

Culmination Presentation

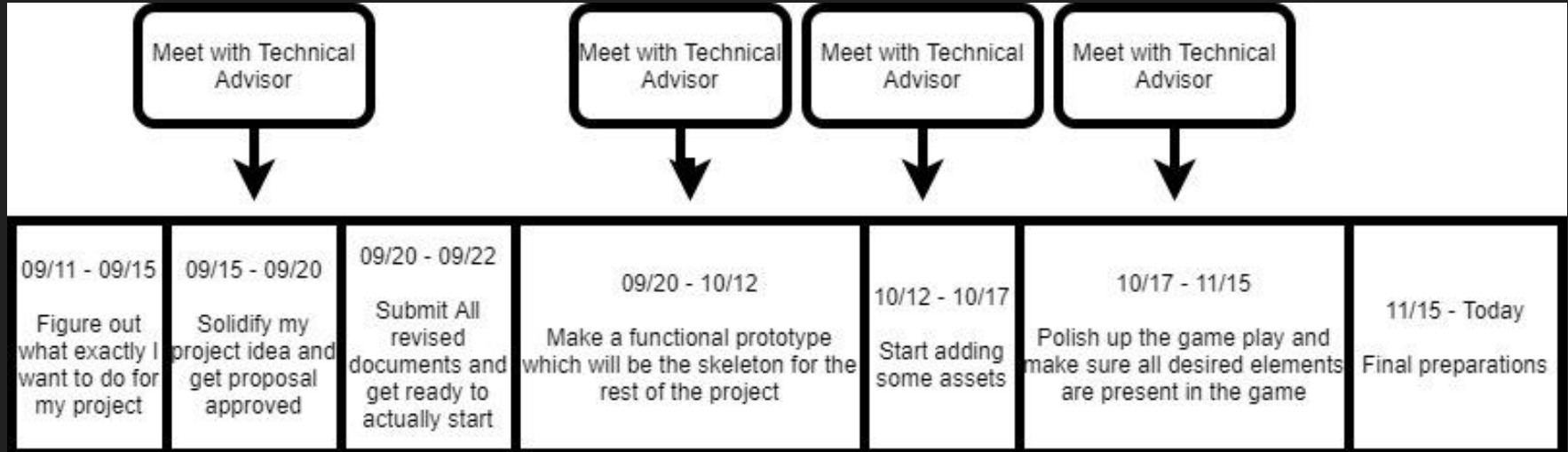
Featuring: “Asteroid Antagony”

By: Giancarlo Macias

Culmination Presentation

Featuring: “Asteroid Antagony”

Schedule/Workflow



What is “Asteroid Antagony”?

Asteroid Antagony is a video game that I have been constructing on Unity since the start of the semester as my culmination project.

What is “Asteroid Antagony”?

In this 2D game, you play as a little spaceship flying through space(ofcourse) while there are many kinds of Asteroids and other projectiles also flying through space(primarily AT you). The goal is to survive as long as possible by dodging anything that comes at you.

What is “Asteroid Antagony”?

Although I started working on this game this semester, the concept for this game has existed for many years. I actually tried to make this game a long time ago way before I knew anything about coding or Unity. This first iteration was called “Meteor Mayhem!” and was made using Android app inventor for use on a mobile android device.

“Meteor Mayhem!” looked like this:



“Meteor Mayhem!” looked like this:

The screenshot shows the MIT App Inventor web interface. On the left is a component palette with categories like Canvas, CheckBox, Clock, Image, Label, ListPicker, PasswordTextBox, Slider, TextBox, TinyDB, Media, Animation, Social, Sensors, Screen Arrangement, LEGO® MINDSTORMS®, Other stuff, and Not ready for prime time. The central canvas, titled 'Intro_Screen', shows a space-themed background with a large planet, a comet, and the title 'Meteor Mayhem!' in 3D orange text. A blue 'PLAY' button is at the bottom center. Below the canvas is a 'Non-visible components' section with a 'Player1' component. On the right is a properties panel for the selected 'Intro_Screen' component, showing settings for AlignVertical (Center), BackgroundColor (White), BackgroundImage (None...), CloseScreenAnimation (Default), Icon (None...), OpenScreenAnimation (Default), ScreenOrientation (Unspecified), Scrollable (unchecked), Title (Intro_Screen), VersionCode (1), and VersionName (1.0). Below the properties panel is a 'Media' section with a list of image files: Astriodforwork.png, Bluecometforwork.png, and Cometforwork.png. The bottom status bar shows system icons and the time 9:38 PM on 12/14/2013.

