# ThePiHost

**Created By:** Austin Griffith, Joshua Griffiths, Caroline Friedman, Galen Pogoncheff

**Project Tracker (Trello):** https://trello.com/b/UeRWSgir/part-1-project-tracker



**VCS (GitHub):** https://github.com/carolinefriedman/ThePiHost

**Contributions:**

Caroline's Contributions:

Caroline's Contributions (Continued):

| | | | |
|---|---|---|---|
| Branch: master ▾ **ThePiHost** / **Website** / | | Create new file | Upload files | Find file | History |

caroline**friedman** committed on **GitHub** update index      Latest commit 3da42bd an hour ago

..

| 📁 assets | master branch maintenance | 22 hours ago |
|---|---|---|
| 📁 css | master branch maintenance | 22 hours ago |
| 📁 scripts | master branch maintenance | 22 hours ago |
| 📄 .DS_Store | master branch maintenance | 22 hours ago |
| 📄 index.html | update index | an hour ago |

Austin's Contributions:

📖 **carolinefriedman** / **ThePiHost**

| <> Code | ⓘ Issues **2** | Pull requests **0** | Projects **0** | Wiki | Pulse |
|---|---|---|---|---|---|

Branch: master ▾    **ThePiHost** / **Emulator** /

**augr4882** adding emulator app and Games

..

| 📁 Nintendo 64 | adding emulator app and Games |
|---|---|
| 📁 OpenEmu.app/Contents | adding emulator app and Games |
| 📄 .DS_Store | adding emulator app and Games |

Josh's Contributions:

69 lines (56 sloc) | 1.45 KB

```java
1    import java.awt.Graphics;
2    import java.awt.Color;
3    import java.awt.Rectangle;
4
5    public class AlienBomb{
6      public int x;
7      public int y;
8      public int bombSpeed = 1;
9      public final int width = 3;
10     public final int height = 6;
11     public boolean inPlay;
12     public boolean isShooting = false;
13     public boolean moveLeft = false;
14     public boolean moveRight = false;
15     public int moveSpeed = 1;
16     Rectangle boundingBox;
17     boolean collision = false;
18
```

75 lines (62 sloc) | 1.8 KB

```java
1    import java.awt.Graphics;
2    import java.awt.Color;
3    import java.awt.Rectangle;
4
5    public class Bullets{
6      public int x;
7      public int y;
8      public int bulletSpeed = 12;
9      public final int width = 3;
10     public final int height = 3;
11     public int playerScore = 0;
12     public boolean inPlay;
13     public boolean isShooting = false;
14     public boolean moveLeft = false;
15     public boolean moveRight = false;
16     public int moveSpeed;
17     Rectangle boundingBox;
18     boolean collision = false;
19     public int numShots = 1;
20
```

Josh's Contributions (Continued):



```java
import java.awt.Graphics;
import java.awt.Color;
import java.awt.Rectangle;

public class Shooter{
    public int x;
    public int y;
    public int health = 100;
    public final int width = 30;
    public final int height = 10;
    public int moveSpeed = 3;
    public boolean moveLeft = false;
    public boolean moveRight = false;

    Bullets bullet;
    Rectangle boundingBox;

    //contructor
    public Shooter(int x, int y){
        boundingBox = new Rectangle(x, y, width, height);
```

# Galen's Contributions:

Create new file | Upload files | Find file | History

gpogoncheff javadoc updates                                   Latest commit 9655a2d 23 hours ago

..

| 📁 bin/code | Added database acces for tennis highscores | 6 days ago |
| 📁 libs | Added database acces for tennis highscores | 6 days ago |
| 📁 src | javadoc updates | 23 hours ago |
| 📄 .DS_Store | javadoc updates | 23 hours ago |

```java
 69    public Tennis(){
 70      frame = new JFrame();
 71      this.setMinimumSize(gameSize);
 72      this.setPreferredSize(gameSize);
 73      this.setMaximumSize(gameSize);
 74      frame.add(this, BorderLayout.CENTER);
 75      frame.pack();
 76      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 77      frame.setVisible(true);
 78      frame.setResizable(false);
 79      frame.setTitle(TITLE);
 80      frame.setLocationRelativeTo(null);
 81
 82      /** user controlled paddle */
 83      player = new PlayerPaddle(10, 60);
 84      /** automated paddle */
 85      compplayer = new ComputerPaddle(getWidth() - 25, 60);
 86      ball = new Ball((getWidth()/2), (getHeight()/2));
 87      io = new GameIO(this);
 88    }
 89
 90    /**
 91     * Start point of execution for the thread.
 92     *
 93     * Reason that thread should sleep for a short amount of time:
 94     * http://stackoverflow.com/questions/20634600/why-does-a-game-loop-need-to-sleep
 95     * Smaller increments of sleep time were tried here, but the movement of figures
 96     * on the screen became overly sensitive to keyboard input.
 97     */
 98    public void run(){
 99      while (gameRunning){
100        tick();
101        render();
102        try{
103          Thread.sleep(5);
104        }
105        catch(Exception e){
106          e.printStackTrace();
107        }
108      }
109    }
110    /**
111     * Method used to start executing gameplay instructions.
112     */
113    public synchronized void start(){
114      gameRunning = true;
115      new Thread(this).start();
116    }
117
118    public static synchronized void stop(){
119      System.exit(0);
120    }
```

