MICHIGAN STATE UNIVERSITY®
FEDERAL CREDIT UNION
Building Dreams Together

Michigan State University

Team MSUFCU

Money Smash Chronicle

Project Plan

Spring 2016

MSUFCU Staff:

Whitney Anderson-Harrell

Austin Drouare

Emily Fesler

Ben Maxim

Ian Oberg

Michigan State University Capstone Members:

Wyatt Hillman

Amy Leung

Cory Madaj

Brandon Max

Bobby Zhang

Table of Contents

1. Executive Summary

Michigan State University Federal Credit Union, known as MSUFCU, is the largest university-based credit union in the world in both membership and asset size. MSUFCU provides a variety of financial products and services including deposit accounts, personal and business loans, investments and insurance. MSUFCU is headquartered in East Lansing, Michigan. The credit union is owned and operated by its members located at Michigan State University, Oakland University, and their surrounding communities. As of October 2015, MSUFCU has fifteen branches, a membership of more than 205,000, and over $2.9 billion in assets.

As a university-based credit union, MSUFCU is committed to educating its customers about finances. To improve financial literacy among people of all ages, MSUFCU has commissioned an educational video game called Money Smash Chronicle. Money Smash Chronicle teaches players about finances while providing a simple yet fun and engaging gameplay experience.

The Money Smash Chronicle application is a “match-three” style puzzle game application similar to other popular puzzle games such as Candy Crush Saga. As the player progresses through the game, they answer financial quizzes and encounter gameplay obstacles based on real-world financial situations. This gives users of all ages and skill levels an experience that is both fun and educational. The application is available as both a browser-based web game and a mobile game compatible with both Android and iOS.

2. Functional Specification

The primary focus of the project is to create a game for customers to increase their financial literacy. Although MSUFCU currently has multiple mobile applications in addition to their website, the goal of their existing products is primarily to provide financial services typical of a credit union. MSUFCU is now adding a game called Money Smash Chronicle to their library of applications. The game is designed to both raise awareness of MSUFCU and increase the financial knowledge of its players.

The game is modeled after the existing puzzle game Candy Crush Saga. The main gameplay involves a grid where each tile has one of six game pieces. The user swaps two adjacent pieces in order to make chains of at least three identical pieces. In Money Smash Chronicle, these pieces are represented by different finance-related items. Each type of piece gives the player a different amount of money when the player makes a match with pieces of that type. Each level of the game has a different goal, such as reaching a given score in a certain number of moves or a certain amount of time.

The game is free-to-play and does not incorporate any in-app purchases. Financial quizzes are used in lieu of the microtransactions seen in other puzzle games. Players start with a number of lives, which are lost when levels are failed. When all lives have been lost, the player can wait for their lives to replenish, or they can answer quizzes. These quizzes increase financial knowledge by testing on various financial topics. The quizzes cover topics such as terminology and vocabulary, personal responsibility, the process of obtaining a loan, the rules for paying off loans and credit card debt, and more.

In addition to replacing microtransactions with finance quizzes, the game will also incorporate gameplay elements unique to Money Smash Chronicle that are designed to improve financial literacy. To teach the value of saving and financial responsibility, “life events” randomly occur. These events represent real-life expenses, such as car repairs or emergency medical expenses. When these occur, the money that was accumulated from matching pieces is used to pay the expenses. If the user has not saved enough money to pay these bills, they fail the level and lose a life. Additional gameplay obstacles include bills with impending due dates which must be cleared in addition to the level’s main score goal.

The game has a web version accessible from the browser of any modern computer and a mobile version compatible with Android phones and tablets and iOS devices including iPhone, iPad and iPod Touch.

3. Design Specification

3.1 Overview

The application will be available on any Android device running Android version 4.0.4 or higher, and any iOS device running iOS 8 or higher. The mobile versions can be launched by tapping the relevant icon on the device. After launching, the application goes straight to the main menu.

The web version can be accessed by visiting a website on a browser on any modern computer. As soon as the user enters the website, the main menu is launched.

3.2 Game Interface

To support users of all ages, all user interface elements are designed to be as simple and intuitive as possible. Menus have large, clearly-labeled buttons. The art direction is colorful and all gameplay features are clearly presented.

3.2.1 Landing Page

Figure 1.2 displays the home page of the mobile application, which is accessed as soon as the application is started. From here, a user can begin the game, which will take them to the level select page and eventually to the gameplay. Selecting settings will allow the player to change game volume and to check the how-to-play section.

