|  |  |  |  |
| --- | --- | --- | --- |
| **CS102** | **Spring 2015/2016** | Project Group | 5H |
| Instructor: | **Özcan Öztürk** |
| Assistant: | Dorukhan Arslan |

|  |  |  |
| --- | --- | --- |
| **Criteria** | **TA/Grader** | **Instructor** |
| Presentation |  | a |
|  |  |  |
|  |  |  |
|  |  |  |
| Overall |  |  |

~ 2D Rocket Game ~

**Rocket Fool**

**Ömer Ekmekcioğlu, Alperen Erkek, Levent Berk Güngen,**

**Hakan Yaman Sözer, Hakan Türkmenoğlu**

|  |
| --- |
| Requirements & U.I. Report  ( First Draft ) |

# Introduction

## Initial Problem Discussion

Although it is a relatively new field today, space travel is expected to be an integral part of mankind’s future. Therefore, learning about it would be beneficial to everyone, especially if it increases the interest and investments in space programs. However, the physics involved in space flight can make it seem unearthly and difficult to visualize.

Science fiction films that involve space travel often show spacecraft zipping from one planet to another in (almost) straight lines. Although taking such straight paths is intuitive and theoretically possible, in practicality it is extremely inefficient for spacecraft (at least the ones of today that use propellants) because of the effects of gravitational forces.

In order to help space travel receive the attention it deserves and to prevent the misconceptions about it, it needs to be taught efficiently. A fine way to do this would be to show a simulation, or to boldly go a couple steps further and exhibit a game. As often suggested, well-designed educational games are great for teaching because they can make the teaching material both easier to understand via the aid of technology and more interesting via the element of fun. With this cause in mind, the Rocket Fool team plans to create this product.

## Product Overview

The product is

* a computer game, which can fit into the genres “simulation”, “educational” and “puzzle”;
* primarily about piloting a rocket from planet to planet in a 2D space environment with realistic gravitation while managing additional details like fuel consumption and avoiding obstacles;
* a desktop application, because a desktop applications this format is good for games that do not require online connection;
* targeted to a general audience although it would be most beneficial for students (middle school or higher).

# Similar Products Review

## Kerbal Space Program

*Kerbal Space Program* is an award-winning, independently-developed simulation game that allows players to control various aspects of a space program run by little green men called “Kerbals”, from financial policy management to space craft design, to flying these crafts across a virtual 3D solar system. \*\*cite

The game has highly realistic models of gravitation, thermodynamics, air resistance, collisions, material stress, and more. It serves as an excellent example of a game for space enthusiasts although it can be noted for having a steep learning curve (it *is* rocket science after all).

The game also has a version, *KerbalEdu*, specifically designed for classroom environments, reinforcing the idea of how games can be great for teaching real-world physics.

## Into Space 2

*Into Space 2* is a popular, upgrade-based flash game that involves piloting a rocket during its ascent into orbit in a 2D atmosphere. It has additional gameplay elements like power-ups, obstacles, or missions to enrich what otherwise would be a very repetitive simple task. \*\*cite

## Angry Birds Space

As a part phenomenal mobile device game series *Angry Birds,* the game *Angry Birds Space* is based around launching birds to attack pigs in a 2D space environment with little planets that affect the birds’ trajectories with their (extraordinarily) strong gravitation. It makes players have to think about the effects of gravity while trying to certain places in space in order to solve the puzzles. \*\*cite

# Product Features

## Core Features

* Approximately 10 levels
  + The levels have varying objectives to keep things interesting such as but not necessarily space exploration, cargo delivery, satellite positioning, racing other rockets, rescue missions, or searching for resources.
  + Objectives will be marked with beacons.
  + The levels are connected by a story line.
  + The level maps have several planets, moons, and obstacles.
  + Bodies have very high densities in order to reduce the otherwise astronomical distances while retaining strong gravitational forces.
  + No procedural generation
  + The window will be a “camera” centered on the rocket on all times while the entire level does not have to fit on the screen.
* The controls: The rocket can be tilted using the left and right arrow keys. The up and down arrow keys can be used to increase or decrease thrust respectively.
* The physics engine:
  + Realistic gravitation
  + Decrease in rocket mass as fuel is burned
  + Collisions: The rocket will be destroyed if it collides with anything, including planets (landing can only be initiated by heading to the right spot).
* Finite rocket fuel and fuel management (to make players care about efficiency)
* Information windows at starts of levels to teach about the game and the physics and mathematics involved:
  + Topics covered for sure are: Newton's law of universal gravitation, elliptical orbits and eccentricity, the Hohmann transfer, the bi-elliptical transfer, gravitational assists, gravitational potential energy and escape velocity.
  + Topics that would ideally be covered include: Newton’s laws of motion, Kepler’s laws of planetary motion, the delta-v concept, uniform circular motion.
* A toggled in-game display to show information like the forces acting on the rocket and possibly equations
* Take-offs and landings shown as basic cut scenes that the players do not control
* Trajectory estimations: The position of the rocket after X minutes can be calculated by the program, possibly while displaying the equations, and shown.
* Progress saving
* A pause menu

