# Design Document:

Yet-To-Be-Named Android Game

(... or Yetbenamdag)

## Minh Trung Pham

## October 21, 2012

# Overview

The project will consist of an Android Game in which users can shake their android device to execute moves but also yell at the device to skip levels.

# Project Definition

The game will be an Android Device Game in which the screen automatically scrolls while the player attacks oncoming enemies while dodging environmental hazards like pitfalls. The game will also utilize the Android accelerometer as a command for the in-game character to execute a super maneuver of some sort, unique to each character. Furthermore, the game will also utilize the microphone to detect user yelling. Upon reaching a ‘yelling threshold’ the game will end the current level, and skip the user to the next one. Users have the option of skillfully playing through each level to reach the end, or simply scream at their device to progress.

# User Stories

John Von Busy-Life twiddles his thumbs as he sits patiently at the Denver International Airport. Fatigued from his ventures with baggage and security, he awaits the arrival of his plane only to hear a loud smoke-crusted voice choke out the words of the planes delay. Nearby he hears the cries of children, and coughing and sneezing of elderly people. Aggravated, John pulls out his Android Samsung Galaxy phone and opens up his new favorite game. The main splash screen lights up an otherwise bleak setting and as he clicks play, the screen dissipates to show a selection of poorly-named characters, obvious rip-offs of famous videogame characters. Disregarding preference, John selects a random player and watches as the screen melts into a race-like setting. An overwhelming countdown timer ticks slowly towards 0 as John braces himself for the exhilaration to follow. Upon the countdown hitting zero, John's choice hero immediately sprints forward as the screen follows. John, having not touched his device yet, keenly watches the screens edge as he knows enemies will soon scroll towards him. As expected, a pixelated gremlin appears and begins scrolling towards John's hero. Lucky for John he is well prepared... he simply taps the screen and his hero executes a poorly animated attack which disposes the gremlin. A slow but satisfying smirk crawls across John's face, but to his surprise the scrolling ground in front of his character suddenly reveals a large gap. He quickly shakes the phone for his hero to execute a super jump but alas his reflexes were poorly timed and the hero plummets off screen, presumably to his untimely death. Outraged, John screams obscenities at his phone, which quickly draws the attention of the people around him. The game having detected John's vulgar swearing, suddenly skips the current level, awards him some completion points, and progresses John's character to the very next level. Taken back by this, John simply clears his throat, and continues the next several levels in similar fashion, cursing and shaking his phone to victory.

# Use Case List

1. PLAYER selects a CHARACTER - As a PLAYER, I want to be able to select a specific CHARACTER to play as.

The PLAYER of the game selects from a list of CHARACTERS which CHARACTER to play as.

The PLAYER, having selected a CHARACTER will proceed directly to the start of the game.

1. PLAYER encounters an ENEMY - As a PLAYER, I want to ENCOUNTER ENEMIES randomly, to make the GAME more dynamic

During the course of the GAME, the PLAYER encounters an enemy that scrolls towards the PLAYERS' chosen CHARACTER. The PLAYER can execute either an ATTACK or a JUMP to confront this ENEMY, or simply AVOID the enemy.

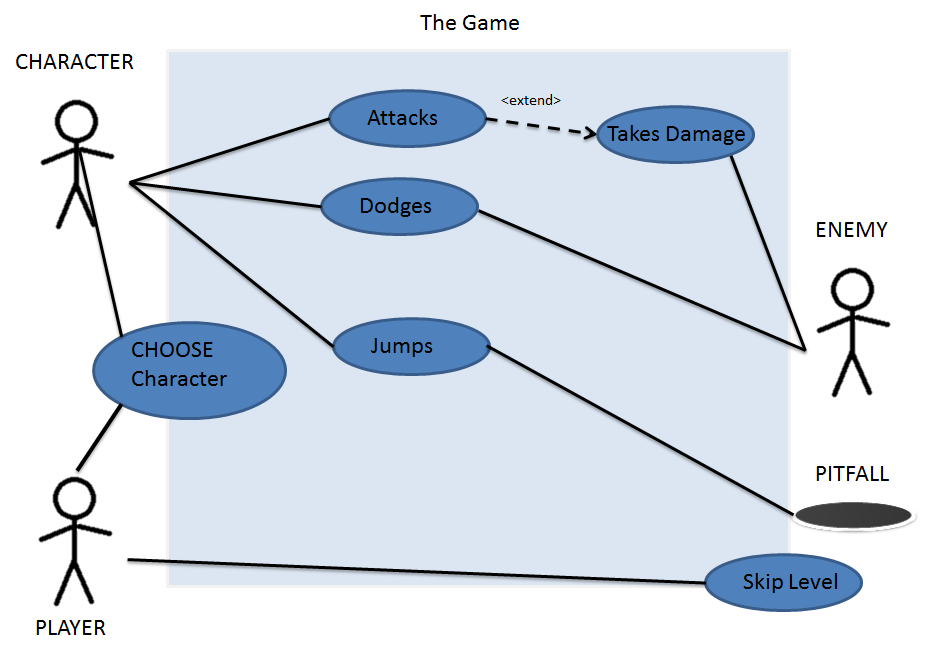
1. PLAYER encounters a PITFALL - As a PLAYER, I want to ENCOUNTER PITFALLS randomly, to make the GAME more fun.  
     
   During gameplay, the PLAYER encounters a PITFALL. The PLAYER must jump past this PITFALL, but should they fail the GAME will end, proceeding to the SCORE\_SCREEN
2. PLAYER ATTACKS an ENEMY - As a PLAYER, I want to be able to ATTACK an ENEMY to safely dispose of it.  
     