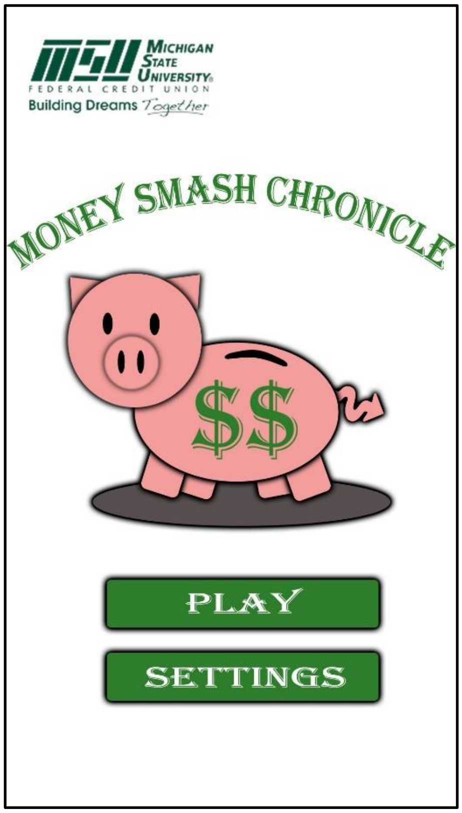


Figure 1.1 - Initial main screen for mobile devices

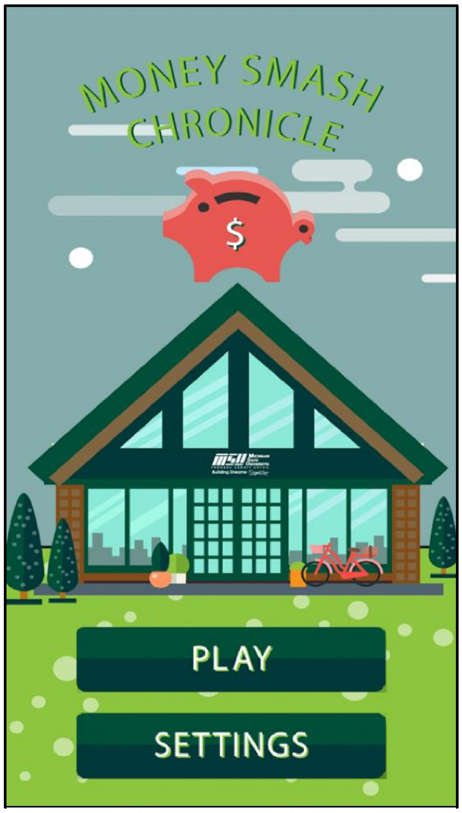


Figure 1.2 - Final main screen for mobile devices

3.2.2 Level Selection

A level selection interface allows the player to choose which level they want to play. The player can see which levels they can select, which levels have already been cleared, and which levels are currently inaccessible. More levels are unlocked as the player progresses through the game. When the user taps an icon for a level, a popup is displayed showing their best scores for the level, as well as the required goals to complete the level. The player will also be able to see how many lives they have remaining and the timer displaying the time until they receive an extra life. Arrow buttons will be provided to return to the main menu or scroll through their options for levels.

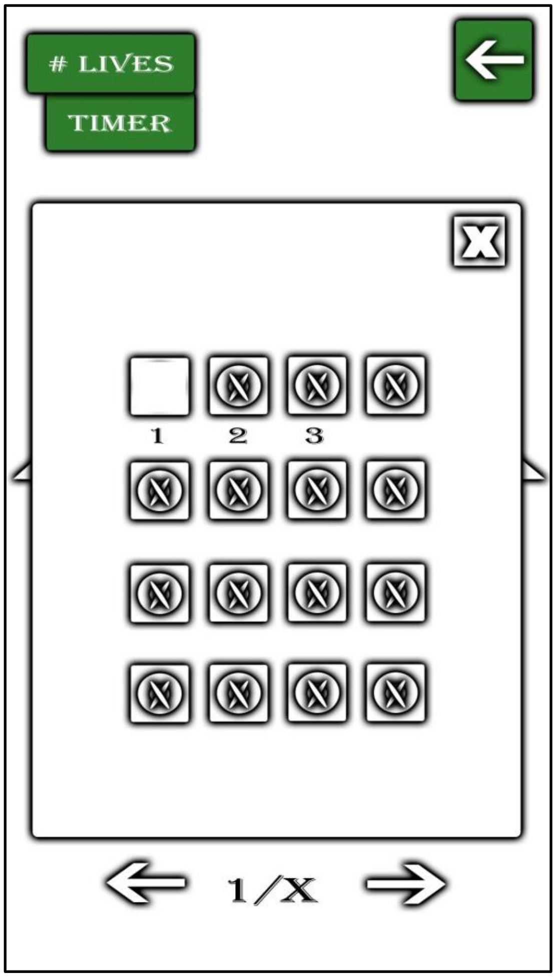


Figure 2 - Initial level selection screen

3.2.3 Game Play

On this interface, shown below in Figure 3, the user will play the Money Smash Chronicle game. The main feature is an 8x8 grid containing generated pieces. The different piece types include the safe, the piggy bank, the gem, the coin, the credit card and the dollar bill. Each piece is clearly visually distinguishable by having a different shape and color. Additional obstacles and special pieces will be determined by the level that the player selects. The player must swap two adjacent game tiles to line up three or more identical pieces. The matched pieces will disappear, rewarding the player with points and an amount of money determined by the type of piece that was matched. Certain special pieces may also occur, providing unique power-ups, additional time or moves, or cash bonuses. Each level will have a different goal, such as making a certain amount of money before running out of time or moves, or clearing a special piece. Additionally, life events modeled after real-life expenses such as medical bills may randomly occur. These events require the player to have a certain amount of money, or else they will fail the level. If they have enough money, the money is subtracted from their amount of money, but not their score.

