Use Case Deliverables

**Game Title: Tendentious Strife**

# Game Requirements

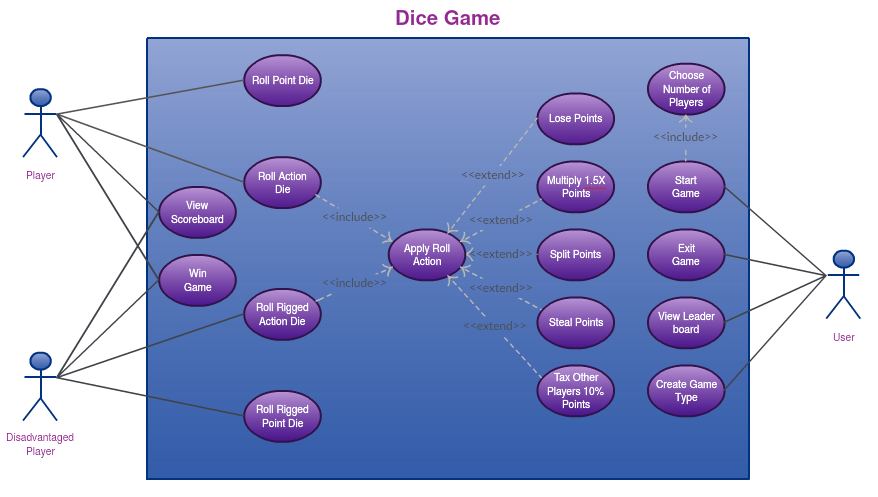
## Functional Requirements:

* Win and lose condition
* Dice must be involved
* Maintain a persistent, visible scoreboard - display user game points
* Allow 3-6 players to play at the same time
* Supports only offline multiplayer
* At the end of the game, conditions must be displayed with a message
* Game consists of two dice rolls per person
  + First roll consists of 1-10, player gains this many points
  + Second roll has many conditions:
    - **Split:** 2 players split their points 50/50
    - **Steal:** Based on points rolled, the player steals that many from another.
    - **Multiply:** Based on points rolled, the player earns 1.5x tham amount
    - **Lose Points:** Player loses the points they gained in the first roll
    - **Tax:** Take 10% of every other player’s points (rounded down)
    - **Nothing:** Simply keep the points gained in the first roll
* Game must have a disadvantaged player(s) in it (slightly rigged die)
  + Disadvantaged player is randomly assigned to a player at the beginning of the game
* Game timer to ensure continuation of game

## Nonfunctional Requirements:

* Game must be visually appealing
* Can’t take more than 10 minutes to set up
* Must be entertaining to play- people should want to play the game
* Game must be maintainable and resilient
* Game should be user modifiable
  + Support different game variants made by us and other players(eventually)
* Game should have a fast response time to user selections
* Game must be robust and stable with no crashes
* Strong documentation and support for game use and play
* Game should have a tutorial to show new players how to play the game
* Keep track of statistics locally on the user’s computer. Track if the game has been completed once or not.
* Game menu should change slightly after the first game played. An ability to create new game types, as well as include a few that the designers incorporated as well.(as easy as changing various variable values such as time, win-loss point conditions. These are to be stored locally.
* Explain the theme/message involved with having a disadvantaged player at the end of the first game.

# Use Case Diagram



# Use Case Scenarios

###### User Scenario: Normal Player

1. roll\_Regular\_Dice

**Description:** Normal Player(s) may be able to roll a regular 10 sides point dice.

**Actors:** Normal(non-disadvantaged) Player, Scoreboard

**Triggers:**

1. User starts a game

2. It is a user's turn to roll

**Precondition:**

1. Player is a non-disadvantaged user

2. It is a normal player's turn

**Scenario:** When Normal Player starts a game along with the other players he or she will have to roll two die to play the game. When they roll a point dice the points will get incremented to their score.

