**Whack-A-Prof**

**FUNCTIONAL REQUIREMENTS**

**GROUP 2**

**10/26/2021**

**Version 0.1.1**

1. INTRODUCTION
   1. Purpose
2. **GROUP REQUIREMENTS OVERVIEW**
   1. General Group Responsibilities
   2. Assumptions/Constraints
3. **FUNCTIONAL REQUIREMENTS**
   1. Objective
   2. Logistical Coding Requirements
   3. Supportability/Game Controls Requirements
   4. Security/Version Control Requirements
   5. Assumptions/Constraints
4. COMPLETION REQUIREMENTS

4.1 Assumptions/Constraints

1. REFERENCES AND IMPORTANT LINKS

# INTRODUCTION

## 1.1 Purpose

To plan out and define the necessary requirements to provide code, version control, graphical design, and sound for our group “Whack-a-Prof” themed game term Project.

# Group requirements AND RECOMMENDATIONS overview

**2.1 General Group Responsibilities**

**Specs:**

* Track progress made by all teams and document it in these functional specifications sheet
* Collaborate with Backbone to determine general game setup and requirements
* Provide QA with all necessary documentation for version control
* Find and Discuss examples of “Whack-a-Prof” game implementations as a baseline for our game

**Graphics:**

* Find and or create the background image, along with the necessary digital assets to serve as the “holes” and “professors”
* Find and or create all necessary background and active game sound effects
* Create graphics for buttons, including play, pause, and more.

**Backbone:**

* Establish and code working builds necessary for the “Whack-a-Professor” game project.
  + This includes all necessary game functions regarding timers, lives, whacking, etc.

**QA:**

* Set up a repository for storage and updating of game files along with specifications through Version Control
  + Keep track of all Subversion Logs/GitHub changes
  + Manage and maintain organization of game versions in preparation for final release version

**2.2 Assumptions and Constraints**

* Any recommendations and or suggestions regarding the general structure of game functions are welcome, this is merely a general guide for group functions regarding the project.

# functional requirements And Specifics

## 3.1 Objective

With members being both the developers and end-user, the game will be structured based on our collective creative insight and knowledge regarding what a “Whack-a-Prof” styled game is.

* The game is meant to be simple, accessible, and most importantly **entertaining**
* The game should run as smoothly as possible given our short development time

## 3.2 Logistical/Coding Requirements

**Gameplay Algorithms**

**LIVES**

* Player will be provided with a “lives” system establishing a number of allowed misses before game termination
  + Suggested baseline of 5 lives (Subject to Change)
  + **When all lives lost:** Gameplay is terminated, A score is provided through a pop-up

**TIMER**

* The game is primarily time based
  + Timer will operate **concurrently** from the rest of the game
  + The timer will be set to around 60 seconds representing the amount of time the player can accrue points through whacking professors
  + **When timer ends:** Gameplay is terminated, A score is provided through a pop-up

**PROFEESSOR HIT/MISSES**

* A **hit counter** is necessary to keep track of points obtained per “Professor” whacked. A separate **miss counter** will also be necessary to track misses.
* When a mole is successfully hit via (onClick or eventListener) a hit is registered increasing the points counter
* When a professor is missed the *miss counter* will be deducted from until reaching 0. **Check Lives section for more info.**

**PROFESSOR/MOLE REQUIREMENTS**

* The professor will be the only non-static object within the interface, being the only thing that needs to move around for gameplay
* The professor will move between each table, throughout gameplay disappearing and reappearing to simulate movement
* Given our limited time **only one professor is necessary**, as long as the mole moves quickly enough between holes while still being hittable

**Miscellaneous Algorithms – Its up to the group to determine if we add these as they are mostly for fun/creativity**

* A Possible Boss Level
  + “Bosses” are professors who occasionally pop-up providing bonus points, yet **requires additional hits (2 to 4)**
* A Pause Button
  + All game functions will stop when paused including movement of “Professors” background and visual animations as well as music

## 3.3 Supportability/Game Control Schematic Requirements

**INPUT/OUTPUT**

* Gameplay will follow the usual setup for a Whack-a-Mole game for the sake of simplicity. The only necessary inputs will be **mouse clicks/movement**
  + Serve to both initiate/pause gameplay through buttons.
  + Control the initiation of a whack or miss
  + Correlate to the movement of the mallet.

(**Miscellaneous Functions:** Inclusion based on opinion of the Group)

* Controls will be executed using a **combination of mouse clicks and key tabs**
  + Each “hole” is to be assigned a specific key letter (A-Z) prompting the player to press the key associated with that specific “hole”
  + Pressing the correct key invokes a whack on the given “professor”. Otherwise, is a miss

## 3.4 Security/Version Control Requirements

**SUBVERSION AND GITHUB**

* Version control will be managed through a combination of our group **GitHub and Subversion Repositories**
  + Link for our repository can be found in the Group 2 Discord Channel
  + **ALL** changes to the schematics and or code for the project will be managed by the repositories, with **QA Team** ensuring that it works well and is organized
* **Subversion** will serve as the primary storage for all game related files at the request of the professor
* **GitHub** will be used for the sake of simplicity to easily get access to our files and or logs online when requested

## 3.5 Interface/Graphical Requirements

**BACKGROUND STATIC/NON STATIC IMAGES**

* The game is to be set in a **school inspired** theme, particularly a classroom setting
  + The game space will be comprised of around **6 to 12** tables which will serve as the “holes”
* All interactive and noninteractive objects not related to gameplay will be placed in the topmost area of the interface, this includes:
  + Start Button
  + Pause Button
  + Timer
  + Score
* The **background image** containing the desks and the correlating “holes” for the professors to pop up from can be set into a fixed position for simplicity
* The **moles** in our game will be substituted for images of professors
* The mouse cursor will be replaced with a **mallet** in accordance with traditional whack-a-mole games.

**SOUND DESIGN**

* The game will require an assortment of sound effects
* **Background Music:**  Loops throughout gameplay, stops when paused or game terminated
* **Hit sound effect:** Initiated when the player successfully hit a mole
* **Miss sound effect:** Initiated when the player does not successfully hit a mole
* Background music and whacking or miss sound effects can be anything that the **Graphics Team** wishes, get creative and have fun with it as we are the end-users.

## 3.6 Assumptions / Constraints

Detail all assumptions / constraints related to Functional Requirements and User Impacts.

* All detailed requirements are subject to change based on our project progress, given our circumstances drastic changes can be made to meet our deadline if the final product is acceptable based on the consensus of the group and opinion of the professor.

# Completion requirements

## 4.1 Assumptions / Constraints

* Given the amount of time provided for this “Whack-a-Prof” project, it without a doubt will not be perfect
* The general requirement for completion is that the game is **STABLE not FLAWLESS**, if we, being both the developers and end users are confident in our work after substantial testing will the game be considered “complete”

**APPENDIX A: REFERENCES or HELPFUL LINKS**

Listed below are all Links referenced in this Functional Requirements document.