Monster Smash GDD

Andrew Perrault – April 1, 2015

Table of Contents

1 Overview

1.1 Story

An asteroid filled with alien green goo breaks up and smashes to earth in thousands of little pieces.

One of the pieces flies into the mouth of our hero and he turns into a giant monster. The monster runs around the city smashing things because...monster.

1.2 Setting/Theme

The game is intended to be simple, playful and ridiculous. The game is intended to be played in short bursts in browser or on a phone. It should be balanced so each level takes between 45 seconds and two minutes. Most players will be dead in only a couple of minutes. If you play well then the whole game should only take approximately 10-15 minutes.

1.3 Player End-Goals

The object of the game is to cause as much destruction as possible. There is a high score board giving players a reason to play for even after beating the game.

1.4 Platforms

The game will be built in HTML5 and Javascript. The initial prototype will support desktop and mobile browsers. In the future packing the game into downloadable bundles for distribution through mobile storefronts will be possible.

2 Mechanics

It will be a 2D side scrolling action game.

The mechanics are simple. Each level consists of a section of a city. You must destroy a minimum percentage of the city to complete the level. You can climb and punch buildings to destroy them.

The key to keeping your monster powers is the green goo from the asteroid. Your goo meter is a combination of a timer and a HP bar. Green goo will slowly decrease over time. When you get shot or hurt your bar will decrease in chunks. If out run out of green goo then the game will end. You can find sources of green goo in the rubble of destroyed buildings. Finding sources of green goo will be required to complete the game.

There will be enemies mostly consisting of police and army men who will attempt to kill you.

3 Controls

You have a directional control that makes the monster run side to side. When you are near the edge of a building you can use the up/down axis to climb.

There is a jump button. Jumping is used primarily to avoid damage and move faster.

There will also be an attack button. Pressing the attack button without the directional control will attack directly in front of you. If you are hanging on a building then you will attack the building. Attacks will be affected by the directional control. You will be able to attack down at an angle or attack up at and angle or straight up. Jumping and attacking in the air is also possible.

There will be two sets of controls depending on whether you are on desktop or mobile.

Game Action	Desktop	Mobile
Directional Controls	WSAD	Virtual Touch Control Stick
Jump	Right Alt	Virtual Touch Button
Attack	Right Ctrl	Virtual Touch Button

4 Game Screens

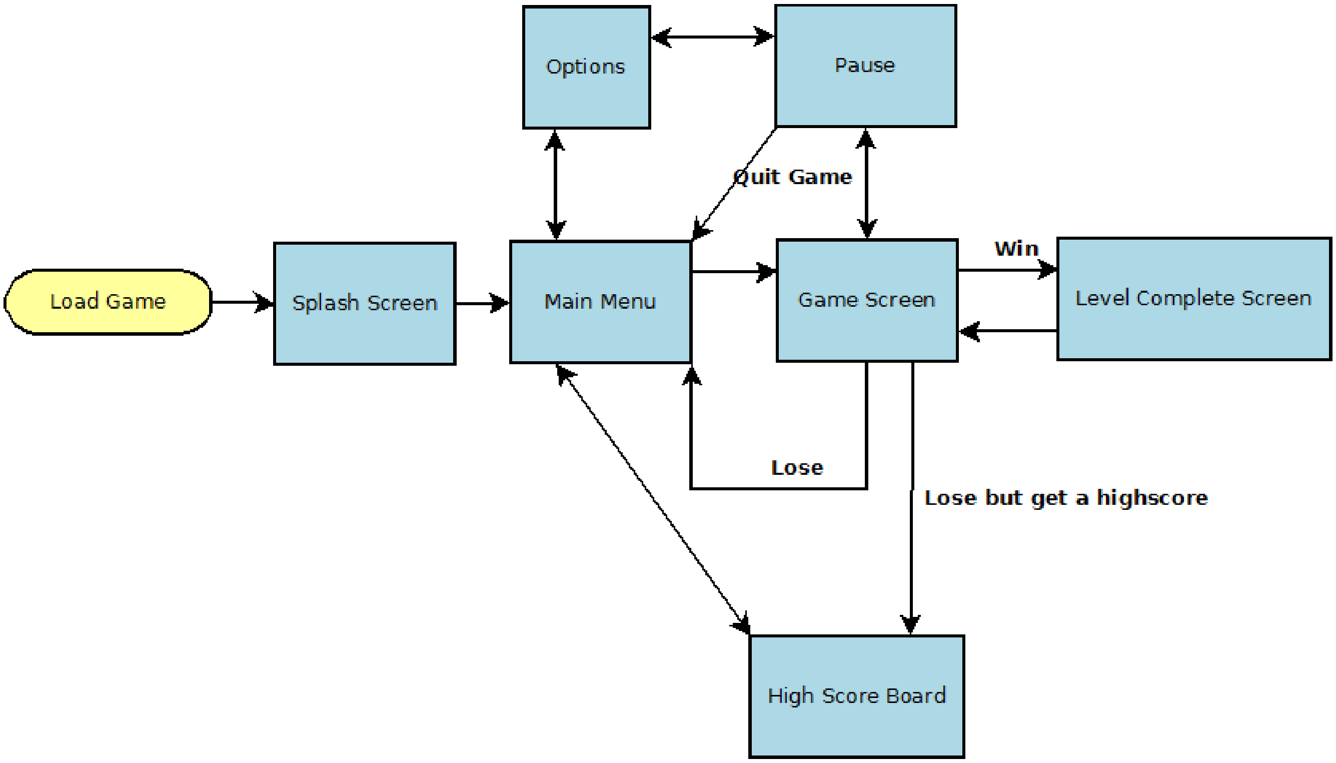

Illustration 1: High Level Game Flow

4.1 Game Screen

4.1.1 Gameplay Layer

This is the main layer where the player, building and enemies will be rendered.