Above the grid display, the player can view the number of remaining moves, score, amount of money and remaining game time. Below the screen, the player can select power-ups and view the life events that have taken place.

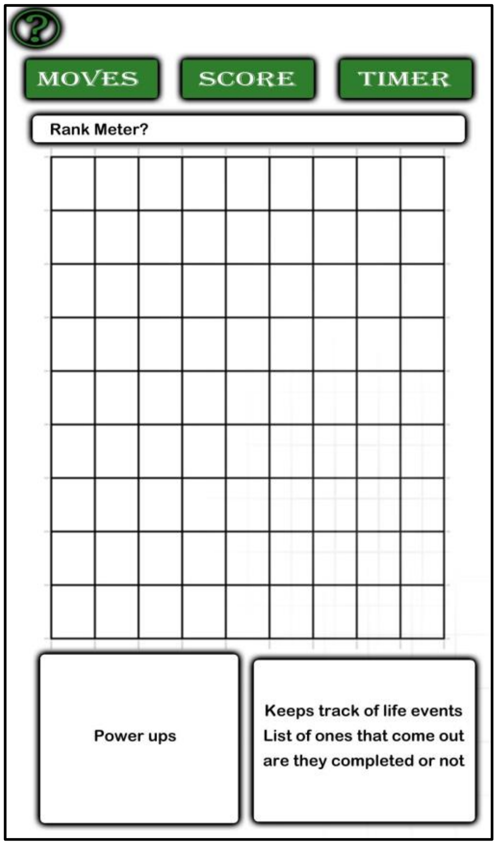

?
MOVES
SCORE
TIMER
Rank Meter?
Power ups
Keeps track of life events
List of ones that come out
are they completed or not