# Galen's Contributions (Continued):

```java
184    private void dbUpdate(Graphics graphics) throws SQLException{
185      Connection conn = null;
186      PreparedStatement updateStmt = null;
187      PreparedStatement readStmt = null;
188      DateTimeFormatter dateFormat = DateTimeFormatter.ofPattern("MM/dd/yyyy HH:mm:ss");
189      LocalDateTime dateNow = LocalDateTime.now();
190
191      try{
192        conn = (Connection) DriverManager.getConnection(CONN_STRING, USERNAME, PASSWORD);
193        String updateQuery = "insert into tennisScores (date, score) values(?, ?)";
194        updateStmt = (PreparedStatement) conn.prepareStatement(updateQuery);
195        updateStmt.setString(1, dateFormat.format(dateNow));
196        updateStmt.setInt(2, playerScore);
197        updateStmt.execute();
198        String readQuery = "SELECT date, score FROM tennisScores ORDER BY score DESC LIMIT 3";
199        readStmt = (PreparedStatement) conn.prepareStatement(readQuery);
200        ResultSet highScores = readStmt.executeQuery();
201        int printLocation = HEIGHT/2 + 90;
202        while (highScores.next()){
203          graphics.drawString("Date: " + highScores.getString(1) + ", Score: "+ highScores.getString(2), WIDTH/2
204          printLocation += 30;
205        }
206
207      } catch (SQLException e){
208        System.err.println(e);
209
210      } finally {
211        if (readStmt != null){
212          readStmt.close();
213        }
214        if (updateStmt != null){
215          updateStmt.close();
216        }
217        if (conn != null){
218          conn.close();
219        }
220      }
221    }
```

Branch: master ▾          Create new file   Upload files   Find file   History

**ThePiHost** / spaceDefendersDB / SpaceDefendersDB / src / **code** /

⊤ **gpogoncheff** Final javadoc documentatino added for rest of space defenders classes    Latest commit 7cbb804 2 days ago

..

| | | |
|---|---|---|
| 📄 Alien.java | Added full javadoc documentation for Alien class | 2 days ago |
| 📄 AlienBomb.java | Final javadoc documentatino added for rest of space defenders classes | 2 days ago |
| 📄 Bullets.java | Final javadoc documentatino added for rest of space defenders classes | 2 days ago |
| 📄 GameIO.java | Final javadoc documentatino added for rest of space defenders classes | 2 days ago |
| 📄 Shooter.java | Final javadoc documentatino added for rest of space defenders classes | 2 days ago |
| 📄 spaceDefender.java | Added full javadoc documentation | 2 days ago |

# Galen's Contributions (Continued):

```java
123    public void tick(){
124      player.tick(this);
125
126      /**
127       * Select a randomly alien to drop a bomb every period specified by timecount
128       */
129      if (timeCount != 0 && timeCount % dropTimer == 0){
130        int x = random.nextInt(3);
131        int y = random.nextInt(10);
132        this.alienMatrix[x][y].bomb.isShooting = true;
133      }
134
135      /**
136       * if all aliens are dead, increase level, reset alien
137       * positions, make aliens mover faster, and make bombs
138       * drop more faster.
139       */
140      boolean allDead = true;
141      for (int i = 0; i < this.alienRows; i ++){
142        for (int j = 0; j < this.alienCols; j++){
143          alienMatrix[i][j].tick(this);
144          allDead = allDead && alienMatrix[i][j].isDead;
145          if (alienMatrix[i][j].y >= player.y)
146            player.health = 0;
147        }
148      }
149
150      if (allDead == true){
151        this.level++;
152        for (int i = 0; i < this.alienRows; i ++){
153          for (int j = 0; j < this.alienCols; j++){
154            alienMatrix[i][j].isDead = false;
155            alienMatrix[i][j].x = 20 + (25*j);
156            alienMatrix[i][j].y = 25 * i;
157            alienMatrix[i][j].moveSpeed = 2;
158            alienMatrix[i][j].bomb.bombSpeed++;
```

