Target
Functional	The system shall accept player-written Python-like commands.	Proposed	want to have	v1
Functional	The system shall parse and validate commands for syntax and parameters	Proposed	Must have	V1
Functional	The system shall execute commands step-by-step with visual movement	Proposed	Shall have	V1
Functional	The system shall support multiple commands per level	Proposed	Could have	V1
Functional	The system shall provide comment and suggestion features	Proposed	Want to have	V1
Non-Functional	The system shall execute commands safely without allowing arbitrary code execution.	Proposed	Must have	V1
Non-Functional	The system shall support scalability for adding new commands	Proposed	Shall have	V1

2.3 Progress System and Saving

This subsystem manages player data, progress tracking, and save/load functionality to maintain continuity between sessions. It ensures that data integrity is maintained and that user experience remains consistent.

Table 2.3: Progress System Requirements

Type	**Description**	**State**	**Priority**	**Target**
Functional	Save player progress locally	Proposed	Must have	V1
Functional	Allow progress reset on request	Proposed	Shall have	V1
Functional	Provide simple menu navigation	Proposed	Could have	V1
Functional	Maintain data integrity in save files	Proposed	Must have	V1
Non-Functional	Ensure portable save files	Proposed	Shall have	V1
Non-Functional	Ensure data persistence during closure	Proposed	Must have	V1
Non-Functional	Provide responsive UI interactions	Proposed	Could have	V1

Chapter 3

Use Case Diagram

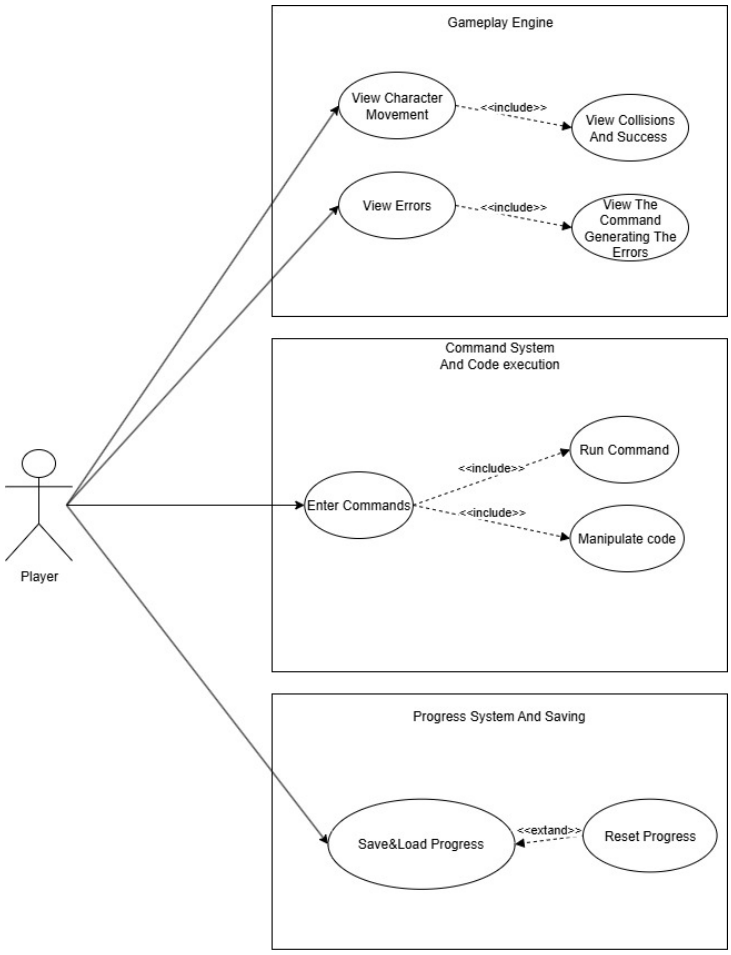

Figure 3.1: use diagram for code quest

Chapter 4

Conclusion

This Software Requirements Specification defines the complete vision and functionality of the Code Quest system. By adhering to these requirements, the development team ensures that the project remains structured, scalable, and educationally impactful.