Requirements and Analysis Document for Tyler4

Version:

Date:

Author:

This version overrides all previous versions.

1 Introduction

The main reason behind the creation of miniMiner is the common interest of platform games. All four members of the group liked the idea of a platform-digging game, and therefore, the concept of miniMiner emerged. The essential goal of the application is entertainment and allowance of “micro-breaks”.

1.2 Definitions, acronyms and abbreviations

To be defined

2 Requirements

2.1 User interface

The application is based on a 2D-platform GUI, with the user digging downwards into a dirt-like environment. The UI will contain a joystick in the bottom right corner that controls the miner. A shop will also be added to the game which interacts with the user’s drilling machine.

2.2 Functional requirements

The user will be able to move, dig and fly until the fuel gauge is empty. On the surface, the user is allowed to sell materials, purchase new equipment, as well as refuel and repair the vehicle. The world in which the player will move in is gravity-based, so there will always be a downward-facing force forcing the vehicle back down to solid ground.

The user should be able to

1. Start a game or change options in the starting menu
2. Play through the game which involves
    a. Moving around the map
    b. Digging through the ground
    c. Gathering minerals
    d. Selling minerals
    e. Upgrading fuel and hull and drill parts
    f. Take damage
    g. Fill fuel and fix the drill
3. Exit the game
4. Credits(?)

2.3 Non-functional requirements

2.3.1 Usability

The game focuses greatly on usability. The game language is English centered.

The user will get a brief rundown of how the program is used and further knowledge will demand some intuition from the user.

2.3.2 Performance

The game will be constructed in a way which will reduce the amount of memory used to process the game which will be very important for a mobile phone user. Input from the user should be almost instantaneous with very little delay(if any at all), and will respond immediately graphically as well. Since it is a offline based game, there will be no network and therefore no latency.

2.3.3 Supportability

The program is mainly focused to be supported for an android mobile device, but will also be supported for a desktop computer. The GUI is constructed in a way to support a mobile screen but will work on a desktop accordingly.

However, the joystick will not be effective in a desktop playing environment.

2.3.4 Implementation

The program will use the Java Environment in which the user running the program will be required to have a device that can run Java programs.

TODO

3 Use cases

3.1 Use case listing

@Use case doc

4 Domain model

Link to UML: https://drive.google.com/open?id=0B2Q_zO7v9KTdR1VSTy1PWFVVS2M

4.1 Class responsibilities

TODO

5 References