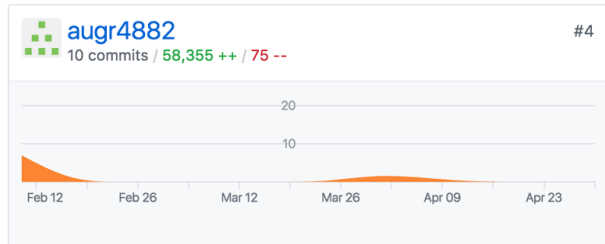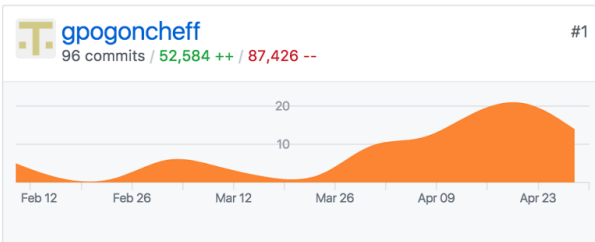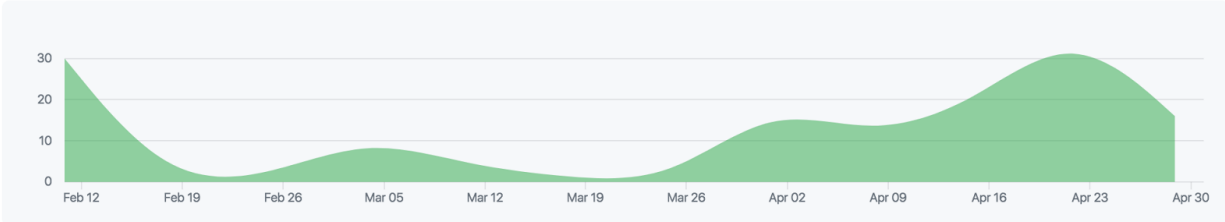| Branch: master ▾ | **ThePiHost** / **AutomatedTesting** / | | Create new file | Upload files | Find file | History |
|---|---|---|---|---|---|---|

| | gpogoncheff master branch maintenance | | Latest commit 9d4baa2 23 hours ago |
|---|---|---|---|

.. 

| 📁 AutomatedTesting_SpaceDefenders | master branch maintenance | 23 hours ago |
|---|---|---|
| 📁 AutomatedTesting_Tennis | master branch maintenance | 23 hours ago |
| 📄 .DS_Store | master branch maintenance | 23 hours ago |
| 📄 TESTING.md | master branch maintenance | 23 hours ago |

| Branch: master ▾ | **ThePiHost** / **Documentation** / | | Create new file | Upload files | Find file | History |
|---|---|---|---|---|---|---|

| | gpogoncheff Added pdf forms of javadocs | | Latest commit fbd9a96 23 hours ago |
|---|---|---|---|

.. 

| 📁 Documentation_PDFs | Added pdf forms of javadocs | 23 hours ago |
|---|---|---|
| 📁 Javadoc-SourceHTML | Added pdf forms of javadocs | 23 hours ago |
| 📄 .DS_Store | Added pdf forms of javadocs | 23 hours ago |

Everyone:



gpogoncheff #1
96 commits / 52,584 ++ / 87,426 --

carolinefriedman #2
30 commits / 9,422 ++ / 3,600 --

joshuaGriffiths #3
14 commits / 137 ++ / 12 --

augr4882 #4
10 commits / 58,355 ++ / 75 --

# ThePiHost

## group members (username, name)

*augr4882*, Austin Griffith

*carolinefriedman*, Caroline Friedman

*gpogoncheff*, Galen Pogoncheff

*joshuaGriffiths*, Josh Griffiths

## Title of project

Who: Joshua Griffiths, Austin Griffith, Caroline Friedman, Galen Pogoncheff

Title: The Pi Host

Vision: Bringing the retro 90's games back to life for everyone to play anywhere anytime because the best games are free.

Screenshots of Automated Test Results for SpaceDefender game:

https://cloud.githubusercontent.com/assets/24575285/24729030/a608392e-1a19-11e7-8346-00dd8deb15e5.png

https://cloud.githubusercontent.com/assets/24575285/24729031/a7756c46-1a19-11e7-8ca4-e7a4ef6766c5.png

https://cloud.githubusercontent.com/assets/24575285/24729038/aa0ffad4-1a19-11e7-9d49-a0b6e9e36848.png

https://cloud.githubusercontent.com/assets/24575285/24729044/ac85a91c-1a19-11e7-8020-6e2dd8b6b1c4.png

https://cloud.githubusercontent.com/assets/24575285/24729050/af6ae9c6-1a19-11e7-990a-befcc7ec7310.png

**Deployment Environment:**
There are two ways of accessing our website from which you can download out application:

  This first method of accessing our website at the url that is hosted by the raspberryPi, thepihost.sytes.net .  In order to access the website hosted by the raspberryPi, the Pi server must must on and hosting.

  The second method of accessing our website is at http://creative.colorado.edu/~cafr3980/the-pi-host/ .  This location offers an identical website as the one hosted by the Pi except that it can be accessed at any time.

Once at our website, simply scroll down the page to the download section and click on "DOWNLOAD" to obtain the application package.  The following steps and dependencies must be observed to launch our application:

  Dependencies:
- Program must be run on a Mac computer
- User must have Java installed

  Download Steps:
1. Place the downloaded archive file in your directory of choice.
2. Untar the file with the command *tar -xvzf ThePiHostApp.tar.gz*
3. Enter uncompressed directory and double click *ThePiHostLauncher.jar*
4. Enjoy

**Documentation (Javadoc):**

Documentation can be found at:
https://github.com/carolinefriedman/ThePiHost/tree/master/Documentation

Contained at this location are both PDF versions of the javadoc documentation (located in Documentation_PDFs) and the HTML source generated by Javadoc (located in Javadoc-SourceHTML).