“Meteor Mayhem!” looked like this:

The screenshot shows the LEGO MINDSTORMS EV3 software interface. At the top, there is a search bar and navigation icons. The main workspace is divided into several sections:

- Component Palette (Left):** Lists various components like CheckBox, Clock, Image, Label, ListPicker, PasswordTextBox, Slider, TextBox, and TinyDB. It also has sections for Media, Animation, Social, Sensors, Screen Arrangement, and other categories.
- Play_Screen (Center):** A window showing a game scene with a ship, a planet, and asteroids. A score display at the bottom left shows "Score: 1".
- Non-visible components (Bottom):** A row of icons for Clock1 through Clock5, Player1, dieingsound, labelclock, and gobackclock.
- Component List (Right):** A list of components on the screen, including Ship, Asteroid2 through Asteroid5, planet, HorizontalArrangement1, Label2, Label1, Clock1 through Clock5, Player1, dieingsound, and labelclock. There are "Rename" and "Delete" buttons at the bottom.
- Properties Panel (Far Right):** Shows settings for the selected component, including AlignVertical (Top), BackgroundColor (White), BackgroundImage (None...), CloseScreenAnimation (Default), OpenScreenAnimation (Default), ScreenOrientation (Unspecified), Scrollable (unchecked), and Title (Play_Screen).

“Meteor Mayhem!” looked like this:

The screenshot displays the App Inventor for Android Blocks Editor interface for a project named "Meteor_Mayhem_Final - Play_Screen". The interface is divided into several sections:

