15.03.06 - Mechatronics and Robotics

Focus (profile): Artificial Intelligence

SPECIFICATION

Job topic: **'THE FALL'**

Roman Tsaregorodtsev, 23930

Aleksey Spirkin, 23930

Novosibirsk

2024

TABLE OF CONTENTS**ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.**

1   TERMS AND ABBREVIATIONS

CdM-8 Mark 5 Processor	Coco-de-Mer 8 Processor, Mark 5
Harvard architecture	System architecture with instructions and data memory sharing the different address spaces
Hitbox	An invisible shape commonly used in video games for real-time collision detection
I/O	Input/output
Von Neumann architecture	System architecture with instructions and data memory sharing the same address space

2 INTRODUCTION

This document (hereinafter referred to as the Specification) is designed to describe development aims for the program product 'The Fall'.

Specification as the part of documentation and the content of Specification meet the requirements to the Digital Platforms project activity. All the presented work and its further result are presented as the project work for the Digital Platforms subject.

3 PURPOSE AND AREA OF APPLICATION

The program product 'The Fall' (hereinafter referred to as The Fall) is the quick-time action game aimed to interactively increase the reaction of a player. In addition, The Fall is meant to also increase the player's decision-making speed.

The Fall is open to free use and further distribution within the limits of Digital Platforms education program.

4 DEVELOPMENT REQUIREMENTS

The development process of The Fall is required to be split into two main stages:

– Development of circuit logic (see the Section 4.2);
– Development of software instructions to be implemented (see the Section 4.3).

Following these stages, the final program product should be presented as the independent computing system with pre-loaded software instructions, being capable of showing the gameplay and allowing the player's interaction.

The technology stack required for development is:

– Logisim for designing the circuit logic and implementing processor into them;
– CocoIDE for writing software instructions, compiling them and creating memory images.

The technology stack can be expanded by choice if it's necessary for development processes.

4.1 Gameplay

As the game is started, the player's goal is to pass through as much upgoing platforms as possible. Platforms should be passed through the gaps in them. To reach the gaps, player should use Left or Right keys, allowing to move along the platforms.

Besides platforms, the lines of coins are moving upwards, in the distance of player's hitbox. To improve the score, the player should collect the coins.

Apart from coins, reaching the lower border of the display should also increase the score. It's important to create a solution to prevent overuse of this option.

As the game continues, it should increase the speed of platforms, coin lines and character.

The game should end by pushing 'Start/End' button whilst the game is on or if the character has reached the upper border of the display.

Depending on how high is the player's score, it can be shown on the leaderboard. The best 3 results should get written in memory of the computing system and automatically shown on the leaderboard.

4.2 Technical requirements

To fulfill the gameplay experience, the design of The Fall's computing system should have the next hardware modules:

- CdM-8 Mark 5 Processor or another version of CdM processor, if necessary, following by tools accompanying its work;
- Memory banks with their architecture to be chosen (Von Neumann or Harvard);
- I/O bus, connecting the circuit logic with processor and memory banks;
- Several displays:
    a) Main display, showing the gaming process;
    b) Leaderboard display, including the score counter;
    c) Game status display, allowing to see the current state of the game (when the game is started, ended or paused);
- Controlling buttons for the following actions:
    a) Starting/ending the game;
    b) Controlling the player's character;
    c) Pausing the game;
- Circuit structure, controlling the in-game surroundings (platforms, coins, player's character, their speed);
- Circuit structure, connecting the above-described buttons to other processes.

4.3 Functional requirements

The software instructions should handle the processes of getting a new score value after the game ends, comparing it with values on the leaderboard and changing the leaderboard.

5 DOCUMENTATION REQUIREMENTS

At the end of development process, it is necessary to create the Explanatory note with further descriptions of developed solutions. The content Explanatory note should be fully based on Specification requirements and recommendations.

6 STAGES OF DEVELOPMENT

It's recommended to split the development process into several consequent stages. These stages are listed on the table below (Table 1).

Table 1 – Stages of The Fall's development process

Stage #	Stage name	Stage description
1	Circuit logic development	This stage requires development of the circuit logic modules that was mentioned in The Fall's technical requirements (see the Section 4.2).
2	Circuit logic QA	This stage requires testing modules working independently from processor for their quality assurance.
3	Software instructions development	This stage requires development of software instructions for proper processor integration.

4	Processor integration QA	This stage requires autonomous testing of software instructions and full testing of hardware with instructions pre-loaded into memory.
5	Explanatory note designing	This stage requires preparing the Explanatory note for mentioning all the architectural decisions during development.
6	Acceptance	This stage requires presentation of The Fall to assure the requirements fulfillment and to demonstrate the operation of it.