**Exception:**

1. There are not enough players (unable to start the game)

2. When timer reaches the end of the game time

**Alternative flow:** None

2. Roll\_Regular\_Condition\_Dice

**Description**: Normal Player(s) along with the first point dice, will roll a secondary(condition) dice.

**Actors**: Normal Player, Scoreboard, Other Players

**Triggers:**

1. When user rolls a dice

**Precondition**:

1. Player is a normal player

2. Player's turn to roll

**Scenario:** When Normal Player has to roll, whether by starting a game or it has became his or her turn to play, they will roll the secondary dice which has the special conditions that will effect their own / or other player's points. After the roll scoreboard will update the gained condition.

**Exception:**

1. Player didn't roll the point dice

2. Timer reaches the end.

**Alternative flow:** \*Main Player signifies the player rolling the dice at the moment

1. Main player rolls a 'split' condition:

1-a. Main player needs to select another player. Then they will combine their points and split them in half.

2. Main Player rolls a 'steal' condition:

2-a. Main player needs to select another player. Main player can steal the number of points they rolled from that player.

2-b. Unable to steal points if a selected player doesn't have any points(0) on their score.

3. Main Player rolls a 'multiply' condition.

3-a. Recalculate the main player's point by 150%(\*1.5).

4. Main Player rolls a 'lose points' condition.

4-a Subtract the main player's point by the number they rolled with their point dice

5. Main Player rolls a 'no action' condition.

5-a. No effect, (basically means the end of the main player's turn)

6. Main Player rolls a 'tax' condition.

6-a. All other players (players that are not the main player) must donate 10% of their current points to the main player.

3. Lose\_Game

**Description**: When someone else gains 100 points that player wins the game and rest will lose the game

**Actors**: Normal Player, Scoreboard, Game System, Other Players

**Triggers**:

1. Some Player(that is not the main player) reaches 100 points

2. It is end of (any) players turn

**Precondition**:

1. Main Player does not have 100 points

2. Other Player have 100 points

**Scenario**:

1. At the end of a player(any)'s turn, the game system will checks the scoreboard
2. If some player (that is not the main player) has 100 points when G.S checks the score game will end.

**Exception**:

1. Two people end with 100 points.

**Alternative Flow:**

1. No one has 100 points

1-a. Game continues

2. It is not the end of the turn

2-a. Game System will not check the scoreboard until it's the end of the player's turn (condition dice taking effect)

3. Main Player has 100 points

3-a. Can't lose when M.P have 100 points (it means M.P have won)

4. Win\_Game

**Description**: When the main player earns 100 points game ends and M.P wins the game

**Actors**:

Main Player, Game System, Scoreboard

**Triggers**:

End of the turn (condition dice taking effect)

**Precondition**:

1. Main Player has 100 points

2. Other Players does not have 100 points

3. It's end of player's turn

**Scenario**:

1. At the end of a player(any)'s turn the game system checks the scoreboard

2. When Main Player has the total of 100 points game will end and M.P wins the game

**Exception:**

1. Two players gain 100 points in the same round.

**Alternative Flow:**

1. It is not the end of a player(any)'s turn

1-a. G.S will not check the score if it's not the end of a player's turn.

2. Main Player does not have 100 points

3. Another Player have 100 points

3-a. Main player cannot win when someone else have already won

5. Check\_Score

**Description**: Displays the scoreboard on the side

**Actors**: Players(any), Scoreboard

**Triggers**:

1. When game starts

**Precondition**:

1. Game is being played

**Scenario**:

1. Display the scoreboard on the side so that players can check their scores

**Exception:**

1. Game is not being played

**Alternative Flow**

1. Scoreboard does not display correctly regardless of the reason (bug, error, etc)

1-a. Display Error Message

6. Check\_Time

**Description**: Display the timer on the side

**Actors**: Player(any), Timeboard

**Triggers**:

1. When the game starts

**Precondition**:

1. Game is being played

**Scenario**:

1. Display the timeboard on the side so that the players can check the remaining game time?

**Exception:**

1. Game is not being played

**Alternative Flow**

1. Timeboard is not displaying correctly regardless of the reason(bug, error, etc)

1-a. Display error message

###### User Scenario: Disadvantaged Player

1. Roll a rigged first die

**Primary Actors:** Disadvantaged player

Scoreboard

**Preconditions:** Player is disadvantaged

It is the disadvantaged player's turn

**Basic Flow of Events:**

1. Player rolls the rigged 1st die (1-10).

2. The scoreboard will add that amount to the player's score.

**Alternative Flows:**

None

2. Roll a rigged second die

**Primary Actors:** Disadvantaged player

Scoreboard

Another player(s)

**Preconditions:** Player is disadvantaged

It is the disadvantaged player's turn

The player has already rolled the 1st die

**Basic Flow of Events:**

1. Player rolls the rigged 2nd die (1-X [undetermined]).

2. Based on the value rolled, an action takes place.

3. The scoreboard updates the action(s) that have taken place.

**Alternative Flows:**

2a. Player rolls a 'split'.