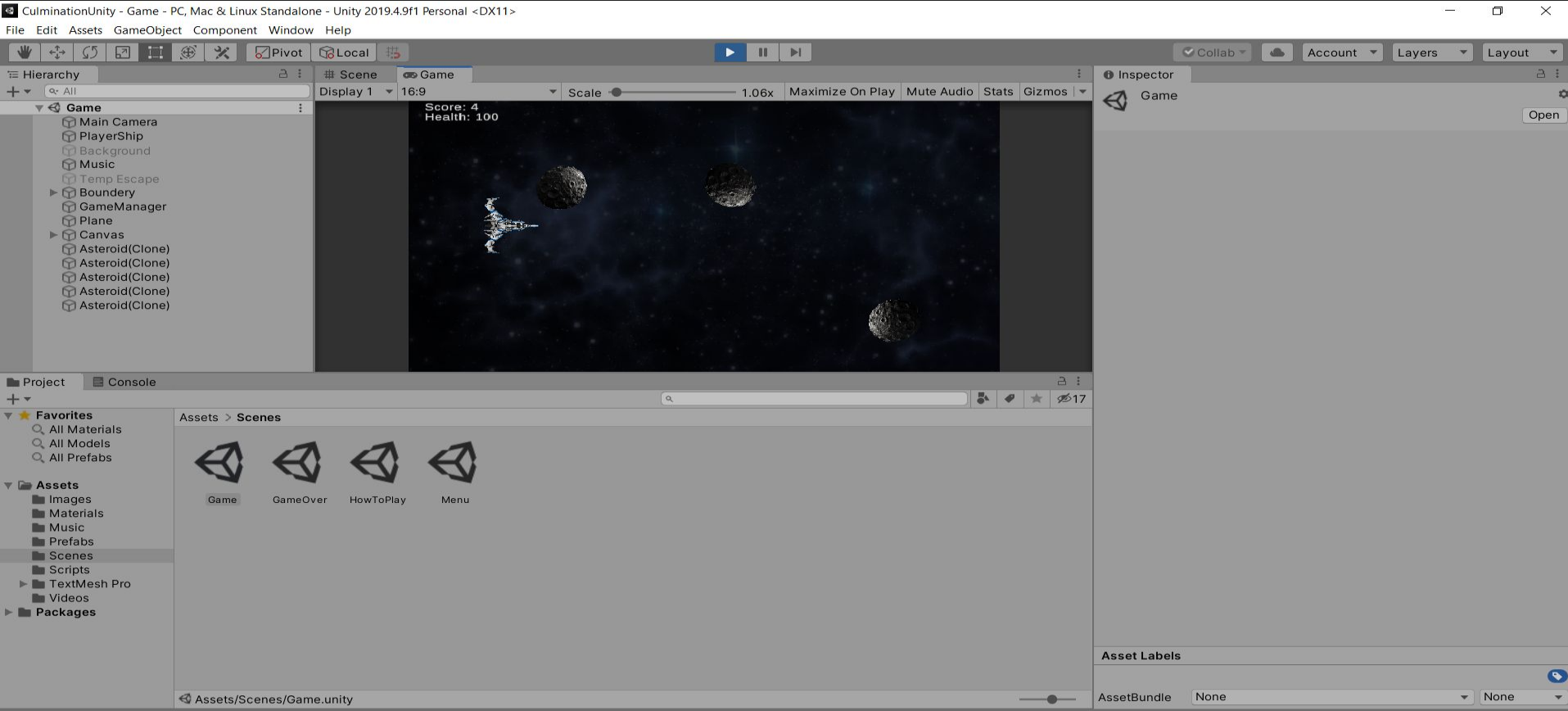
- Top Bar:** Includes "App Inventor for Android Blocks Editor: Meteor_Mayhem_Final - Play_Screen", "Saved", "Undo", "Redo", "New emulator", "Connect to Device...", and a "Zoom" slider set to 100%.
- Left Panel:** A category menu with "Built-In", "My Blocks", and "Advanced" tabs. Below are categories: "Definition", "Text", "Lists", "Math", "Logic", "Control", and "Colors".
- Main Canvas:** A complex arrangement of visual programming blocks:
 - Play_Screen.Initialize:** A sequence of "set" blocks for "planet.Visible" (to false), "Asteroid.Visible" (to false), "Asteroid2.Visible" (to false), "Asteroid3.Visible" (to false), "Asteroid4.Visible" (to false), and "Asteroid5.Visible" (to false), followed by a "call" block for "Player1.Start".
 - Canvas1.Dragged:** A "do" block containing "Ship.MoveTo" with parameters for X (currentX1) and Y (350).
 - Ship.CollidedWith:** A "do" block with multiple "set" blocks for "TextToSpeech1.Speak" (message: "its over dude"), "labelclock.Timer.Enabled" (to false), "gobackclock.Timer.Enabled" (to true), "Ship.Picture" (to explosion.jpg), "Ship.Height" (to 55), "Ship.Width" (to 55), and "Ship.Enabled" (to false). It also includes "call" blocks for "dieingsound.Play" and "Player1.Stop".
 - planet.EdgeReached:** A "do" block with "set planet.Visible" (to false).
 - planet_clock.Timer:** A "do" block with "set planet.X" (random integer from 0 to Canvas1.Width - 80), "set planet.Y" (to 0), "set planet.Visible" (to true), "call planet.PointTowards" (target: Ship), and "set planet.Speed" (to 10).
 - Asteroid2.Timer:** A "do" block with "set Asteroid2.Visible" (to true), "set Asteroid2.X" (random integer from 0 to Canvas1.Width), "set Asteroid2.Y" (to 0), "call Asteroid2.PointTowards" (target: Ship), and "set Asteroid2.Speed" (to 15).
 - Planet Initialization:** A "do" block with "set Asteroid.Visible" (to true), "set Asteroid.X" (random integer from 0 to Canvas1.Width), "set Asteroid.Y" (to 0), "call Asteroid.PointTowards" (target: Ship), and "set Asteroid.Speed" (to 15).
 - Other Elements:** A "when" block for "labelclock.Timer" with a "Label1.Text" block set to "Reports the boolean false" and a "number" block set to 1. A "when" block for "gobackclock.Timer" with a "call open another screen" block (screenName: "Screen1").