Figure 3.1 - Initial game screen

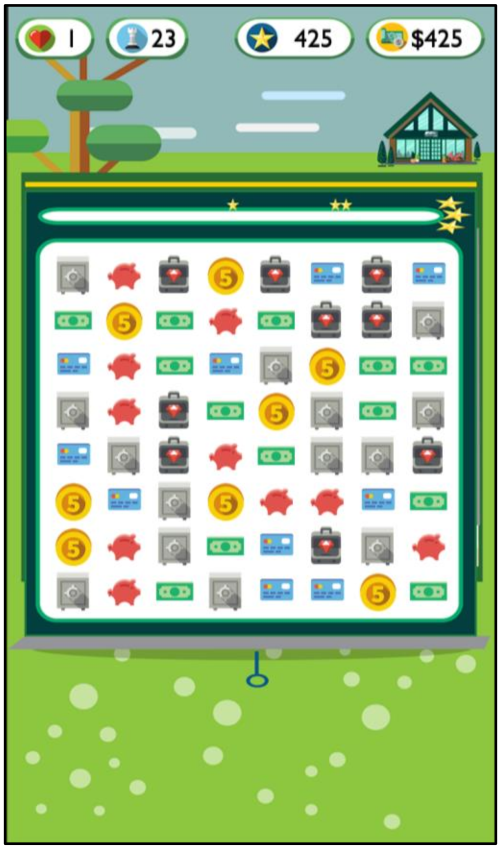


Figure 3.2 - Final game screen

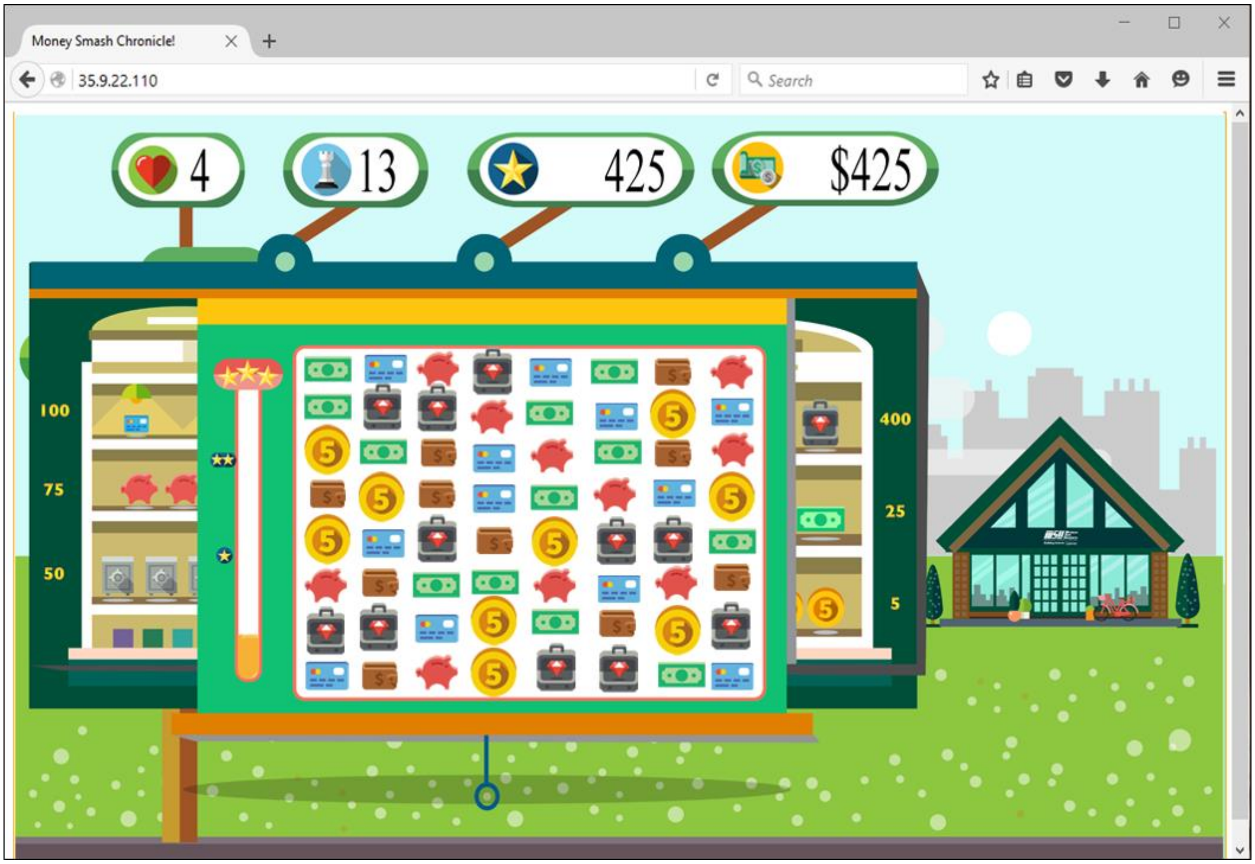
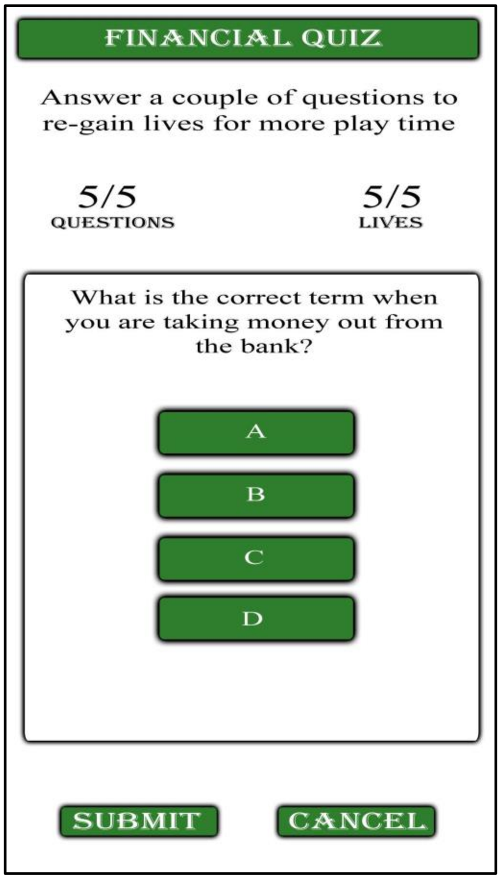


Figure 3.3 - Final game screen for web application

3.2.4 Financial Quiz

If the customer loses all lives, they can wait a certain period of time to get a life and continue, or they can answer a financial quiz to receive up to five lives. The quizzes, shown below in Figure 4, will be the primary educational aspect of the game. Once the customer chooses to take the quiz, they will have five financial multiple choice questions pop up. The questions involve topics such as finance vocabulary, financial responsibility and best practices, and the workings of services offered by a bank or credit union. The number of lives that they will get depends on how many questions they get correct in the quiz. For example, a player who answers three out of five questions will get three lives. A player who gets all questions wrong will not get any lives for that attempt.



FINANCIAL QUIZ

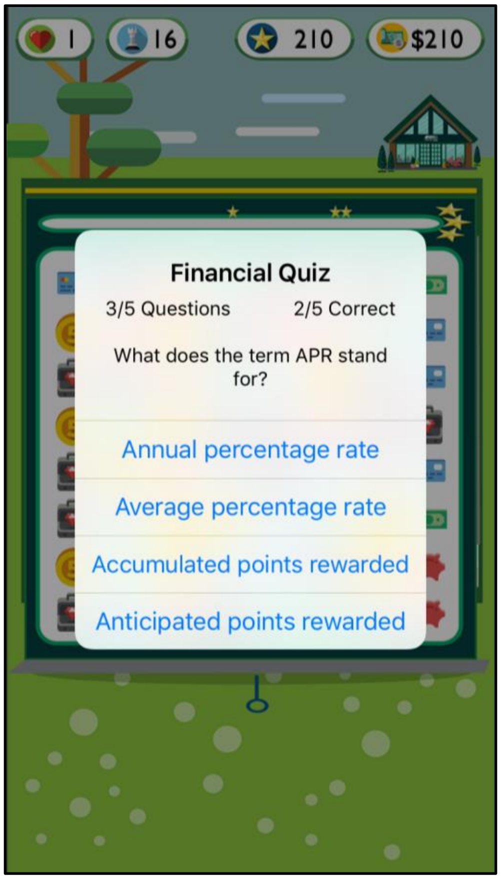
Answer a couple of questions to re-gain lives for more play time

5/5 QUESTIONS

5/5 LIVES

What is the correct term when you are taking money out from the bank?