2a1. Player must select another player with unequal points. They combine their points and split them evenly (round down).

2b. Player rolls a 'steal'.

2b1. Player selects another player. They steal that many points from that player.

2b2. If a player is at zero points, they cannot be stolen from (nothing to steal).

2c. Player rolls a 'multiply'.

2c1. Player multiplies their points by 1.5. This is their new point total.

2d. Player rolls a 'lose points'.

2d1. Player loses the points they rolled in the first roll.

2e. Player rolls a 'no action'.

2e1. Player performs no action for the second roll.

2f. Player rolls a 'tax'.

2f1. Every player except the roller must give the roller 10% of their points (rounded down).

3. Lose the Game

**Primary Actors:** Disadvantaged Player

Scoreboard

Game System

**Preconditions:** Disadvantaged player does not have 100 points.

Another player has 100 points.

It is the end of a player's turn.

**Basic Flow of Events:**

1. At the end of a player's turn, the game system checks the scoreboard to see if any player has 100 points.

2. If the player with 100 points is not the disadvantaged player, the disadvantaged player loses the game.

**Alternative Flows:**

1a. No player has 100 points.

1a1. The disadvantaged player cannot yet lose. Nothing happens.

1b. It is not the end of a player's turn.

1b1. No player may win until the end of a turn, even if they have over 100 points. Continue that turn until it reaches the end.

2a. The disadvantaged player has 100 points.

2a1. Since it is impossible to simultaneously gain points, only one player can have 100 points at the end of a turn. Thus, the disadvantaged player cannot lose, and has instead won. See (4. Win the Game).

4. Win the Game

**Primary Actors:** Disadvantaged Player

Scoreboard

Game System

**Preconditions:** Disadvantaged player has 100 points.

No other player has 100 points.

It is the end of a player's turn.

**Basic Flow of Events:**

1. At the end of a player's turn, the game system checks the scoreboard to see if any player has 100 points.

2. If the player with 100 points is the disadvantaged player, the disadvantaged player wins the game.

**Alternative Flows:**

1a. No player has 100 points.

1a1. The disadvantaged player cannot yet win. Nothing happens.

1b. It is not the end of a player's turn.

1b1. No player may win until the end of a turn, even if they have over 100 points. Continue that turn until it reaches the end.

2a. Another player has 100 points.

2a1. Since it is impossible to simultaneously gain points, if another player has 100 points at the end of a turn, the disadvantaged player has instead lost. See (3. Lose the Game).

5. Check the Scoreboard

**Primary Actors:** Disadvantaged player

Scoreboard

**Preconditions:** Game is ongoing

**Basic Flow of Events:**

1. Scoreboard is persistently displayed on the side for player to view.

**Alternative Flows:**

1a. Scoreboard system not display correctly

1a1. Display error message.

###### User Scenario: Main Menu User

1. Start a Game

**Primary Actors:** User

Scoreboard

**Preconditions:** User is at main menu

User has not started a game yet

**Basic Flow of Events:**

1. The user presses the *Start Game* button.

2. The user selects how many players are in the game (3-6).

3. The scoreboard system initializes the in-game scoreboard (resets it in case it was used).

4. After this, the game will begin.

**Alternative Flows:**

1a. Game does not start.

1a1. Display error; return to main menu.

2a. User selects a number outside of the range from 3-6.

2a1. Display error message; prompt number from 3-6.

3a. Scoreboard does not get reset.

3a1. Display error message.

2. View the Leaderboard

**Primary Actors:** User

Scoreboard

**Preconditions:** User is at main menu

User has not started a game yet

**Basic Flow of Events:**

1. The user presses the *View Leaderboard* button.

2. The scoreboard system retrieves the overall scoreboard and displays it to the user.

**Alternative Flows:**

2a. Scoreboard does not work.