4.1.2 UI

The in game UI will be as simple and small as possible because of the small screen space available. You will have a green goo meter at the top right and a score counter at the top left. On desktop the entire screen will be available for play. On mobile some space at the bottom left and bottom right will be take up by the virtual touch controls.

4.1.3 Background Layer

The background layer is a static image of the sky displayed behind the gameplay layer. The image will change depending on which level you are on but will not change during the level.

4.2 Splash Screen

The splash screen is the first screen the player sees. It has a simple picture and a progress bar to tell you that the game is loading.

4.3 Main Menu

The main menu has buttons to play the game, view the high score board or view options screen.

4.4 High Score Board

This is a record of the most number of points achieved by the player in previous runs through the game. It will have two modes. Read only mode is accessed from the main menu to view previous scores. If you achieve a high score then you will go to the high score screen with the option of entering your name and .

4.5 Options

The options screen can be accessed from the main menu or the in game pause overlay. The options screen has buttons to disable or change the volume of the games music or sound. The options screen will have a return button the will return to the screen that opened options originally.

4.6 Level Complete Screen

This screen opens every time the player completes a level. It will calculate bonus points based on remaining goo.

4.7 Pause Overlay

The pause overlay will pause the game and give access to the options screen.

5 Enemies

5.1 Police

Low damage and low rate of fire.

5.2 Soldiers

More damage and higher rate of fire than police.

5.3 Jeeps

Jeeps move quickly and have a mounted machine gun.

5.4 Helicopters

Attack helicopter with a machine gun.

6 Technical Design

6.1 Entity System

All enemies and the player will inherit from a common entity object. The entity object have hooks for the rendering system and a simple finite state machine.

6.2 Building System

The buildings in the game will be built using a tile system. The middle tiles can be repeated to allow the building to scale without using multiple tile files. Also the sections of the building can be individually damaged simply by offsetting the tile when the building is rendered.

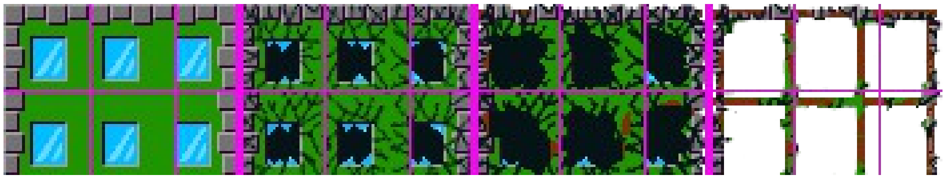

Illustration 2: Example Building Tile Sheet

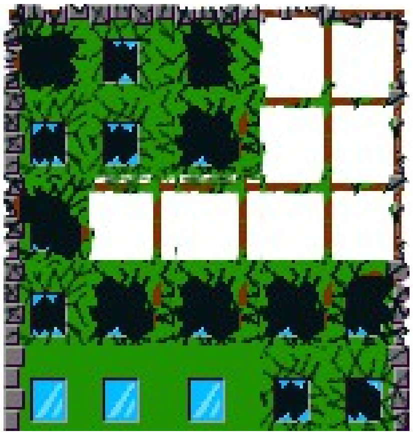

Illustration 3: Damaged building using tiles

6.3 Rendering System

Responsive controls are very important this game. I would like to do an experiment to see if I can offload rendering to a second worker thread to keep the main thread responsive even on slow devices. Web workers do not have access to the DOM specifically the canvas. It may be possible to create an artificial buffer context in a worker thread and send a typed array of data back to the main thread for rendering. I will need to setup and profile a test to see if the gain from doing most of the canvas calls in a worker thread outweigh the disadvantage of sending the messages back and forth.

7 Appendix A Options for the Future

This section is speculation for the future. Most likely none of these features will make it into the prototype but the are possible next steps if the prototype goes well.

7.1 Monetization

There are two possibilities for monetizing the game. Using either method the main game would be free to play.

First I could put ads on the loading screen and end of level screen.

The second possibility is in game purchases. Different monsters as well as different levels could be bundled into downloadable content pack. You could also let players purchase some kind of upgrade to health and damage using real money.

7.2 Online Leader Board

Tie the players individual local high score board to a global leader board. People will be more likely to keep playing if they see their high scores climbing a leaderboard compared to other players.

7.3 Multiplayer

This is a complicated addition but it would be fun to have multiple monsters playing at the same time.

7.4 Downloadable version iOS

Apple apps store version.

7.5 Downloadable version Android

Google play version.

7.6 Downloadable version Windows

Windows store version.

7.7 Controller Support

Controller support on the browser version using HTML5 gamepad API.