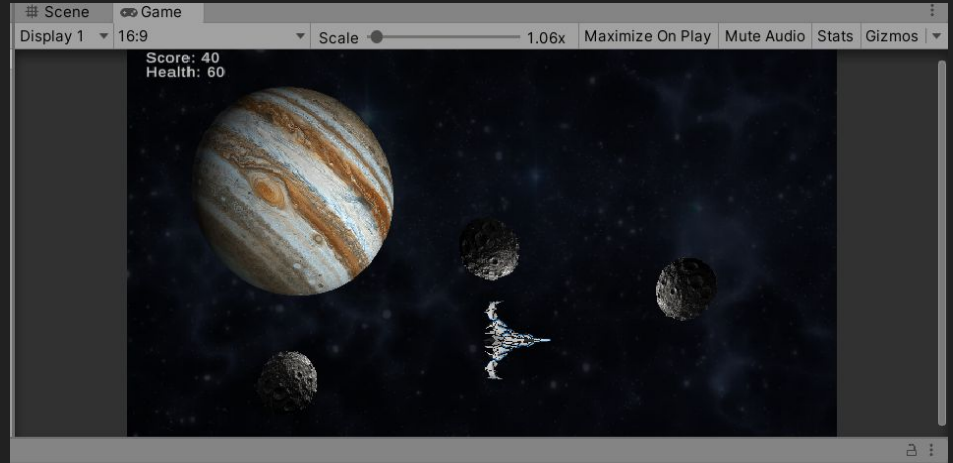
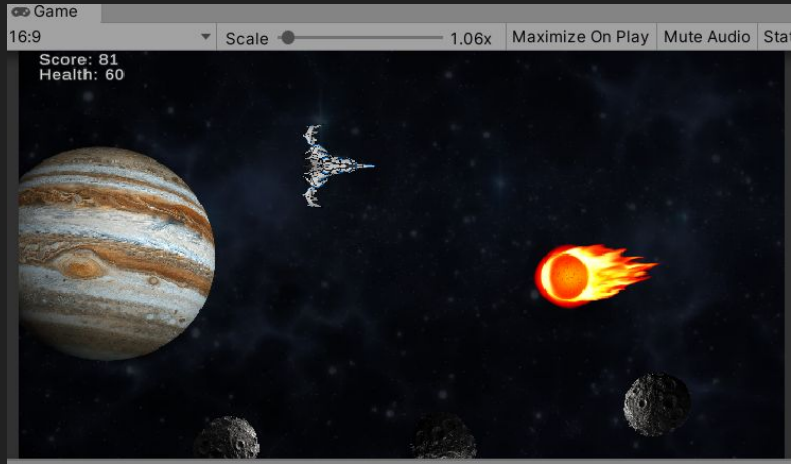
Out with the old
In with the New



This is what workin on my game looks like these days...



This is what workin on my game looks like these days...



This is what workin on my game looks like these days...

The image shows a dual-screen view of a Unity development environment. The left screen displays a game menu titled "ASTEROID ANTAGONY" with options "How To Play" and "START". The right screen shows the C# code for the GameManager script, which handles spawning asteroids and planets.

Unity Hierarchy (Left):

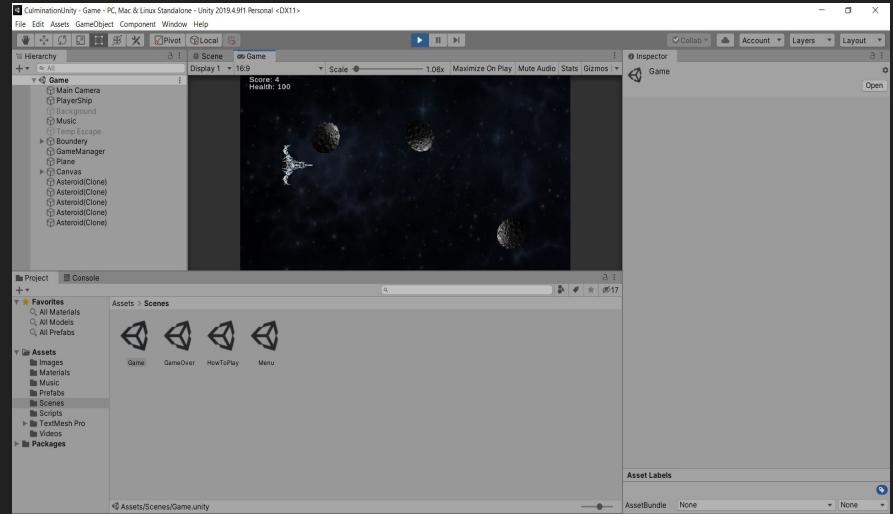
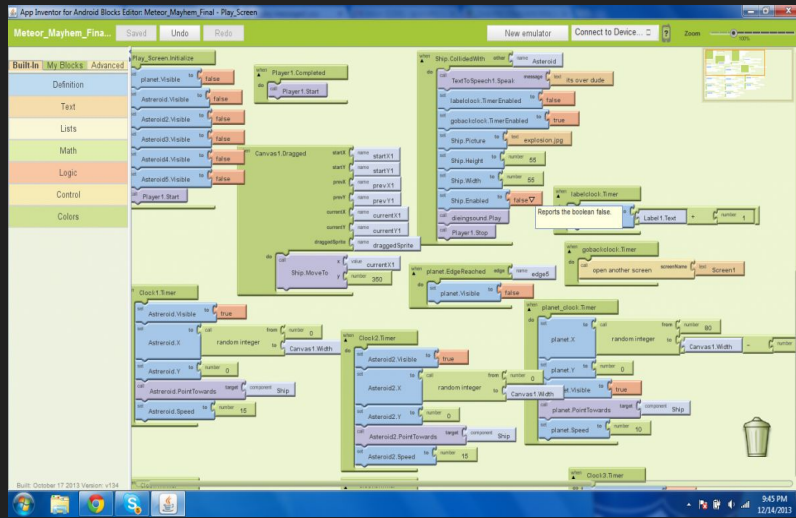
- Menu
 - Main Camera
 - HowTitle
 - Canvas
 - EventSystem
 - Music
 - Background