Figure 4.1 - Initial quiz screen



Financial Quiz

3/5 Questions 2/5 Correct

What does the term APR stand for?

Annual percentage rate

Average percentage rate

Accumulated points rewarded

Anticipated points rewarded

Figure 4.2 - Final quiz screen

3.2.5 Game Over

If the customer loses all their lives, the game ends, and they are directed to a “Game Over” page. This page contains a “Continue” button and a “Quit” button, shown in Figure 5.1. By hitting the “Continue” button, it will take you to the page shown below in Figure 5.2, which contains more detailed statistics about the player’s performance in that game, as well as a “Continue” button which takes them back to the level select screen if they have more lives, or to a quiz if they have run out of lives. By hitting the “Quit” button, it will take you out of the game and return you to the main menu.

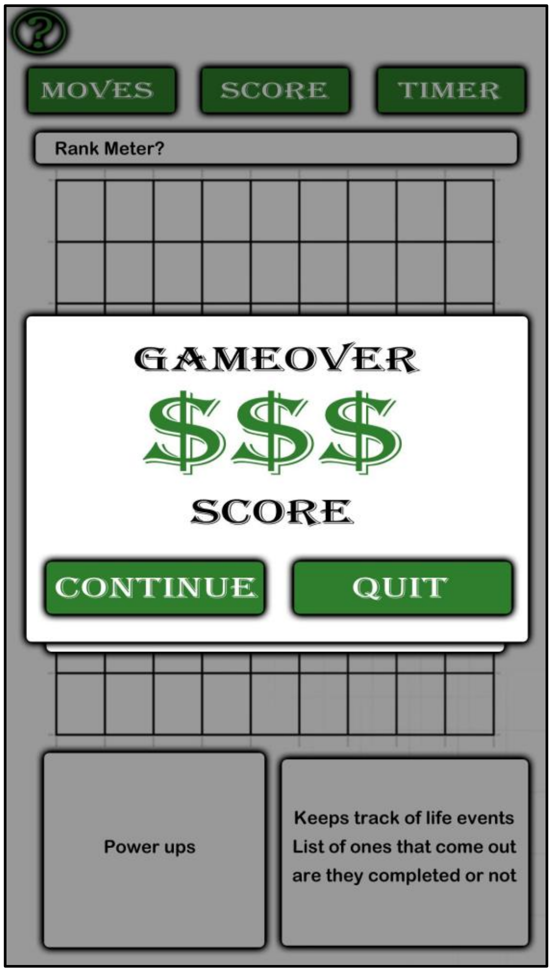

?
MOVES
SCORE
TIMER
Rank Meter?
GAMEOVER
$$$
SCORE
CONTINUE
QUIT
Power ups
Keeps track of life events
List of ones that come out
are they completed or not

