Software Requirements Specification

CS101 Projects 2014

PAC-MAN
GROUP CUSE

ADITYA AGARWAL-14D110015

RISHABH JAIN-140110065

RISHABH DOSI-140110066

HARSHITA MOTWANI-140110075

...
...

Table of Contents

1 INTRODUCTION

This document gives a small overview over the popular arcade game Pac-man.

It also gives the details about the user interfaces and their specifications.

1.1 Definitions, Acronyms, and Abbreviations

GUI-Graphical User Interface.
OpenGL-Open Graphic Library.

1.2 References

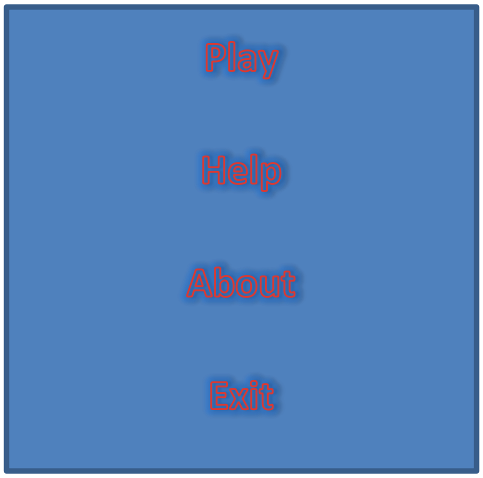
- https://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0CB0QFjAA&url=http%3A%2F%2Fdownload2.polytechnic.edu.na%2Fpub4%2Fsourceforge%2Fp%2Fpa%2Fpacman2family%2FPacMan2.pdf&ei=n70CVbGmH82OuATexYDQBQ&usg=AFQjCNEg4DzSi9YhtawFRvFLTRScHMdFow&sig2=AlhsdIIqwk6vlUNa_cFjIA&bvm=bv.88198703,d.c2E

- https://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&cad=rja&uact=8&ved=0CDIQFjAC&url=https%3A%2F%2Fwww3.ntu.edu.sg%2Fhome%2Fehchua%2Fprogramming%2Fopengl%2FCG_Introduction.html&ei=Or4CVei6J8yjugTz-oGIBQ&usg=AFQjCNET0O27e286BY6V816_cI56aVDd-Q&sig2=MDl8bSLnrvB5NWAjDeJsiA&bvm=bv.88198703,d.c2E

- https://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&cad=rja&uact=8&ved=0CCgQFjAB&url=http%3A%2F%2Fstackoverflow.com%2Fquestions%2F8733663%2Flearning-modern-opengl&ei=f74CVe_lMMO0uASg7oDICw&usg=AFQjCNHPXZ4RHB1YiTxbzXUg5o-FnHW6Ig&sig2=csrMYTlrWBTF8bz4GvyZvg&bvm=bv.88198703,d.c2E

2. Overall description

2.1-Perspective



2.1.1- Play

A. Easy

B. Normal

C. Tough

2.1.2- Help

Let's break it down as follows:

- Pac-man
- Pac-dots
- Ghosts

Pac-man is a yellow character who wants to get out of the playing area - called a maze. In order to do this it needs to 'eat' all of the little yellow dots -

Pac-dots, whilst avoiding the monsters, which are called Ghosts. If Pac-man touches a Ghost then one of Pac-man's three lives will be lost.

2.1.3- Game Rules

To further explain the rules let's break it down as follows:

- Pac-man, our hero, munches his way around the maze, eating all of the Pac-dots.
- In each corner of the maze there is a "Power Pellet", which when Pac-man eats one, the Ghosts turn blue or yellow for a limited time. During this duration the Pac-man can get extra points by eating the Ghosts.
- When the player accumulates 10000 points, he gets an additional life.

2.1.4-To move

The user has to navigate the Pac-man using arrow keys.

2.2-Product functions

The game provides user with varying difficulties with which user can play

It is a single player game

1. Maze-a predefined grid.

2. Levels-Higher the level, (i.e. easy, difficult or hard) more is the difficulty.

3. Movement-Pac-man can move in the four cardinal directions.

4. Cherry-Appears for the Pac-man periodically at random positions in the grid.

5. Collisions-collision of the Pac-man with ghosts leads to death.

6. Lives-Pac-man has 3 lives.

7. Score-Points are earned whenever food (Pac-dots) or cherry are eaten.

2.3-User characteristics

The game is for recreational purposes and may be played by people of all ages and professions. It has no limitations and no special skills are required to play it. It is in fact quite simple to play yet with sufficient challenge that keeps it interesting.

