Knight’s Guard

Concept of Operations

COP4331C, Fall, 2014

Team Name: Group 1

Team Members:

- Megan Postava
- Katie Jurek
- David Moore
- Miguel Corona
- William Adkins
- Jonathan Bennett

Modification history:

Version	Date	Who	Comment
v0.0	09/11/14	Jonathan Bennett	Imported template from previous classes
v0.1	09/12/14	Jonathan Bennett	Completed sections “The Current System” and “The Proposed System: Needs, Users & Modes of Operation”
v0.2	9/15/14	Megan Postava	Completed sections “Implementation” and “Operational Scenarios”
v1.0	9/16/14	Jonathan Bennett	Added minor changes to “Implementation” and final formatting.

Contents of this Document

The Current System

There is one existence of a UCF themed tower defense video game made by a group of previous students. It is for the mobile Android platform only, and available freely from the Google Play store. However, the game is very simple in functionality, and does not use the theme of UCF much except for the UCF logo and the idea of football players being the enemies. Otherwise, the map and method of attack is generic.

The Proposed System: Needs

In *Knight's Guard*, the UCF theme is a critical part of the game design. This game is of the tower defense genre with a UCF theme, where the goal for the player is to defend the UCF student union from waves of attacks by enemies that come from many different paths on the UCF campus.

The game map is the actual UCF campus map, units and towers are based on UCF campus buildings, and the language of the game parodies typical college student life (e.g. you can take a withdrawal to restart the level, your goal is to survive a set amount of time called a "semester", etc.).

For these reasons, Knight's Guard will be a better representation of the UCF theme. Additionally, the team intends to build a more complex and interesting form of gameplay, where the player has several options of towers, defends against enemies from multiple paths, and has the ability to earn in-game currency to purchase new upgrades and abilities. With these new features and a focus on creating a balanced difficulty of play, *Knight's Guard* will be a higher quality video game to play.

This video game is intended to be playable on Microsoft Windows and mobile Android platforms, with the priority being placed on finishing the game for the Microsoft Windows platform first. With two successful platforms available, the game will be available to play for more end users.

The Proposed System: Users and Modes of Operation

Users

- End-user: This user is the one who will play the game. They will download the game for free, from the game web site (for Microsoft Windows) or through Google's Play store (for Android). The user can play for as long and as many times as they want for no monetary charge (outside of their Internet provider in order to download the game).

Modes

- Standard Play Mode: The game has one mode for the player to use. This mode starts the game on Level 1. In order to advance to new levels, the player must successfully finish the level they are on. As the player progresses, they may gain new abilities in the game, but these abilities are reset whenever a new game is started.

The Proposed System: Operational Scenarios

Normal Use of System

Normal use includes downloading the game from the game website, opening the file, and playing Knight's Guard. The end user will be able to do all of these things. The game has one mode, and allows the player to start waves of enemies, place towers that automatically shoot enemies, place towers that can be clicked on for a more powerful attack, and monitor damage they have taken from enemies.

Example Use: Wanting to Play an Internet Game

A user wants to play a game from the internet. They find the game website and download Knight's Guard for free. The user opens the game, and begins play on normal mode. They are able to place a few towers, determined by their starting currency, and can click a button to start the first wave of enemies. Between waves, they can place more towers based on how much currency they obtain from the destruction of enemies.

Loss of Internet Connection

A loss of internet connection while playing the game will have no effect on the game itself, but the game will not be able to access the Leaderboard in order to view it or add scores. If the internet connection is lost during game download, the game must be downloaded again once internet connection is re-established.

Incorrect Input

If a user enters input that is not a game control while playing Knight's Guard, it will be ignored by the system and will not crash the game.

The Proposed System: Operational Features

Must Have:

1. Ability to run on Windows 7
2. Ability to place defensive towers in designated areas
3. Ability to start enemy waves
4. Ability to unlock stronger towers at higher levels
5. Able to see currency count, able to use currency to buy towers
6. Ability to see health of base, health of towers
7. Different types of towers; more powerful "click to use" towers
8. Different types of enemies with different sprites

Would Like to Have:

1. Portability, ability to run on Android
2. Online Leaderboard to compare scores

The Proposed System: Implementation

Technologies:

- GameMaker Studio
- GML (GameMaker Language)
- Google Docs
- GitHub
- Trello
- Apache web server, PHP & MySQL database (for online leaderboards)

The game will be developed using GameMaker Studio. It will be programmed using GameMaker's native language, GML. The primary release will be on Windows PC, with a secondary goal of porting it to Android. GML is a scripting language similar to Python, and so it is easy for the team to learn. In addition, one of the group members is highly experienced with GameMaker Studio and has written multiple games in it, including a Tower Defense game. For communication, the group uses Google Docs, and has a project repository on GitHub for code version control. For tracking tasks, the group is using Trello.

Releasing the game on Windows only is limiting in the fact that the game is not portable. To solve that, a secondary goal is release on the Android platform. However, that limits image file size and forces the team to make sure it can run smoothly on less powerful processors. Being that it is a simple game concept, though, it shouldn't be much of a problem in the long run. The primary platform, therefore, can be said to be stationary, while the secondary is mobile.

The game does not require an internet connection, as there is no "multiplayer" mode. Another secondary goal is a leaderboard in which the top players are highlighted, though that only requires internet connectivity in order to submit the score. The game itself does not rely on a connection. As for usage, users should not be playing this game any time they are doing something that needs their full attention, such as driving, walking, or being in lecture.

Choosing the development environment involved a long comparison period between GameMaker Studio and Unity. While a lot of Unity's features were considered to be convenient (built in source control, known as an industry standard), its 2D system was newly implemented, and the possibility of bugs still existing in the system was considered to be a large negative.

That being said, many of GameMaker's features are locked in the paid Professional version, such as the ability to port to the Android system. The group is in possession of a Professional License, and therefore is able to overcome this limitation. The group reached a conclusion to use GameMaker Studio due to the fact that one member was already highly experienced in the platform, and that it was easy to learn and has many tutorials online.