Figure 5.1 - Initial game over popup

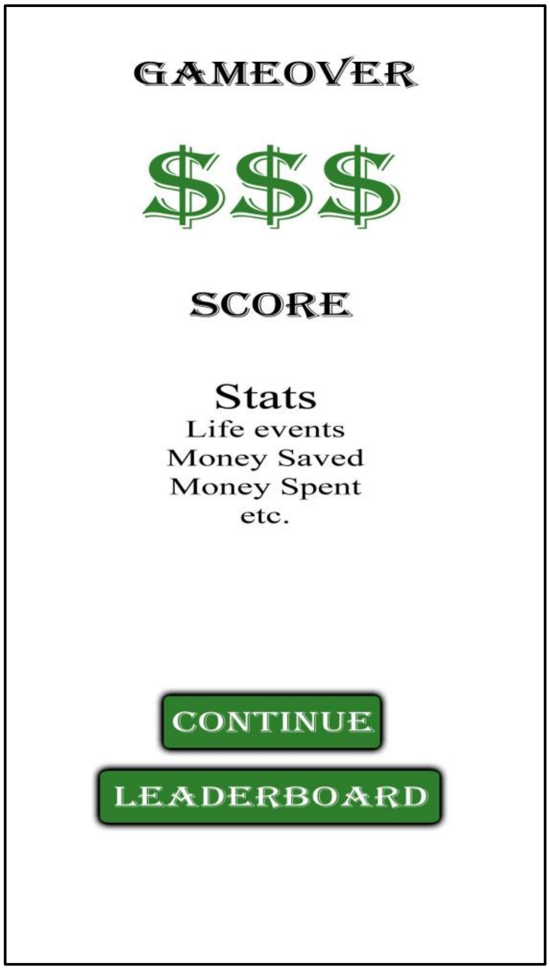
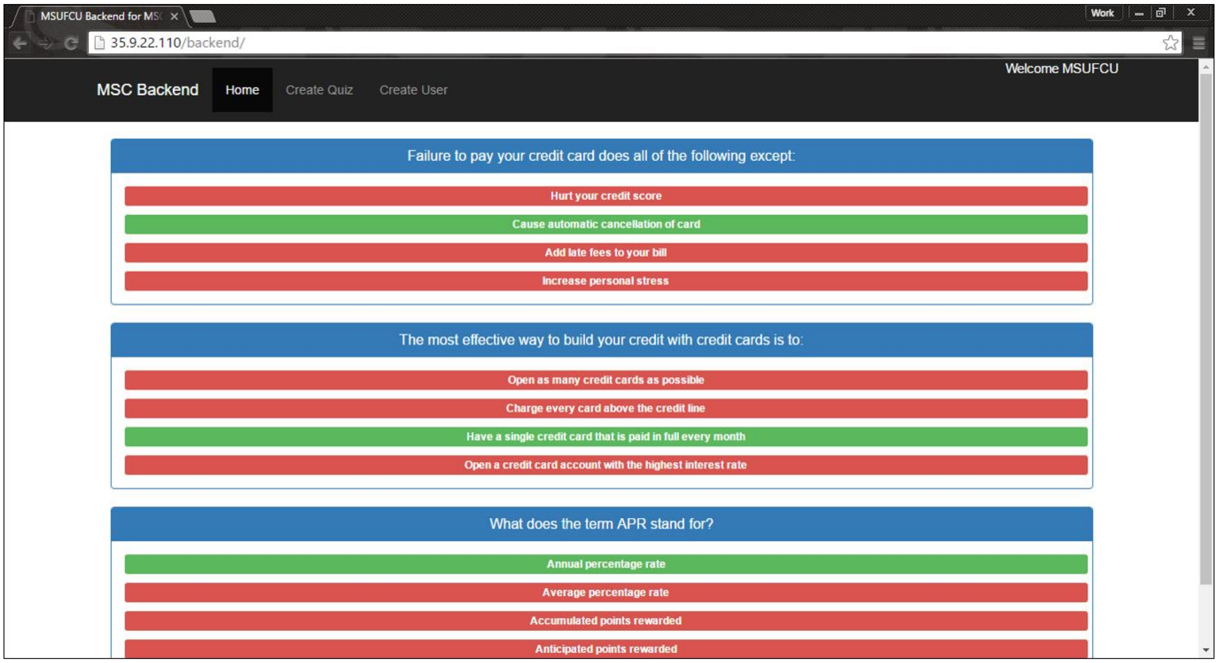


Figure 5.2 - Initial game over screen

3.3 Administrative Interface

The backend contains administrative features, shown below in Figure 6.1. An administrator can log in to add more quiz questions and add additional users.



Failure to pay your credit card does all of the following except:

Hurt your credit score

Cause automatic cancellation of card

Add late fees to your bill

Increase personal stress

The most effective way to build your credit with credit cards is to:

Open as many credit cards as possible

Charge every card above the credit line

Have a single credit card that is paid in full every month

Open a credit card account with the highest interest rate

What does the term APR stand for?

