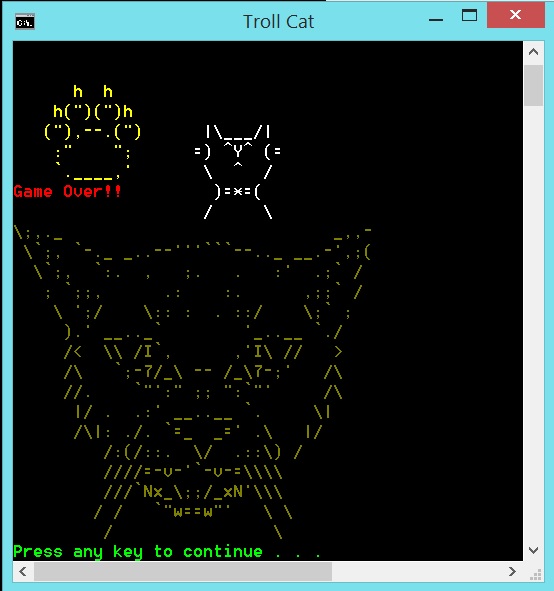
Laser Cat

Official Design Document

//insert the most updated picture here



Copyright 2014 Team 18

A Game Proposal By:

143343M Lim Wei Heng,

140522J Almeda Glenn Paolo Maestro,

143182M Toh Yi Fun Troy,

142471B Brandon Tan.

//Please add in a content page to show what is where. This part should be done last.

Target Audience

* 3 years old and above

Game Concepts

- Game Concept

- Genre

- Retro 1980s Handheld Game

- Game Goals

//If there are any more goals you can think up of, please write it down but keep it concise

- To allow players of the modern era experience the previous era of gaming

- Have a very basic understanding on cat behaviour (i.e. Cats get excited easily over moving objects)

- References

//any additional references available you may add it here

- Cat behaviour

- Laser pointers

- Octopus (1980s game)

Story

//you may want to add in a picture to illustrate the general idea of this ‘story’. A comic will do

- Genre

- Classical LCD game

- High Concept

- Players will play with the cats using their laser pointer.

- Synopsis

- You have decided that it is time to play with your cat. Using your laser pointer, you attract its attention with its bright light. Pulled by its magnificence, it starts to chase after it and it is up to you to keep it occupied by avoiding its impulsive attacks.

Gameplay

- Overview

- This section elaborates on general mechanics of the game.

- Abilities and Controls

- Basic Controls

//you will want to help update this component too based on the latest game updates.

- Directional pad to move the laser pointer which is represented as six red ‘0’s in a 3x2 rectangle.

- HUD

- Health Points is displayed on the top left corner of the screen

- Current Score is displayed on the top right corner of the screen

User Interface

//Be sure the information given here is correct to the final update. Otherwise, add/delete the lines to make the info here correct

- Overview

- Five main UI screens will be used: Title, Game, Instructions, Options and High Scores.

- Title

- This screen will host the following options: Start Game, Instructions, Options, High Scores and Exit screens.

- User is expected to enter a number to define the choice given.

- Instructions

- Self-explanatory, this screen teaches the player on how to play the game, and tells of the power-ups and hazards the player will face.

- Game

- This screen is the main gameplay of the game. The player will get to play the game until he loses all his health points or he decides to quit the game.

- Options

- This screen will lead to the following options: Difficulty settings and Credits.

- Difficulty allows the player to choose between an easy game and a harder game.

- Credits show the names of the team behind the game.

- High Scores

- Also self-explanatory, this screen shows the player the high scores achieved by the people before this.

Technical Design

//Same instruction as User Interface. Any additional features that I have missed out please add in

- Overview

- To create the game, we made use of Microsoft Visual Studios 2010 and Notepad programs

- Microsoft Visual Studios 2010

- The bulk of the programming has been done in this platform.

- Making use of the provided framework (SP1), we worked on the codes related to game mechanics and dynamics

- Notepad

- Text files is used to help print out aesthetics-related material for the game

- Animation

//Update this section to the latest version

- Overview

- This section elaborates on the animation schemes used in this project

- ASCII Art

- Codes related to the ASCII family is used to provide visual appeal. Most art is related to the cat with the exception of the player’s laser pointer

- Colour Coding

- Colours are used to help indicate the current game status to the player.

- User Interface

- A set of colours was chosen to help identify Health Points and Score. Depending on the chosen difficulty, the score will either increase slowly or quickly, based on elapsed time.