Assets > Scripts (Left):

- AestroidM...
- BackButton
- developer...
- GameMan...
- go2game
- go2how
- Leave
- Move
- planetmove

GameManager.cs Code (Right):`46 // Update is called once per frame
47 void Update()
48 {
49
50 healthText.text = "Health: "+health.ToString();
51 scoreText.text = "Score: " + score;
52
53
54 if (asteroidTimeTillSpawn <= 0)
55 {
56 Vector3 pos = new Vector3(xSpawn,Random.Range(ySpawnMin,ySpawnMax),0);
57 Instantiate(asteroidPrefab,pos,Quaternion.identity);
58 asteroidTimeTillSpawn = asteroidSpawnDelay;
59 }
60 else
61 {
62 asteroidTimeTillSpawn -= Time.deltaTime;
63 }
64 ////////////////
65
66 if (planetTimeTillSpawn <= 0)
67 {
68 Vector3 pos = new Vector3(xSpawn, Random.Range(ySpawnMin, ySpawnMax), 0);
69 Instantiate(planetPrefab, pos, Quaternion.identity);
70 planetTimeTillSpawn = planetSpawnDelay;
71 }
72 else
73 {
74 planetTimeTillSpawn -= Time.deltaTime;
75 }
76 ////////////////
77
78 if (commetTimeTillSpawn <= 0)
79 {
80 Vector3 pos = new Vector3(xSpawn, Random.Range(ySpawnMin, ySpawnMax), 0);
81 Instantiate(commetPrefab, pos, transform.rotation * Quaternion.Euler(0, 0,
82 commetTimeTillSpawn = commetSpawnDelay;`

Although the way things are done have changed, my overall goal hasn't changed.



My Goal:

My goal is to become a Lead Game Designer. I don't want to be in charge of telling people what to do without first being good at it myself. Because of this, I want to be versatile in the sense that I can help with any aspects of creating video games, whether that be, Level Design, UI/UX Design, Coding, or just being creative.

But first things first...

The thing about creating videogames is that it's not easy. People often become infatuated with the idea of being a Game Designer, and then quickly back down once they see that it's not just a walk in the park. I am currently in the stages of realizing how difficult it is to actually make a good game, but instead of choosing to give up, I am deciding to push forward and learn how to overcome any challenges that come up... especially the infamous coding part.

I want this project to serve as a testament to my dedication to Game Design.

My thoughts

I believe that I've come a long way, from knowing absolutely nothing, to now, being able to create my own video game.

Although I did indeed complete my game for the sake of this project, it's not Game Over. This is a good start, but I know I can do better. Even after this semester ends, I will continue to improve on my game and keep adding more to it and optimizing it until I can proudly publish it on a public platform to be played by the world.

Thank you Prof. Hosni

Without my technical advisor, my game wouldn't be even close to where it is today. The help and guidance is very much appreciated.

Asteroid Antagony Video Game

Giancarlo Macias

Emerging Media Game Design, NYCCT, Brooklyn New York 11201

Introduction

For this culmination project I will be embarking on the journey of creating my own video game from scratch via Unity. My game will be about the player controlling a spaceship trying to dodge asteroids. This game is inspired by an earlier version I made many years ago on Android App Inventor.

Process

This game will be created entirely on the Unity game engine platform. I will be using royalty free assets in my game to save time. My job will be to make the design choices I see best fit and then code them into reality.

Conclusion

My goal was to not only create my very own video game, but to also learn a lot while doing so. I definitely achieved that goal, but i'm also walking away with a future goal of continuing to improve and get better at my craft.

Meet with Technical Advisor

Meet with Technical Advisor

Meet with Technical Advisor

Meet with Technical Advisor

| | | | | | | |
|---|--|--|--|--------------------------|--|--------------------|
| 09/11 - 09/15 | 09/15 - 09/20 | 09/20 - 09/22 | 09/20 - 10/12 | 10/12 - 10/17 | 10/17 - 11/15 | 11/15 - Today |
| Figure out what exactly I want to do for my project | Solidify my project idea and get proposal approved | Submit All revised documents and get ready to actually start | Make a functional prototype which will be the skeleton for the rest of the project | Start adding some assets | Polish up the game play and make sure all desired elements are present in the game | Final preparations |





"That's all Folks!"

TM & © 1993 Warner Bros. - All Rights Reserved