Annual percentage rate

Average percentage rate

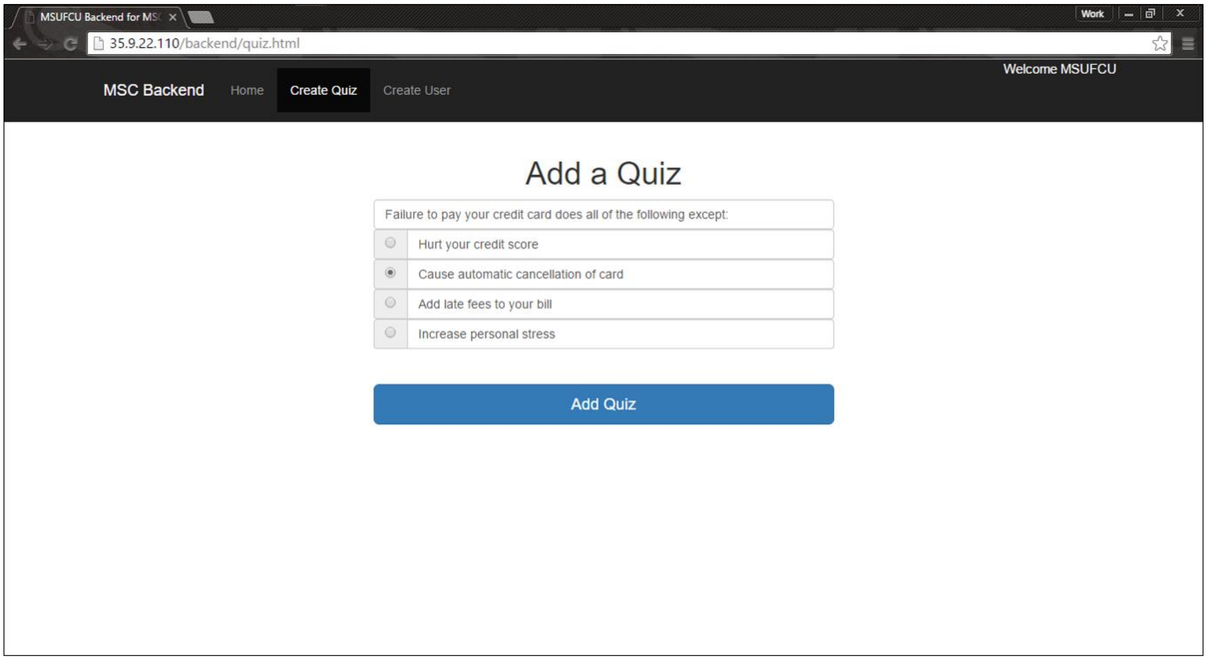
Accumulated points rewarded

Anticipated points rewarded

Figure 6.1 - Administrative backend website

3.3.1 Adding Questions

Shown below in Figure 6.2 is the interface for adding quizzes to the game. Since the game connects to the backend to retrieve quizzes, the added quizzes will be available to all users instantly with no need for a mobile user to update the application. The page contains “Create Quiz” and “Create User”. The “Create Quiz” button takes the user to a page that allows them to define a quiz question and the four possible choices. The “Create User” page allows the user to create more accounts usable by MSUFCU personnel to give them quiz adding privileges.



Add a Quiz

Failure to pay your credit card does all of the following except:

○ Hurt your credit score

● Cause automatic cancellation of card

○ Add late fees to your bill

○ Increase personal stress

Figure 6.2 - Administrative website to add quiz questions into Money Smash Chronicle

4. Technical Specification

4.1 System Architecture

Figure 7 provides a high level overview of the application's structure.

4.1.1 Server Side Architecture

The server side architecture is an LEMP stack; Linux, Nginx, MySQL, PHP. The API uses Laravel 5 to provide a restful API that allows data retrieval using the GET and POST HTTP methods. The API controls leaderboard scores, quiz questions, and quiz answers which will be stored in the MySQL Database. When accessing the API via GET, the server sends data in JSON format. For security and reliability purposes, the backend uses PDO and prepared statements to access the MySQL Database.

4.1.2 Android Architecture

The Android application is programmed in Android Studio using Java. The application requires a minimum Android version level of 4.0.4. To assist in development, the libGDX game framework is used. libGDX is an open-source game framework which simplifies the processes of screen-switching, game state management, sprite drawing and animation. A libGDX Game class contains the initialization functions as well as handling screen transitions.

The menu screens, gameplay screen, level select screen and quiz screens all have a corresponding class derived from the libGDX Screen class. Each screen contains a Stage object. A Stage is part of Scene2d, which is the libGDX scene graph. The Stage is populated by Actors. The buttons, HUD elements, and game pieces are all Actors. The Actor class contains many functions that allow for easy animation. The Stage handles the rendering of all actors during each loop of the main game loop.

The application connects to the server to retrieve quizzes. Level specifications, assets, and the user's progress are stored on the device. The level specifications are stored in JSON.

4.1.3 iOS Architecture

The iOS version is programmed in Swift using the Xcode development environment. This program requires iOS 8 or higher. The uses a Model-View-Controller structure. Animations and the game loop are handled by SpriteKit, which is Apple's game framework built into iOS. The application connects to the server to retrieve quizzes. Level specifications, assets, personal information needed for Apple's Game Center, and the user's progress are stored on the device. The level specifications are stored in JSON.

4.1.4 Front-Side Architecture

The front-side web architecture is written in Javascript and includes a game framework called Phaser IO, which allows easy creation of HTML5 Canvas apps. Phaser IO allows the ability to switch between Canvas and WebGL depending on the capabilities of the browser browser. If the browser does not support HTML5 Canvas, WebGL is used instead with the same Javascript code. The web app will use sessions for storing game progress and achievements. Assets will be stored on the server side.