- If the player gets hit by the cat tail or cat paw, the Health Points will deplete for as long as it is touching the tail/paw.

- Frames per second

- Expected amount of frames per second is 5. Certain flickering effects are expected but it should not affect the gameplay adversely.

- Rewards

//Given that there are power ups, you may want to add in these features here. Be sure to provide sample screenshots about this component

- Overview

- The score is increased according to selected difficulty.

- The score can be influenced by a certain power ups

- The cat paw will move at random positions of the game space in a progressive manner.

- Power ups

- There are three power ups – Health, Immobilize and Invincibility

- Health adds current health points to player by 10 but is capped at 50 health points at any one time

- Immobilize will force the player to freeze on the spot but will have their score drastically increased.

- Invincibility will allow the player to move anywhere and ignores any damage received by cat tail or paw. Lasts for about 3-4 seconds.

- Camera

//this component may be redundant. It is up to you whether you want to keep this or not.

- Overview

- The camera angle is considered to be overhead. It is not adjustable.

- Sound

//this component is outdated. Please update it accordingly

- FX

- Beeping sounds upon selecting different options in the menu.

- Higher-pitched beeping if the player touches a hazard. Same beep used for touching power-ups.

Directory

//this section needs to be filled up on where you can find what. This component is necessary.

* Overview
  + This section contains information where the files are located
* The Framework folder holds the basic framework of the game. Inside are console.h, timer.h, console.cpp and timer.cpp.
* The programming for the credits is stored inside credits.cpp and credits.h.
* The programming for the difficulty is stored inside difficulty.cpp and difficulty.h.
* Important headers like <iostream> are stored in a global header file, global.h.
* Controls, rendering of the game, item and enemy mechanics are stored in game.cpp and game.h.
* The programming for the high scores is in highscore.cpp and highscore.h.
* The programming for the instructions is in instructions.cpp and instructions.h.
* Main.cpp receives player input, updates the game, and limits the frame rate through a main loop. It also initializes the variables, cleans free memory, and runs the menu loop.
* Said menu loop is stored in menu.cpp and menu.h.
* Another such loop is in options.cpp and options.h, for the options submenu.

Separate from the programming are .txt files that make up most of the graphics via ASCII characters arranged to form art. Collectively, they are:

* Character.txt
* Credits.txt
* Dead.txt
* Deadspace.txt
* Difficulty.txt
* Highscore.txt
* HS.txt
* Instructions.txt
* Item.txt
* Item1.txt
* Item2.txt
* Item3.txt
* Lpaw.txt
* Mpaw.txt
* Options.txt
* Spaw.txt
* Tail.txt
* Title.txt

- Development

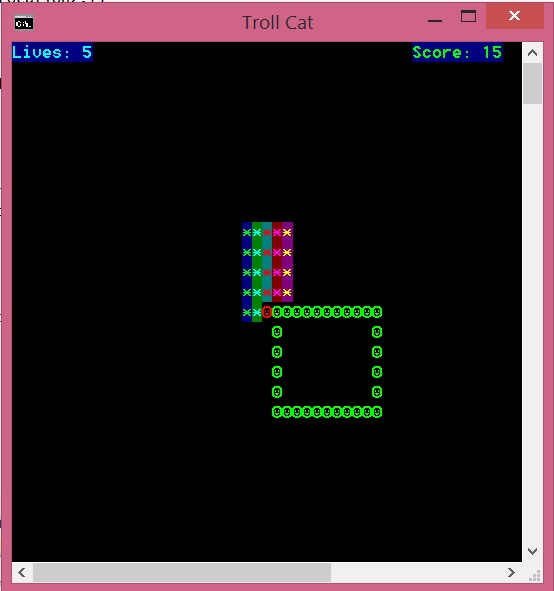
//this component may need some updating. Do provide a set of pictures of our progress based on days. If that is too much, you may provide pictures of progress in a gap of 2 days (i.e. if you placed Monday screenshots, the next phase set of screenshots should be from Wednesday and so on.) Please provide a brief description of everything here. If you run into issues, please let me know.

- Overview

- The development of the game is documented in this section.