2a1. Display error message; return to main menu.

3. Quit the Game

**Primary Actors:** User

**Preconditions:** User is at main menu.

**Basic Flow of Events:**

1. Player presses the *Quit Game* button.

2. Game exits.

**Alternative Flows:**

1a. User is currently in-game.

1a1. Use the alternate button (still labeled as *Quit Game*).

2a. Game does not exit.

2a1. Display error message, recommending user to forcefully close.

## User Stories

###### User

As a user I want to be able to start a new game in order to start playing the game.

As a user I want to be able to view the scoreboard to view previous games and player stats.

As a user I want to quit the game in order to exit the application

As a user when I start the game I want to be able to choose the amount of players in order to match the amount of people who will play the game on the same machine (3-6)

As a user of a game that has already been completed once, I want to be able to create new gametypes in order to modify the game to better suit the needs of me and my current company.

As a user of a game that has already been completed once, I want to be able to start a new game under a custom gametype in order have more fun with the game and prevent it from going stale.

###### Player

As a player I want to be able to roll my regular point die when it is my turn in order to progress the game

As a player I want to roll my regular action die to determine what action I can perform this turn.

As a player If I roll a steal I want to be able to take an amount of points away from another player of my choosing in order to try to bring myself closer to my personal victory.

As a player if I roll a split I want to be able to split the sum of my points and another player of my choosing in order to even out my chances of winning with preferably a player that has more points than me.

As a player if I roll a multiply I want to earn 1.5X the amount of points that I would have received from my previous point roll in order to bring myself closer to personal victory.

As a player if I roll a lose points action I want to lose the points that I would have earned this turn in order to introduce some amount of balance to the game’s actions.

As a player if I roll a tax I want to be able to take a 10% tax off of the total points owned by each player in order to bring myself closer to victory.

As a player I want to be able to win the game in order to give me some kind of incentive for playing the game.

###### Disadvantaged Player

As a disadvantaged player I want to be able to roll my rigged point die when it is my turn in order to progress the game.

As a disadvantaged player I want to roll my rigged action die to determine what action I can perform this turn.

As a disadvantaged player If I roll a steal I want to be able to take an amount of points away from another player of my choosing in order to try to bring myself closer to my personal victory.

As a disadvantaged player if I roll a split I want to be able to split the sum of my points and another player of my choosing in order to even out my chances of winning with preferably a player that has more points than me.

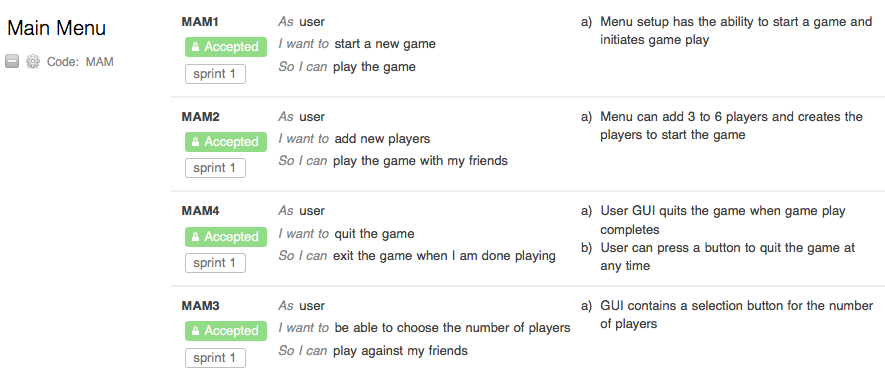
As a disadvantaged player if I roll a multiply I want to earn 1.5X the amount of points that I would have received from my previous point roll in order to bring myself closer to personal victory.