2.4-Constraints

There are presently no constraints.

2.5-Assumptions and Dependencies

2.5.1 The user should have working knowledge of computers.

2.5.2 The user should understand English.

2.5.3 The machine on which software is installed is having the minimum required hardware components.

2.5.4 This game is only meant for computers, cannot be run on mobile or any other device.

3. Details

3.1 Functionality

In Pac-man

- A Pac-man (protagonist, object) will be collecting food (Pac-dots or other objects).
- Some ghosts (antagonists, object) will chasing the Pac-man.
- Pac-man will have to collect all food before ghosts catch it.

3.2 Supportability

There are currently no issues related to supportability.

The computer only needs to support graphics (GCC compiler).

3.3 Design Constraints

There are presently no constraints on design and memory.

3.4 On-line User Documentation and Help System Requirements

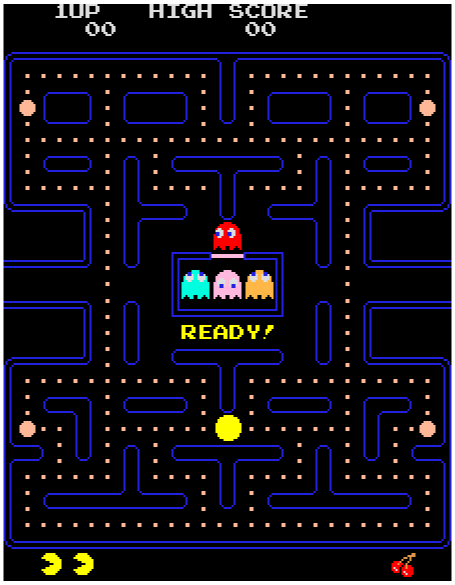
- https://docs.google.com/forms/d/1G0Gi4xSAYQwPULHoZBXzmlvtps_y0ZJcduV4ZyaeLPg/edit?usp=sharing
- http://forum.qt.io/category/35/independent-developers

3.5 Interfaces

3.5.1 User Interfaces

The software is based on GUI standards. The main frame consists of a background and buttons to select various maps.

Show high scores, etc. Interface of the game would be somewhat similar to the following screenshot



3.5.2 Hardware Interfaces

The program needs minimal specifications to run.

The input will be usual keyboard and mouse, and output will be through screen.

3.5.3 Software Interfaces

The game is constructed in a C++ compiler, namely Code::Blocks version: “svn build rev 7041”. “Allegro 5(generally used for making games, also contains ‘OpenGL library’” version: “allegro-5.0.10-mingw-4.7.0” is being used for the development of the code.

Allegro - The system will provide 2D vies rendered in allegro as a visual cue.

3.5.4 Communication Interfaces

Will provide some email address to give crash reports (computer generated, if any) if the project is well before time.

4. Quality Control

We will ensure that our Pac-man has all the features identical to that of a typical Pac-man game.

Such as-

- We will not move the ghosts randomly, we will make them to **CHASE** the Pac-man.
- At some places we will have dots with special power that will make ghosts unconscious (random movement) and weak so that Pac-man can kill them.
- We will have different levels in the game, higher the level ghosts will become faster and less number of special power dots.

We will add some additional features (IF POSSIBLE) as well such as-

- We will add constraint of time as well. We will have adequate time limit to complete the game.
- We will also have dots of special powers at some places, which on taking will speed up the Pac-man for a little time.
- We will have through channels (if Pac-man will go into a channel it will then emerge from the opposite channel).
- We will use advanced graphics (like allegro).

5. Risk Management

We might have many risks and we have planned to manage them like-

- If we by chance are unable to make chasing algorithm of ghosts on Pac-man, then we will at least ensure random movement of ghosts.
- We might be unable to add additional features, but we will at least somehow make a typical Pac-man game.
- We will certainly face problems related to bugs and errors, which we can solve by taking test cases.
- We might be unable to give superior or advanced graphics (like allegro), but we will at least give simple graphics.