- First Phase

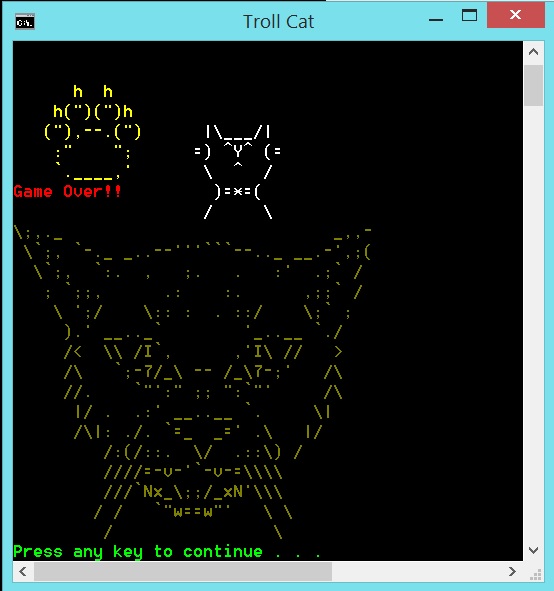
- This is when our game production was in its early stages. We established the fundamentals of the game here.



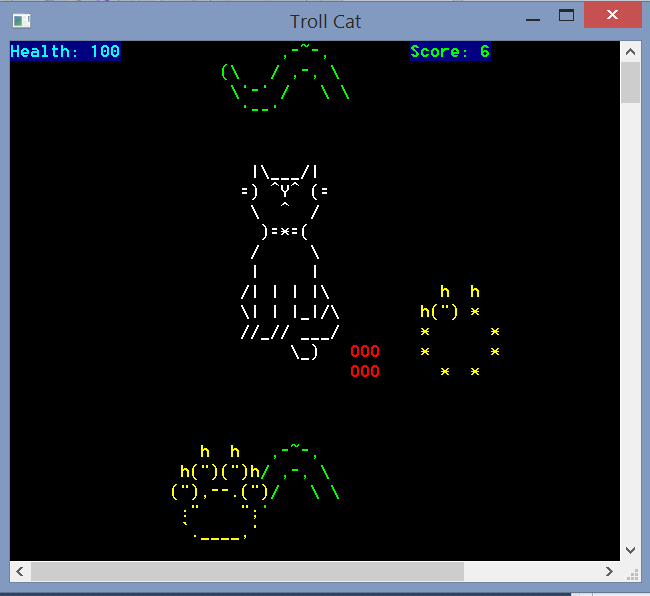
* Red dot = Player’s position
* Green square = Danger zone
* Coloured square = Dead zone (players cannot access this area)
* Second Phase  
  - The game mechanics and aesthetics have been improved.



* Game Over screen
  + We also included the “Game Over” component when all lives have been depleted



* Phase 3
  + The general sense of difficulty and the indicator for player’s life has been changed and increased.



Game Guide

* Overview
  + This section will provide some information to help players to get started.
* Basics
* Your character sprite is the bright red rectangle.
* Use the arrow keys to move around.
* The growing circles are the shadow of the cat’s paws. They don’t hurt you, but warn you of their incoming presence.
* Tails are constantly sweeping the area. They will hurt you if you touch them. Avoid.
* Small power-ups will constantly appear. Each will grant you five points and an additional effect.
* Basic Tactics
* Despite the constant danger, you must know when to move, and when to freeze. As the cat paw positioning is completely random, careful observation and quick reflexes are key.
* Power-ups are there for a reason. Always be on the lookout for them, as they give you five points each when you pick them up in addition to their effects.
* Always keep your health as high as you can, even if you have to sacrifice some of it. Some health gained is better than none.
* Advanced Tactics
* Carefully judge whether it is a good idea to pick up the phase or immobilise power-up. The points you can gain without them may be more than you can earn after getting them.
* There are a few blind spots for the tails where they do not sweep. Make use of them for a quick immobilise if you feel lucky.
* Do not rely on the blind spots for quick breathers. The random cat paws may swarm the place, forcing you into the tails’ path.

To add:

1. Quality Assurance – what do we do to ensure that our game will consistently perform at its peak (point out certain features of our game – highscores we check for spaces and replace with underscore)
2. Reviews – What some of our GDT friends say about our game. Good and bad, all should be included
3. Game Guide – What a player NEEDS to know before we play the game. This component is critical for those who see this game for the first time.
4. Game Flow – Which part of the game menu directs you to where.
5. Any additional content – What you need to know can be referred to in the sample GDD given in POGD assignment 3 package.

Note:

1. Ask and ask often. If you do not know how to deal with certain parts, please let me know and we will work out a solution.
2. Don’t be afraid of mistakes. What you think is valuable may not be the case with the rest of us. In such situations, be sure to justify your actions if you really want it in there.
3. Game Guide is best kept concise. If you can’t do it, just do whatever you can first and leave the rest to trim it.

Technical Design Document