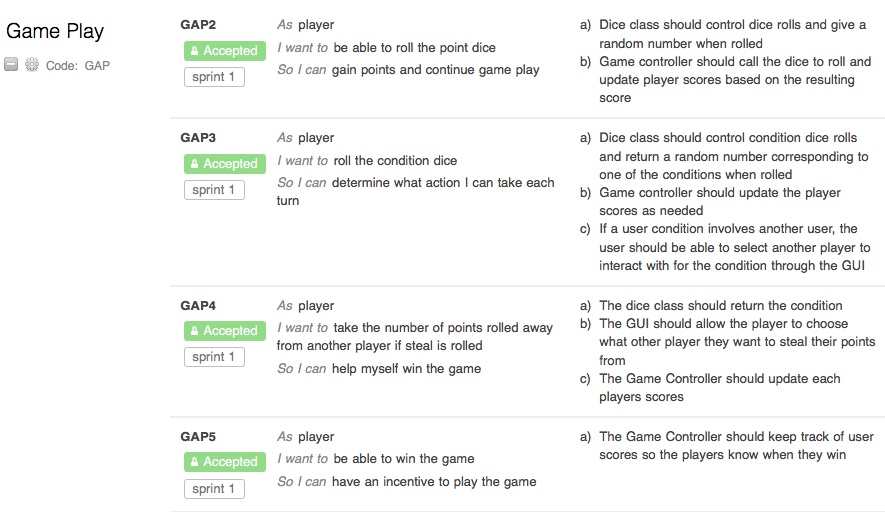
As a disadvantaged player if I roll a lose points action I want to lose the points that I would have earned this turn in order to introduce some amount of balance to the game’s actions.

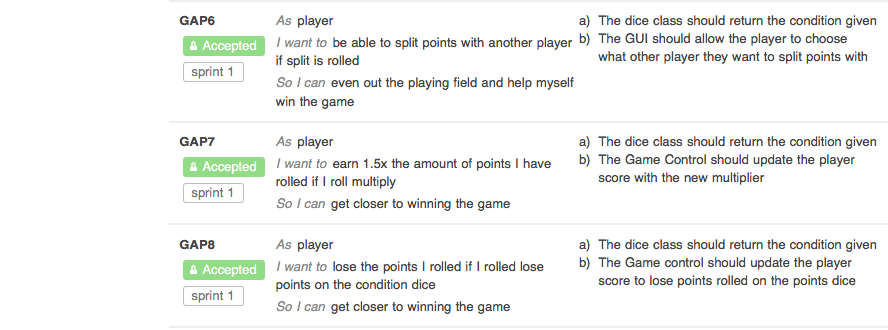
As a disadvantaged player if I roll a tax I want to be able to take a 10% tax off of the total points owned by each player in order to bring myself closer to victory.

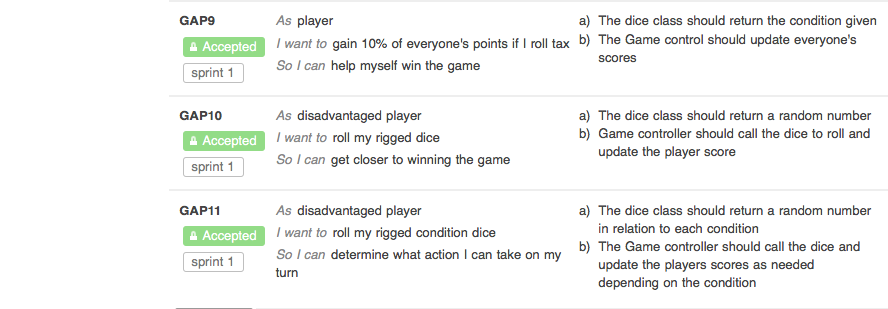
As a disadvantaged player I want to be able to win the game in order to give me some kind of incentive for playing the game.

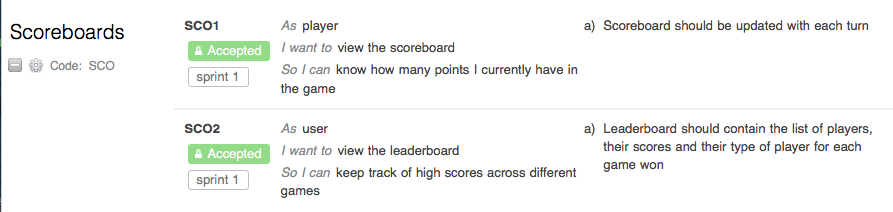
# Backlog











# Class Diagram