   Having encountered an ENEMY, the PLAYER ATTACKS the ENEMY. If the ATTACK connects and HITS the ENEMY, the ENEMY will DIE and pose no further threat. Should the ATTACK MISS, there is no further consequence.
3. PLAYER TOUCHES an ENEMY - As a PLAYER, I want ENEMIES to pose a real threat so that I may have incentive to deal with them.  
     
   Having encountered an ENEMY and FAILING to dodge/kill the ENEMY, the player touches the enemy and loses HIT POINTS.
4. PLAYER FALLS in a PITFALL - As the DESIGNER, I want there to be PENALTIES for players not avoiding the environmental hazards.

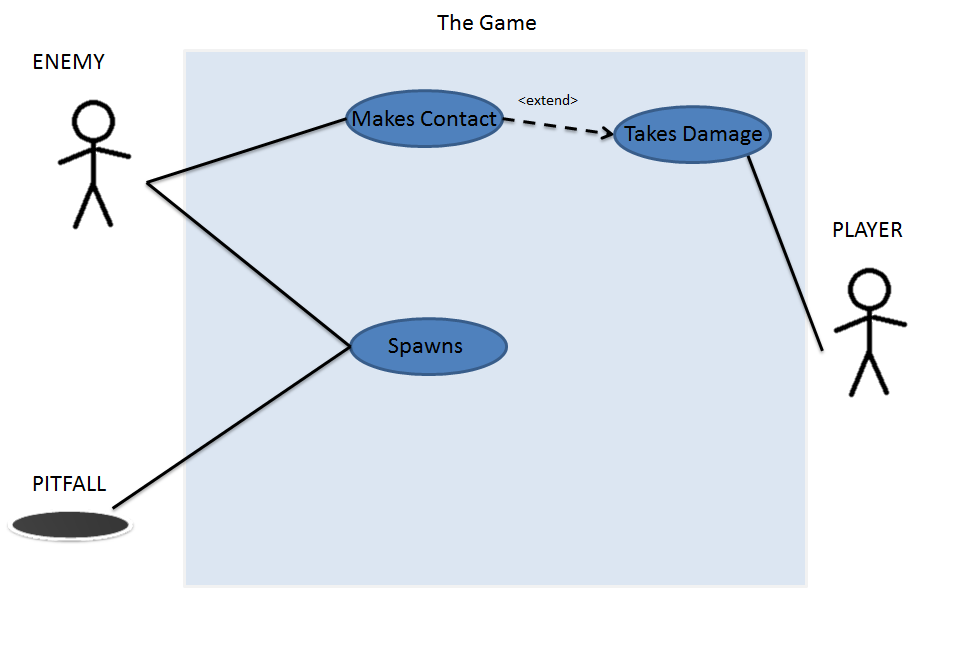
The PLAYER for any given reason, fails to dodge a PITFALL and falls in. The PLAYER loses all HIT POINTS and the game will cease, going to the SCORE SCREEN.

1. PLAYER loses all HIT POINTS - As the DESIGNER, I want there to be some method of LOSING the game.   
     
   Regardless of reason or method, the PLAYER has lost all HIT POINTS and at this point the GAME will end.

(I know it’s a little verbose, but I felt these use cases are the most important, and the game isn’t really described without all of them)

# Use Case Diagram





# PLAYER selects a CHARACTER

Title: As a Player, I want to be able to Choose a Character the Play the Game as.

Primary Actor: The Player

Precondition: The User has Started the Game

Postcondition: The Player has Selected a Character and the Game can now start.

Main Success Scenario: The Player has successfully chosen a Character with which to play the game.

Steps: 1. A list of characters as shown to the Player  
 2. The Player selects which ever character he/she wants  
 3. The Character list transitions to the Start of the Game

Variations: If any error occurs in the Player’s selection, the Game

Will default to the starting splash screen again.

Frequency: This will occur Everytime a Player starts the app and   
 starts the game.

Open Issues: How will more Characters be made available to users?

# PLAYER encounters an ENEMY

**Main Success Scenario**

During the course of the game, the player encounters an enemy that scrolls towards the player‘s chosen character. The player can execute either an attack or a jump to confront this enemy, or simply avoid the enemy. The player is expected to experience this use case multiple times during the course of one game. It is completely possible and maybe even probable that multiple enemies appear within a very short time frame. Upon dealing with this enemy, in whichever fashion is favored by the player, the game continues uninterrupted.

**Alternate Scenarios**

**Player fails to dispose or dodge the enemy.**

The player is unable to attack or avoid the enemy and the enemy has made contact with the player. The result is that the player loses hit points and in the event hit points are no longer available, the game ends and the player is taken to the gameover score screen.

**Enemies Fail to appear**

If a long period of time occurs in which the player does not encounter enemies, the player resumes playing the game dodging other obstacles rather than enemies. The player cannot lose hit points and will not lose the game unless the user fails to properly address a pitfall.

# PLAYER encounters a PITFALL

During gameplay, the player will occasionally encounter a pitfall scrolling towards him/her. The objective of the player will be to dodge this pitfall using any means necessary. Pitfalls will appear with varying size though it will be guaranteed that no pitfall is ever completely unavoidable. Thus, regardless of the character chosen, the game will always generate pitfalls of appropriate size.

Upon failing to doge a pitfall, the player will fall down and lose all hitpoints immediately ending the game. The player will then see the gameover score screen.

# Screen Specification

