**Project Title:**  ICT1009 Java Game Project

**Name of Game:** Monkey Invaders

**Game Objectives:** To pan across a screen and shoot descending swarms of monkeys, preventing them from reaching the bottom of the screen.

**GamePlay:**

The player controls a Plane that moves back and forth on the X-Axis near the bottom of the screen. The player one has three controls, left, right and fire. The player's objective is to destroy all of the Monkeys before they reach the bottom of the screen. There will be about 40 Monkeys on the screen at once. The Monkeys also move along the X-Axis, all at the same rate of speed. As time progresses the Monkeys move down the Y-Axis towards the player. While the player destroys more Monkeys, the Monkeys move faster. The Monkeys drop bombs at the player which will kill the player if they are hit. Once all of the Monkeys in a mission have been destroyed, a new mission begins with the Monkeys slightly lower. There is no winning for the player as the player will always die at some point in the game. The goal is to get the highest score.

This game will be adapted to be multiplayer by allowing each play to control with a set of keyboard keys and split-screen.

There will be a live scoreboard displaying the scores and life of each player.

**Additional Gameplay Ideas:**

Include special Boss Monkey. These would have higher hit points(requiring two or three missiles to destroy), move faster and have a faster rate of fire. They would also move more randomly, not just from the left side of the screen then back again. They would not descend, and therefore can only win by killing the player.

Pros: Adds a challenging aspect to the game.

Cons: Getting the balance right would prove difficult.

**The basic rules of the game are as follow:**

* The Player can only move along the X-Axis on the ground.
* The Player will have a limited speed, which might change according to the difficulty level.
* The player will fire missiles at the Monkeys. These missiles have limited velocity and limited rate of fire, both of which may vary during the gameplay.
* The player has an unlimited amount of missiles.
* If a missile hits a Monkey, the Monkey will die. This may not include a special kind of Monkey, such as a boss Monkey who may have more hit points.
* The missiles cannot travel through the shields or Monkeys or Monkey Bananas or PowerUps if they are included in the game.
* The Player shall have three lives.
* The Player shall beat a mission when all of the Monkeys have been killed.
* The Player will receive certain point values for killing each monkey. Also, the Player will receive additional points for completing the mission quickly.
* The Monkeys will also move along the X-Axis. They will all move at the same speed in order to avoid a collision
* The Monkeys shall drop down the Y-Axis closer to the Player as time passes.
* The Monkey’s velocity shall increase as the number of Monkeys decreases. This shall be on a set amount, not increasing for each Monkey killed.
* The Monkeys shall drop bananas on the Player. If any banana hits the Player, the Player dies.

|  |  |
| --- | --- |
| **Must-Have:**   * Develop a game environment   + console or GUI   + Option to use pre-existing API’s with proper justification of usage * Multi-player game: 2/3 players * Separate engines:   + Game engine: * Player move left and right using the keyboard * Collision box * Plane and Evil Monkey   + Maths engine * Score-board   + Display lives and points * Player Lives   + 3 lives per player * Visually pleasing * Proper usage of OOP concepts   + Classes   + Objects   + Inheritance   + Polymorphism   + Error Handling * Reusability of code   + Modularity * Functionalities   + Various features of the game * Innovation   + Level personalisation | **Should Have:**   * 2 player game * Difficulty levels * Randomness |
| **Could Have:**   * Powerups * Change in bullets * Boss enemy (requires hitting specific hitboxes to take down) * Moving vertically | **Would Have:**   * Different types of “planes” (levels) * Different types of weapons |