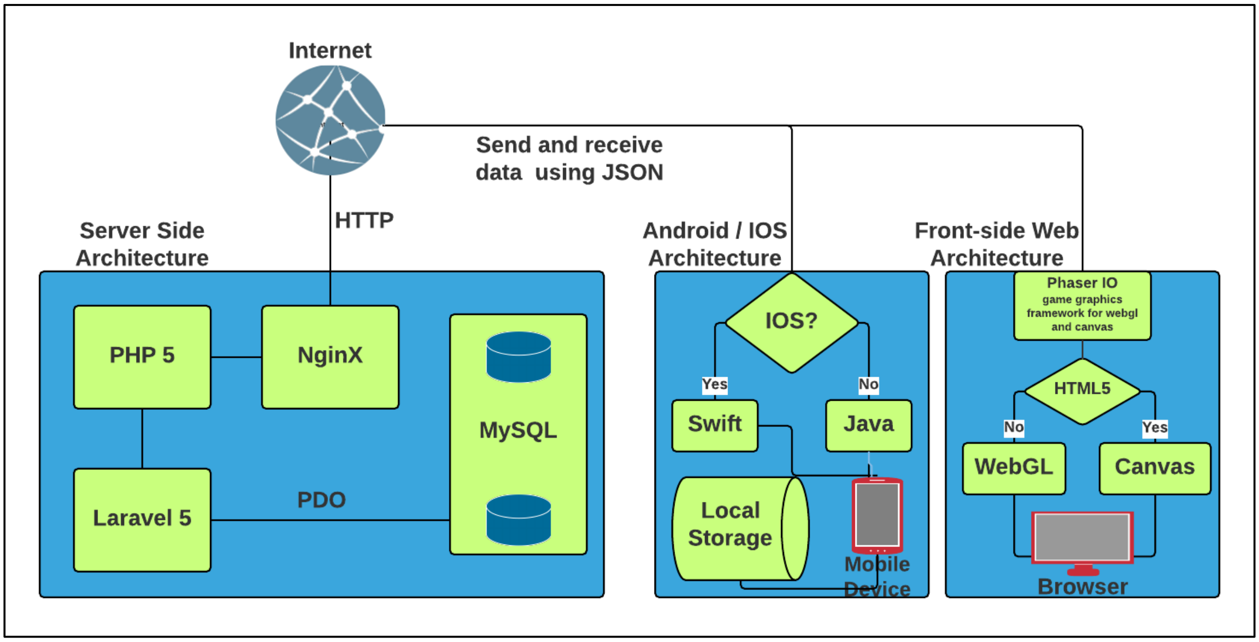

Internet
Send and receive data using JSON
HTTP
Server Side Architecture
Android / IOS Architecture
Front-side Web Architecture
PHP 5
NginX
MySQL
Laravel 5
PDO
IOS?
Yes
No
Swift
Java
Local Storage
Mobile Device
Phaser IO game graphics framework for webgl and canvas
HTML5
No
Yes
WebGL
Canvas
Browser

Figure 7 - Money Smash Chronicle System Architecture

5. Risks

- Gameplay Issues
    - Many gameplay aspects requested by client are incompatible with Candy Crush, or are otherwise difficult to implement
    - Mitigation: Need to make a lot of creative game design decisions under guidance of client
    - Status: Much more actual playtesting is needed to adjust point values and goal conditions. We have made many changes to the initial gameplay standards, but most work has been theoretical so far.
- No iOS or WebGL experience; minimal game framework experience
    - None of the team members have developed for iOS or with WebGL before, game engine experience is limited
    - Mitigation: Use tutorials, references and peers for help
    - Status: Mitigated. Team has reached high comfort levels in these technologies, and is now able to develop with efficiency and quality.
- Resource Issues
    - Resource-intensive project and no resources yet; no experienced designers
    - Mitigation: Work with client on design ideas
    - Status: Mitigated. Client has provided us with graphics for all game pieces and game screens. Team is using the provided graphics and Adobe Illustrator to create clean and visually appealing graphics.
- Cross-platform and cross-version issues
    - Potential difficulty in making the game look and feel similar across the three platforms and the different versions of each platform
    - Mitigation: Need to work together closely and test thoroughly and frequently
    - Status: Mostly mitigated. Android and iOS versions are very similar in appearance, besides some differences in pop-up window appearances and animation. Game feels similar on all platforms in terms of features like animation speed, though some fine-tuning is still needed. Android version looks the same for devices of all screen sizes and resolutions.

6. Timeline

Week 1 (1/11 – 1/15)

- Met with team, established weekly meeting times with course, clients and within team
- Downloaded basic OS and some software
- Initial client conference call

Week 2 (1/18 – 1/22)

- Downloaded GitHub, Android Studio, Xcode
- Began project plan
- First drafts of layouts for application and website
- Made test application and basic website

Week 3 (1/25 – 1/29)

- 1/25 – Status Report Presentation
- Continue working on Project Plan Proposal/Presentation
- Begin game skeleton code

Week 4 (2/1 – 2/5)

- Skeleton code completed for all platforms
- Begin implementing the core game features: matching/creation/etc...
- Get client confirmation on mockups and their designs

Week 5 (2/8 – 2/12)

- 2/10 - Project Plan Presentation
- Continue with implementation of the core game features
- Menu screen and game screen w/ matching are completed
- Look into getting application to scale for different resolutions

Week 6 (2/15 – 2/19)

- Core game features completed for all platforms
- Begin work on financial aspects of games: power-ups/quiz on death/scoring

Week 7 (2/22 – 2/26)

- Continue work on quizzes
- Begin life events

Week 8 (2/29 – 3/4)

- Alpha Presentation
- Finish financial aspects of games
- Begin work on back-end systems: user storage/highscore API/game centers for iOS & Android/question storages

Week (3/7 – 3/11)
Spring Break

Week 9 (3/14 – 3/18)
Finish work on back-end systems
Begin work on levels systems/design & look finalization
Port Android version to LibGDX framework

Week 10 (3/21 – 3/25)
Start project video
Finish porting Android version

Week 11 (3/28 – 4/1)
Finish work on levels system/design & look finalization
Finish obstacles and life events

Week 12 (4/4 – 4/8)
?? – Beta Presentation
Improve aesthetics
Playtesting and value adjustment

Week 13 (4/11-4/15)
?? – Beta Presentation
In person client meeting with MSUFCU
Improve aesthetics
Playtesting and value adjustment

Week 14 (4/18-4/22)
Continued debugging, additional final modifications and wrap up

Week 15 (4/25-4/29)
4/25 – Project Videos
4/27 – All Deliverables Due
4/28 – Design Day Setup
4/29 – Design Day