

Ripoff Card Game

Azkaban Dabmentors

Last Updated 12-09-2019

Team Members: Tyler Wallschleger, Brandon Kyle, Dakota Simpkins

Stakeholders: CSC 340 Class

I HAVE ABIDED BY THE UNCG *Academic Integrity Policy* ON THIS ASSIGNMENT.

1.2 Table of Contents

1. Introduction

1.1. Title: Page 1

1.1.1. Project Name: Page 1

1.1.2. Team Name: Page 1

1.1.3. Date: Page 1

1.1.4. Team Members: Page 1

1.1.5. Stakeholders/Company – Honor Code: Page 1

1.2. Table of Contents: Page 2-3

1.3. Purpose: Page 4

1.4. Document Conventions: Page 4

1.5. Intended Audience: Page 4

1.6. Definitions/Jargon: Page 4

1.7. Project Scope: Page 4

1.8. Technical Challenges: Page 4

1.9. References: Page 4

2. Overall Description

2.1. Product Features: Page 4

2.2. User Characteristics: Page 4

2.3. Operating Environment: Page 4

2.4. Design and Implementation Constraints: Page 4

2.5. Assumptions and Dependencies: Page 4

3. Functional Requirements

3.1. Primary: Page 4

3.2. Secondary: Page 4

4. Technical Requirements

4.1. Operating Systems/Compatibility: Page 4

4.2. Interface Requirements: Page 4

4.2.1. User Interface:	Page 4
4.2.2. Hardware Interface:	Page 4
4.2.3. Software Interface:	Page 4
4.2.4. Communications Interface:	Page 5
5. Nonfunctional Requirements	
5.1. Performance Requirements:	Page 5
5.2. Safety/Recovery Requirements:	Page 5
5.3. Security Requirements:	Page 5
5.4. Policy Requirements:	Page 5
5.5. Software Quality Attributes:	Page 5
5.5.1. Availability:	Page 5
5.5.2. Correctness:	Page 5
5.5.3. Maintainability:	Page 5
5.5.4. Reusability:	Page 5
5.5.5. Portability:	Page 5
5.6. Process Requirements:	Page 5
5.6.1. Development Process Used:	Page 5
5.6.2. Time Constraints:	Page 5
5.6.3. Cost and Delivery Date:	Page 5

1.3 Statement of Purpose: We are designing a card game called Ripoff

1.4 Document Conventions

1.5 Intended Audience: The intended audience for this project is the CSC340 class.

1.6 Jargon/Definitions: “Card” and “Deck” most commonly refer to the data structures used to represent a card and a deck rather than a physical card or a deck of cards.

1.7 Project Scope: This project will receive no further updates past 11/09/2019, the due date of the project.

1.8 Technical Challenges: Only one of the people involved with this project has any experience in APIs.

1.9 References/Citations

2.0 Overall Description

2.1 Product Features: Our card game will have matches over the internet and allow users to buy card packs with real world money.

2.2 User Characteristics/classes: Our users will be people who want to play a card game. Some users will not spend any money on packs, others will spend some money, and others will spend a lot of money. There might be users who are inexperienced with card games and other users who are experienced.

2.3 Operating Environment: We can only guarantee that this program will work on Windows 10.

2.4 Design Constraints: We must finish the application in a limited window, which may mean that some features will have to be cut.

2.5 Assumptions/Dependencies: We assume that Windows 10 will still be a usable operating system by the due date and that Java will still be supported by the due date.

3.0 Functional Requirements

3.1 Primary: We need a login, a way to access the store, a way to spend money, a way to open packs, and ability to play the game itself.

3.2 Secondary: Users could create their own tournaments which would boost card pack sales.

4.0 Technical Requirements

4.1: Operating Systems/compatibility: We can only guarantee that it will run on Windows 10.

4.2 Interface Requirements: User must own a computer with Windows 10 installed, a monitor, and a keyboard and mouse to use our application.

4.2.1 User Interface: Keyboard and mouse will be used for the user to interact with the application.

4.2.2 Hardware Interface: The project will have to communicate with a server for transactions and playing matches over the internet.

4.2.3 Software Interface: The project will have to interface with a transaction software for users to spend money for card packs.

4.2.4 Communications interface: We will use the Stripe API, which will be used for ecommerce transactions within the game.

5.0 Nonfunctional Requirements: The project will require the ability to operate a mouse and keyboard and a computer.

5.1 Performance Requirements: The project should have no issue performing when making transactions. The game matches themselves should also have no issue.

5.2 Safety and Recovery: If an error occurs, the application will close. When the user restores the application, their most recently saved data should still be available.

5.3 Security Requirements: We will need security for transactions so that there is no possibility of someone acquiring cards without payment or payment information being stolen.

5.4 Policy Requirements (optional)

5.5.1 Availability: The application will not be sold. It will be displayed in the CSC340 classroom and available for download on Github.

5.5.2 Correctness: It is not essential that the application is completely accurate with the game logic, but it is essential that parts of the application relating to transactions or monetary exchange is handled in the most accurate way possible.

5.5.3 Maintainability: There will be no updates after the due date of 12/09/2019.

5.5.4 Reusability: The application can facilitate transactions between people with the ecommerce system. Users of the application could also use it to run tournaments.

5.5.5 Portability: The application should be able to be used on all computers that support the most recent version of Java and have keyboard and mouse input but there are no guarantees.

5.6.0 Process Requirements: All team members must abide by our methodology and come to meetings where we will discuss the development of the project.

5.6.1 Methodology: We will design our application using MVC architecture.

5.6.2 Time constraints: The project must be completed by 12/09/2019.

5.6.3 Cost and Delivery Date: The project will launch on 12/09/2019 and will cost nothing but our time.