TeamTris

CREATED BY

STEVEN DELLAMORE
dellamoresteven@gmail.com
TeamTris
CS407

Contents

1 StartScreen

Author: Steven Dellamore
Description: Startscreen will build the startscreen and create all the buttons needed for the user to get into a game with their friends. The mouseClicks and the keyboard imports all all forwarded to this class when gamestate == 0

1.1 constructor

Author: Steven Dellamore

```
constructor ()
```

Description: The constructor gets called when making a startscreen object. It will init all the values and set up the socket listener for the server to send things too. Here are the init values of the class variables:

```
this.TokenBoxText = "";
this.usernameBoxStroke = false;
this.usernameText = "username";
this.usernameTextTouched = false;
this.gameStateStartScreen = 0;
this.titleAnimation = [300, 500, 400, 700];
```

These varibles will be updated throughout the life of start screen. `this.TokenBoxText` will init the token box to nothing, since the user has yet to do anyhting. the `this.usernameBoxStroke` will be set to false so the program knows if the user as tried to sumbit. `this.titleAnimation = [300, 500, 400, 700];` is the starting position of the title, and will fall every X frames.

Parameters:
void : constructor takes no params

Returns:
StartScreen : An object of start class class

1.2 draw

Author: Steven Dellamore

$draw\,(\,)$

Description: This funcion will be ran at 60 frames a second and will call all the functions needed to draw the launch screen. The draw function will call the title functions, the highscore functions, and call the join and create button rendering/hitboxes with `Buttonloop()`. Depending on what `this.gameStateStartScreen` is evaluated to.

```
switch (this.gameStateStartScreen) {
        case 0:
                this.drawUsernameBox();
                break;
        case 1:
                this.drawTokenBox();
                break;
}

0 = this.drawUsernameBox, 1 = this.drawTokenBox
```

Parameters:
void : draw takes no arugments

Returns:
void : something should go ehre

1.3 animateTitle

Author: Steven Dellamore

$animate\,Title\,(\,)$

Description: Will check and add/subtract the locations of the T's falling when you go to the launch screen.

```
if (this.titleAnimation[i] > 0) {
        this.titleAnimation[i] -= 10;
}
```

Once `this.titleAnimation[i]`, where `i` is between [0,4], is negative, the array index will no longer be decremented.

Parameters:
void : animateTitle takes no aruguments

Returns:
void

1.4   drawUsernameBox

Author: Steven Dellamore

drawUsernameBox()

Description: This function will draw the white username box onto the screen displaying the `this.usernameText` in the center. This function will also use `this.usernameBoxStroke` to display the red outline around the username box.

Parameters:
void : drawUsernameBox takes no aruguments

Returns:
void

1.5   drawTitle

Author: Steven Dellamore

drawTitle()

Description: This function will draw the title (Teamtris) onto the launch screen. Also, the function will be responable for displaying the current falling location of the two T's falling at the start of the screen. We make rects based on the current location of `this.titleAnimation`.

```
let yStart;
rect(-windowWidth / 4.3, (yStart = windowHeight / 2.6) -
                this.titleAnimation[0], squareSize, squareSize)


rect(-windowWidth / 4.3, (yStart - (spaceBetweenSquares)) -
                this.titleAnimation[0], squareSize, squareSize)


fill(255, 0, 0) // fill red


rect(-windowWidth / 4.3, yStart - (2 * spaceBetweenSquares) -
                this.titleAnimation[1], squareSize, squareSize)


rect(-windowWidth / 4.3 - spaceBetweenSquares,
                yStart - (2 * spaceBetweenSquares) - this.titleAnimation[1],
                           squareSize, squareSize)


rect(-windowWidth / 4.3 + spaceBetweenSquares,
                yStart - (2 * spaceBetweenSquares) - this.titleAnimation[1],
                           squareSize, squareSize)
```

The important thing to note is to see the y val of the rect is being changed by 10 every frame in `function animateTitle()`.

Parameters:
void : drawTitle takes no arugments

Returns:
void

1.6   drawTokenBox

Author: Steven Dellamore

drawTokenBox ( )

Description: This function will draw the token box once the user clicks "join game". It will display the token box and the accept button. Unlike other buttons, all mouse clicks are handled.

