Product Backlog
Team 12

Austin Dewey, Joseph Nathan, Joseph Landis, Steven Motherwell, Stephen Kramer

1.   Problem Statement

There are currently no bug-free marketed applications which deliver the complete casino gaming experience.

2.   Background Information

Many people are always excited to partake in casino games in the hopes of winning it big! There is also a market for the more casual player who just wants to enjoy the experience of lounging around a casino and playing a few casino games. Our application targets both players by giving the first a chance to practice strategy while giving the latter a chance to maybe learn new games and enjoy the games they know and love. We are setting out to bring the entire casino experience in a no-risk, helpful way so that all people can experience and learn some of the popular games that are played in casinos today.

3.   Environment

The application will be installed and run on a computer, the program will be developed using C# and available C# add-ons and forms. If time allows we will port the application to be made available on web browsers such as Google Chrome, Mozilla FireFox, and Internet Explorer.

4.   Functional Requirements

Backlog ID	Functional Requirement	Hours
1	As a user, I would like to be able to save my game progress. (credit winnings/losses)	7
2	As a user, I would like to be able to use my credits that I've earned in other casino games.	4
3	As a user, I would like to be able to bet my credits in specific increments.	4
4	As a user, I would like to cash in my credits for prizes.	8
5	As a user, I would like a UI which displays my cards and table in an understandable way.	12
6	As a user, I would like the option to view in-depth instructions for each casino game.	2
7	As a user, I would like all odds to be randomly calculated by considering the realistic odds.	4

8	As a user, I would like an easy method for leaving a game and joining a new game.	7
9	As a user, I would like to be able to view my prizes and user information in a specific section of the application.	5
10	As a user, I would like to be able to easily restart with a set amount of credits.	3
11	As a user, I would like to be able to play against or with other computer players. (if time allows)	-
12	As a user, I would like an inviting main menu where I can select which game I would like to play	6
13	As a user, I would like to be able to play multiple types of poker including Jack's High, 5 Card Draw, and (if time allows) Texas Hold'Em	25
14	As a user, I would like to play table games such as Craps or Roulette. (if time allows)	-
15	As a developer, I would like a player object which is cross compatible with all casino games.	5
16	As a developer, I would like a dealer object which can be used in all card based casino games.	10
17	As a developer, I would like to store user information locally without using a database.	10
18	As a user, I would like to be able to continue playing where I left off.	2
19	As a developer, I would like to make the application playable via web browser, (if time allows)	-
	Total:	**114**

5. Non-Functional Requirements

1. Our application will present a clean UI.
2. We must ensure that the program is structured to mitigate possible latency.
3. The cards in card based games must be easily distinguishable.
4. In all games specific counts need to be tracked to ensure no doubles are being dealt or used inappropriately.
5. The application must maintain its usability in all casino games.
6. Images need to be created to depict the prizes and if possible aspects of the UI.
7. The application must maintain the same theme in all stages of application transition.

6. **Use Cases**

Case: Enter a game of Poker   **System Responses**
1. Select type of poker   2. Application loads requested poker game
                                                          3. User is prompted to place bet
4. User sets and confirms bet   5. Application deals hand to user

Case: Leave a game of Poker   **System Responses**
1. Select "Leave Game" option   2. Application loads dialog to confirm user action
3. User confirms "Leave Game" action 4. Application updates user's credits
                                                          5. Application loads main menu UI

Case: Fold current poker hand   **System Responses**
1. User opts to fold current hand of cards 2. Application updates all other players to action
                                                          3. Application updates betting pot
4. User waits until current hand ends   5. Application loads new hand for all users

Case: User wins a poker hand   **System Responses**
1. User wins current poker hand   2. Application updates all players with this info
                                                          3. User's winnings total is updated
                                                          4. Application deals new hand to all players

Case: User loses all credits in poker hand **System Responses**
1. User loses an "all-in" betting hand   2. Application updates player's purses
                                                          3. User is prompted to leave game
                                                          4. Application loads main menu UI

5. User waits until purse is restored

Case: User raises a bet   **System Responses**
1. User opts to raise a bet during poker hand
2. User sets and confirms bet   3. Application updates other players to action
                                                          4. Application updates user's purse
                                                          5. Application prompts players to call or fold

Case: User saves the game   **System Responses**
1. User selects "Save Game"   2. Save dialog box pops up
3. User selects save location and clicks "save" 4. Credits and player information are saved
                                                                   to a file

Case: User changes to a different game **System Responses**
1. User selects "Change Game"   2. Change game dialog box pops up

3. User selects desired game	4. UI is updated accordingly

Case: User gets "help" information   **System Responses**

1. User selects "Help"	2. The help form corresponding to the current game type is displayed
3. User selects "OK"	4. The help form closes