## Desirable Features

* Take-offs and landings shown as basic cut scenes
  + There will be multiple good or bad take-off/landing scenarios based on the player’s decisions made on a window before the launch (eg. how much cargo to pack, how much initial thrust to give, approach speed, etc. )
  + These decisions will be made on a pre-launch window where the player will configure the rocket according to suggestions made by the game and what the ğşayer thinks would do.
* Sound: Sounds for thrusters and for collisions/crashes. Music and interface-related sound effects can be added depending on what is available. They will be composed or used with appropriate licensing. If there is sound, a muting option will also be provided.
* Score keeping means (eg. based on remaining fuel or mission duration)
* Time-warping: speeding up time to quicken the travel over long distances (if the develops decide that it is worthwhile)
* Pop-up boxes with the “Did you know?” sort of interesting facts about space travel
* Enabling/disabling full-screen mode

## Optional Features & Possible Future Additions

* A branching story line
* Power-ups
* Multiple save files
* Intricate AI for NPC’s
* Some planets that have atmospheres with basically-modeled atmospheric drag or realistic atmospheric drag
* Rocket customization/upgrading options (in terms of aesthetics or engineering)
* A level editor
* Multiplayer features
* Achievements & stat keeping (eg. number of crashes)
* In-game shop
* Wormholes / black holes
* Adjustable difficulty (eg. different engine efficiencies)
* Limited electrical power

## Not Planned Features

* 3D graphics
* Thermodynamics
* Realistic lighting
* Orbital decay other than atmospheric drag
* Extravehicular activities
* Plane piloting
* Changing engine ISP’s
* Telemetry issues
* Random equipment failure

## Stylistic Choices

* Game title, slogan: (undecided)
* Graphics: Cartoonish, consistent graphics with basic animations for details like rocket plumes are preferred. They will be drawn or used with appropriate licensing. A rounded, simplistic, smooth art style with block colors is preferred.
* Sound: Medium/high-quality sound effects are preferred. Possibly, sound effects for aesthetics (eg. clicking buttons) will also be included.
* Setting: The game is set in a not-so-distant future where space travel is uncommon but not as rare as today. Planets may be based on our solar system in order to give information about them (or not).
* Characters: Characters may be based on Bilkent University students and faculty if needed consent and permissions are acquired. Some aliens may too be included. Characters are currently not expected to be very deep.
* Game’s overall attitude: humorous, informative, casual
* Music: Music that fits the setting and attitude will be sought.

# User Interface

* Once it is run, the game will not require any inputs until the main menu appears.
* Main menu
  + The menu will have buttons to each of these screens: Level list (play button), credits, quit, options. There will also be a mute button in the corner.
  + The user will left-click a button once to use it. No other inputs apply.
  + Menu design will match the art style of the game and be intuitive to use.
* Options menu
  + These options will be provided with single-left-click buttons: full-screen, erase save file, mute, back to menu
* Credits
  + All necessary accolades, disclaimers, and etc. will be displayed. There will be a single-left-click back button.
* Level list
  + There will be a button for each level and the back button, all single-left-click buttons.
* In-Game
  + The rocket can be tilted using the left and right arrow keys. The up and down arrow keys can be used to increase or decrease thrust respectively.
  + Mission objectives will be displayed on the upper left corner of the screen.
  + Stats, like the remaining amount of fuel will be displayed on the upper right corner.
  + The bottom-middle part of the screen will have a toolbar for various single-left-click buttons (see other bullet points).
  + The window will be a “camera” centered on the rocket on all times while the entire level does not have to fit on the screen.
  + The camera will have 2-3 zoom settings, each centered on the rocket. Alternatively, a mini-map may be used instead. The buttons for zooming, as well as the mini-map would be located at the bottom middle. The mini-map can be minimized to be re-opened from the toolbar.
  + The camera may rotate only by 180 degrees only when toggled by a button on the bottom middle. This would help confused players regain their orientation.
  + The button to toggle the overlay to display forces, as well as the one to calculate and display the rocket’s trajectory will also be on the bottom middle.
  + Pressing “Esc” or the spacebar will open the pause menu, which offers single-left-click buttons for un-pausing, returning to the main menu, and muting the game.
  + Additional text will appear as pop-up windows with single-left-click buttons on them to close them.
* Cut scenes
  + The cut scenes are expected to be less than 30 seconds long and users will be shown and given the option to skip them using the space bar.
  + Dialogue will appear in speech bubbles.
  + The cut scenes may proceed as comic panels or basic animations.
* Pre-launch window (if implemented)
  + Pressing “Esc” or the spacebar will open the pause menu, which offers single-left-click buttons for un-pausing, returning to the main menu, and muting the game.
  + A vertical slider on the right side of the screen will be clicked and dragged by the player to decide on the initial thrust given to the rocket.
  + Horizontal sliders arranged in a vertical list will be on the left side of the screen, with which the player adjusts values such as the amount of resources to pack.
  + A large single-left-click launch button in the middle of the screen will start a launch cut scene.

# Conclusion & Reflections

This report describes the motives for designing this game and it also provides a detailed outline of its features. \*\*complete this – ne yazcam ki buraya anlamadım

# References

Abc

# Concept Art

abc