Project Requirements for CS172 Final Project

Group 5

Steven Mumford & Eric Klicker

Part 1:

Our project will be a user-operated game that allows the user to select the game of their choice. Their choices will be either Blackjack or Texas Hold’em (poker). When the user begins they will be prompted with a menu that asks for their name. From there they will be able to select which game they wish to play. While playing BlackJack the computer will keep track of wins, losses, and winning percentage based off of the user that plays. If a different user decides to play, the computer will keep track of their individual stats as well. Poker will do the same thing and keep track of stats for the individual player. In poker, we will implement CPU players that preform very basic betting to mimic other players in the game. For both games, if the user wants to quit they can exit to the menu at any given time.

|  |
| --- |
| Game |
| String name;  Games: vector<string> |
| +menu(user: User\*): void  +intro(): User\* |

|  |
| --- |
| Blackjack (Derived from game) |
| -name: string |
| +play(User): void virtual  cardOutput(int): void virtual  hit(int&): int virtual |

|  |
| --- |
| Texas HoldEm (Derived from game) |
| -name: string |
| +play(user: User\*): void virtual  +tiebreaker(user1: User, user2: User): User virtual |

|  |
| --- |
| Card |
| -suit: string  -value: string |
| +Card(suit: string, value:string)  +getSuit(): string  +getSuit(): string  +displayCard(): void  +isFace(): bool  +getValueAsInt(): int  +getSuitAsInt(): int  +operator==(Card&): bool  +operator>(Card&): bool  +operator<(Card&): bool  +operator!=(Card&): bool |

|  |
| --- |
| Deck (Has a card) |
| -vector<card\*> Deck |
| +Deck()  +~Deck();  +at(index: int): Card\*  +shuffleDeck(): void  +clearDeck(): void |

|  |
| --- |
| User |
| -name: string  -money: int  -blackJackNumberOfWins: int  -blackJackNumberOfGames: int  -texasHoldEmNumberOfWins: int  -texasHoldEmNumberOfGames: int  -folded: bool  -vector<Card\*> hand |
| +User()  +User(name: string)  +getName(): string  +getMoney(): int  +getBlackJackNumberOfWins(): int  +getBlackJackNumberOfGames(): int  +getTexasHoldEmNumberOfWins(): int  +getTexasHoldEmNumberOfGames(): int  +getFolded(): int  +setMoney(MONEY: int): void  +addToMoney(winnings: int): void  +setFolded(bool): void  +incrementBlackJackNumberOfWins(): void  +incrementBlackJackNumberOfGames(): void  +incrementTexasHoldEmNumberOfWins(): void  +incrementTexasHoldEmNumberOfGames(): void  +saveStats(): void  +showStats(): void  +sortHand(); void  +handAt(index: int): Card\*  +showHand(): void  +addToHand(card1: Card\*): void  +highCard(): Card\*  +getHighValues(): int  +getHighStraight(): int  +understandHand(): int  +clearHand(): void  +blackJackbet(): int  +texasBet(): int  +fold(): void |

|  |
| --- |
| CPU (Derived from user) |
| -name: string -money: int |
| +CPU()  +CPU(int)  +blackJackBet();  +texasBet(rank: int, otherBet: int, check:bool) |