

REPO URL: <https://github.com/wailingwalrus/CSE360-Dice.git>

Group Members:

Peter Kharlakian

Kyle Zinke

Josh Gold

Nick Henson

Sudeep Aravind

This game is, in essence, modified blackjack with dice instead of cards

The Game and its rules:

- Each player begins with 1,000 dollars
- The players may choose how many rounds to play.
 - If there are multiple players, they must all choose the same # of rounds before the game can begin
- At the start of each round, each player will enter their own individual bet in dollars, which will be placed into the jackpot
- The goal of the game is to score **13 points** to win the jackpot
 - If a player accumulates more than 13 points, they immediately lose the round and their bet stays in the jackpot.
 - If no player achieves 13 points, then the player with the highest number of points wins, and takes the jackpot.
 - In the case of a tie (More than one player has 13 points, or more than 1 player has the highest score without going over 13), then the jackpot will be split evenly among the winning players
 - If no one wins (every player breaks 13), then no one claims the jackpot and it is forfeited.
- Each player begins with two dice. Each player then rolls both dice and the total number on both dice is the number of points the player starts the round with
 - Each players points (and their rolls) are kept private until the round is completed.
 - After the initial rolls, each player will get a turn to either request another dice which they must roll to increase their score (they may do this as many times as they would like), or they may hold. Remember that if a player breaks 13 points, they automatically lose the round.
 - The order will be randomly generated at the beginning, and must stay consistent for the entire game
- If a player runs out of money before the number of rounds expires, they must quit the game.

- The game will be called “Dicey Dice” (Creative right?)

Use Cases:

As a player, I want to record my statistics so that I can compare my scores with other users.

As a player, I want to view other user’s statistics, so that I can see how I rank.

As a player I want to be able to play a game against a computer.

As the game, I want to be able to save the score database so that I can keep track of scores when I am not open

As the game, I want to be able to load the score database so that I can keep track of the results from previous games

Use Case Scenarios

- **Record Statistics**
 - **Actors**
 - Player
 - Game
 - **Trigger**
 - User plays a game
 - **Goal**
 - The player’s ranking is updated to reflect the result of the most recent game
 - **Preconditions**
 - Player has been registered with the scoring system
 - **Tasks**
 - Player plays a game
 - Game determines winner of the game
 - Game updates Player’s ranking and win/loss history
 - **Variations**
 - Player loses
 - Game records the player as having lost the game
 - Player wins

- Game records the player as having won the game
 - Post Conditions
 - Player's ranking history has been updated
- View Statistics
 - Actors
 - Player
 - Trigger
 - Player attempts to view player rankings
 - Goal
 - Player views the rankings and match history of themselves and other players
 - Preconditions
 - Some games have been played and their scores recorded
 - Tasks
 - Player clicks on the view rankings tab
 - Player browses through rankings
 - Variations
 - Player may also search for a player by name or ranking and view their specific ranking and match history
 - Post conditions
 - Player has learned about the ranking information they wanted
- Play Computer
 - Actors
 - Player
 - Game (Computer Player)
 - Trigger
 - User opens the program and selects play game
 - Preconditions
 - Player has been registered with the scoring system
 - Tasks
 - Player chooses the number of rounds they wish to play
 - Player plays the first round
 - Player places their bet
 - Game places its bet
 - Player and Game roll their initial dice
 - Player and Game roll additional dice if they choose too
 - Player and Game reveal their points

- The pot is given to the winning player
 - Another round is played until the number of rounds is exhausted
 - The winner of the set is determined by the final amount of points between each player
 - Score is then recorded
- Variations
 - Player ends game before it is finished
 - Score is not recorded
- Post conditions
 - Player has won a game
 - Score has been recorded
- Save Database
 - Actors
 - Game
 - Player
 - Triggers
 - Game is closed
 - User selects save database
 - Goal
 - Database of scores is successfully saved to disk that it can be loaded on the next instance of the game
 - Preconditions
 - There are entries in the score database
 - Tasks
 - User starts to exit the game
 - Game saves the score database on the disk
 - Game finishes exiting
 - Variations
 - Player may instead select the option save score database from the game menu
 - Player is prompted to choose a location to save the database
 - Database is saved in the location specified to the disk
 - Postconditions
 - Score database is saved to disk
- Load Database

- Actors
 - Game
 - Player
- Triggers
 - Game is opened
 - User selects load database
- Goal
 - Database of scores is successfully loaded from disk
- Preconditions
 - There are entries in the score database
- Tasks
 - Player starts the game
 - Game loads the database from the default save location
- Variations
 - Player may instead select the option load score database from the game menu
 - Player is then prompted to select the score database to load
 - The game then loads the selected database replacing the current database in memory
- Postconditions
 - The score database has been loaded by the game and may be viewed or modified

Dicey Dice