Parameters:
void : drawTokenBox takes no aruguments

Returns:
void

1.7 mouseClickedStart

Author: Steven Dellamore

$$mouseClickedStart()$$

Description: This function is being called whenever `gamestate = 0` AND the user clicks their mouse. First, we will check what `this.gameStateStartScreen` is. If its `0`, we will check the `function ClickedLoop()` to see if the user is clicking on the join game, create game, or highscore score buttons. If the user clicks on a the create game button with a valid username we are going to send them into the lobbyscreen.

```
// Creating my lobbyscreen object
mLobbyScreen = new LobbyScreen(
        new Player(
                this.usernameText, Math.floor(Math.random() * 100), true));

gameState = 1; // Switch to lobby screen
```

We need to create a new Player, and set their ownership value to 0. We see its constructor defined here:

```
constructor(username, id, owner){
    this.username = username;
    this.id = id;
    this.owner = owner;
```

6

```
    this.playerNum;
}
```

We then pass this object into the lobbyscreen and switch the `gameState = 1` to move us

Parameters:
void : mouseClickedStart takes no aruguments

Returns:
void

1.8   drawHighScoreButtonCheckMouse

Author: Steven Dellamore

drawHighScoreButtonCheckMouse ( )

Description: This function is being called whenever the user clicks with gamestate of the startscreen == 0. This function checks if the mouse is over the highscore button and returns true if it is, false if its not.

Parameters:
void : drawHighScoreButtonCheckMouse takes no aruguments

Returns:
bool : true => If mouse is over score button false => If mouse is not over score button

1.9   drawHighScoreButton

Author: Steven Dellamore

drawHighScoreButton ( )

Description: This function will draw the three bars in the bottom left of the screen. It will also check if the mouse is over the button and highlight according.

Parameters:
void : drawHighScoreButton takes no arugments

Returns:
void

1.10   keyPressedStart

Author: Steven Dellamore

```
keyPressedStart ()
```

Description: This function will be called whenever the user clicked on a button on the start screen. general/keyPressed.js is where this function will be called.

Parameters:
void : keyPressedStart takes no arugments

Returns:
void

2   Team

Author: Steven Dellamore, Richard Hansen
Description: The team class will contain all the other players that are in your game, the team name and the token for your lobby. Once new players come addPlayer will be called to push a newplayer onto the playersInTeam

array.

2.1   constructor

Author: Steven Dellamore

constructor ()

Description: The constructor gets called anytime someone joins or create a game.

Parameters:
void: no parameters

Returns:
Team : A object of the class

2.2   addPlayer

Author: Steven Dellamore

addPlayer (player)

Description: The add player function gets called whenever a bot or a real player joins your lobby. This function will also be called to populate the lobby when you join.

Parameters:
Player player: This parameter is the new player/bot that is joining your team.

Returns:
void : no return

3 Player

Author: Steven Dellamore, Richard Hansen
Description: Every user will have their own object of the Player class. This is going to be passed around to other people in the lobby. This class will tell the game screen who is who and will help identify moves.

3.1 constructor

Author: Steven Dellamore

constructor (username, id, owner)

Description: The constructor takes in three things, a name, id and a owner flag. It will then create an object of `Player` and init all class varibles. This Class is used throughout all stages of the program.

Parameters:
String username : username of the new Player
int id : id, `[0,4]`, of the new player.
boolean owner : `true` or `false` if they are owner

Returns:
Player : An object of Player class

3.2 setPlayerNum

Author: Steven Dellamore

setPlayerNum (num)

Description: Will set `this`.`playerNum` equal to `num`. This is just a helper function.

Parameters:
int num : sets the `this`.playerNum = num

Returns:
void : returns nothing

4   General

Author: Steven Dellamore, Richard Hansen
Description: This is an abstract class that will hold mouseClicked and key-Pressed p5 functions.

4.1   mouseClicked

Author: Steven Dellamore, Richard Hansen

mouseClicked ( )

Description: Will be called whenever the user clicks on anywhere on the screen. Once called, it will go straight into a switch to decide where to route to based on the gameState

```
switch (gameState) {
    case 0:
        // start screens mouseClicked
        mStartScreen.mouseClickedStart();
        break;
    case 1:
        // lobby screens mouseClicked
        mLobbyScreen.mouseClickedLobby();
        break;
    case 2:
        break;
    case 3:
        break;
}
```

The varibles `gameState`, `mStartScreen`, `mLobbyScreen` are all defined in sketch.js

Parameters:
void : takes no parameters

Returns:
void : returns nothing

4.2   keyPressed

Author: Steven Dellamore, Richard Hansen

mouseClicked ( )

Description: Will be called whenever the presses a key. Once called, it will
go straight into a switch to decide where to route to based on the gameState

```
switch (gameState) {
  case 0:
    mStartScreen.keyPressedStart();
    break;
  case 1:
    mLobbyScreen.keyPressedLobby();
    break;
  case 2:
    mGameScreen.keyPressedGame();
    break;
  case 3:
    // mScoreScreen.keyPressedScore();
    break;
}
```

The varibles `gameState`, `mStartScreen`, `mLobbyScreen` are all defined in sketch.js

Parameters:
void : takes no parameters

Returns:
void : returns nothing