MOBILE COMPATIBLE GAMES

What is our GOAL for this MODULE?

We will make the Trex game Mobile Compatible.

What did we ACHIEVE in the class TODAY?

- Adjusted the game dimension to fit all screen sizes.
- Made the game touch-friendly for smartphones.
- Hosted the game online on GitHub.
- Built a web wrapper around the game to generate an APK file.

Which CONCEPTS/ CODING BLOCKS did we cover today?

- OOP concept revision added more functionality
- Dimensions
- Touches for smartphones
- Generating APK files

How did we DO the activities?

1. Make Trex game mobile compatible by adjusting the game dimensions to fit all the screen sizes. Adjust the width and height of sprites according to the screen size:

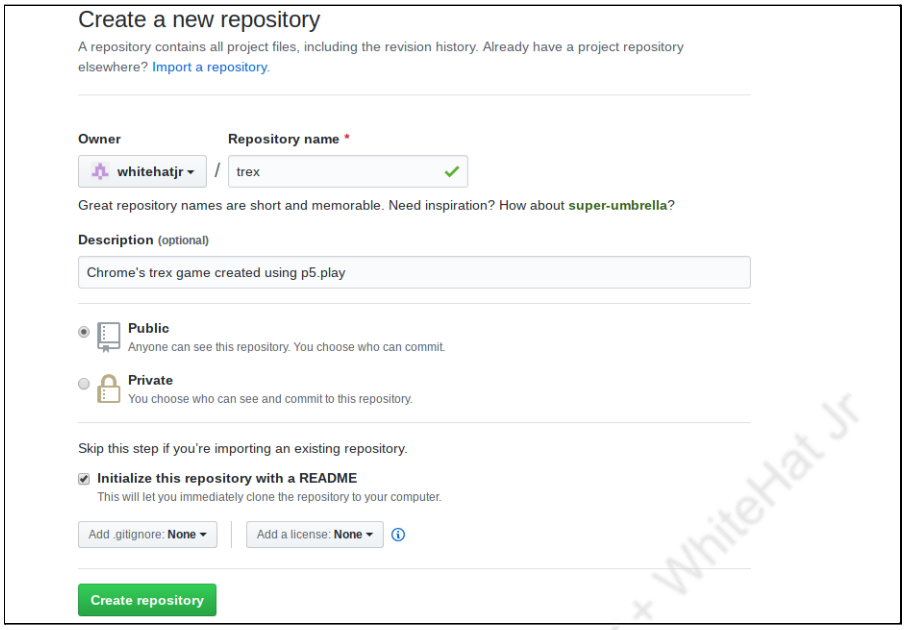
```
function setup() {
  createCanvas(windowWidth, windowHeight);

  sun = createSprite(width-50,100,10,10);
  sun.addAnimation("sun", sunAnimation);
  sun.scale = 0.1

  trex = createSprite(50,height-70,20,50);


  trex.addAnimation("running", trex_running);
  trex.addAnimation("collided", trex_collided);
  trex.setCollider('circle',0,0,350)
  trex.scale = 0.08
  // trex.debug=true

  invisibleGround = createSprite(width/2,height-10,width,125);
  invisibleGround.shapeColor = "#f4cbaa";
```

Output:

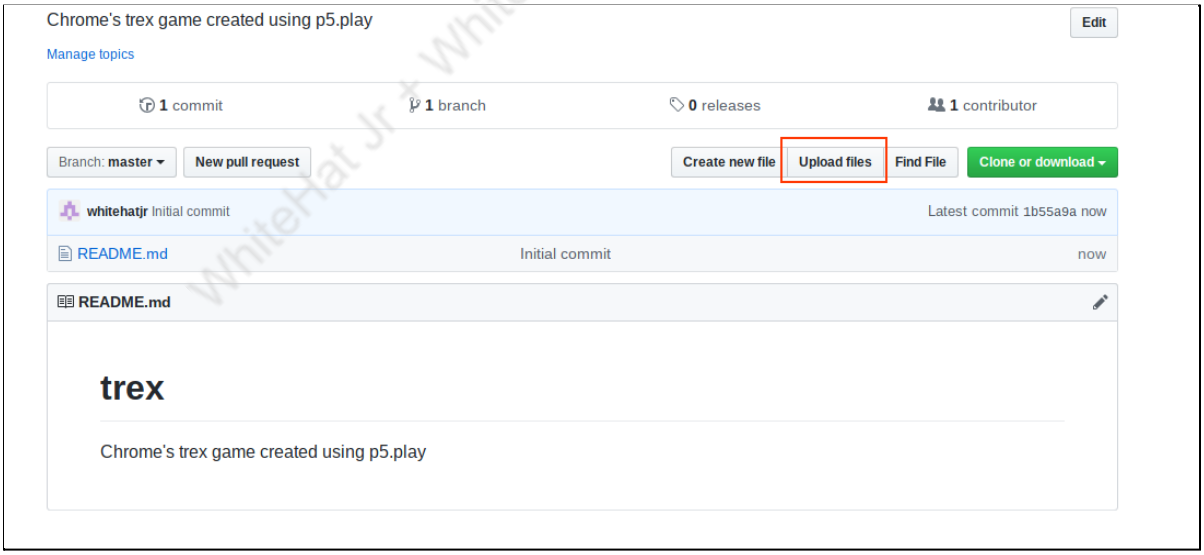
2. Check if there is a touch or a tap as the phone doesn't have a spacebar key to make the Trex jump:

```
if (gameState===PLAY){
  score = score + Math.round(getFrameRate()/60);
  ground.velocityX = -(6 + 3*score/100);

  if((touches.length > 0 || keyDown("SPACE")) && trex.y  >= height-120) {
    jumpSound.play( )
    trex.velocityY = -10;
    touches = [];
  }

  trex.velocityY = trex.velocityY + 0.8

  if (ground.x < 0){
    ground.x = ground.width/2;
  }

  trex.collide(invisibleGround);
  spawnClouds();
  spawnObstacles();
```

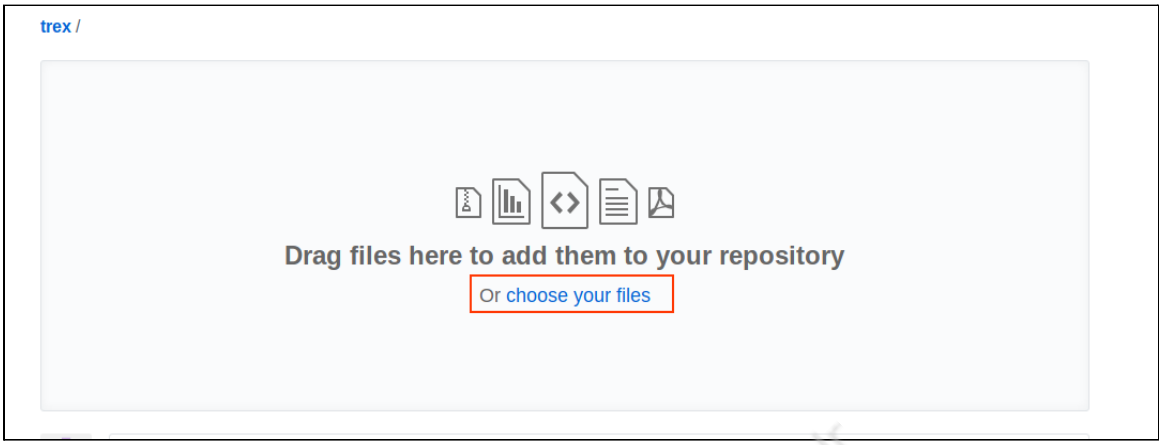
3. Host game on GitHub:
    - Create a new repository:



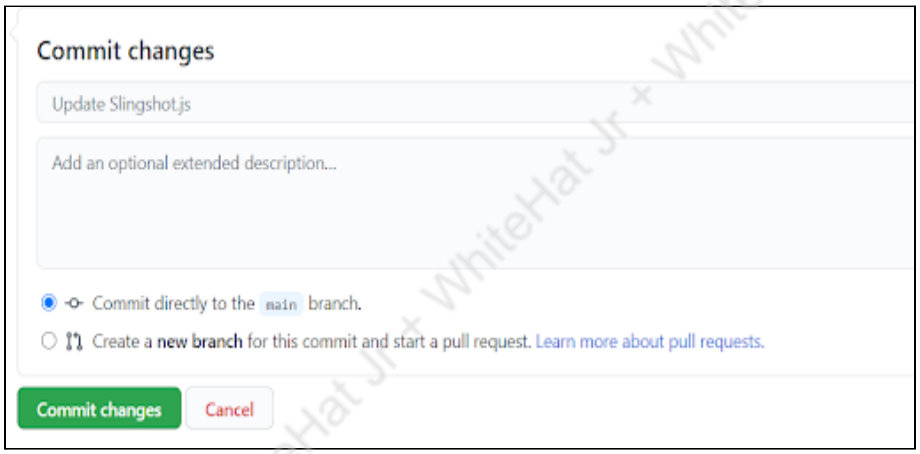
- Select **Upload files** to add the files:



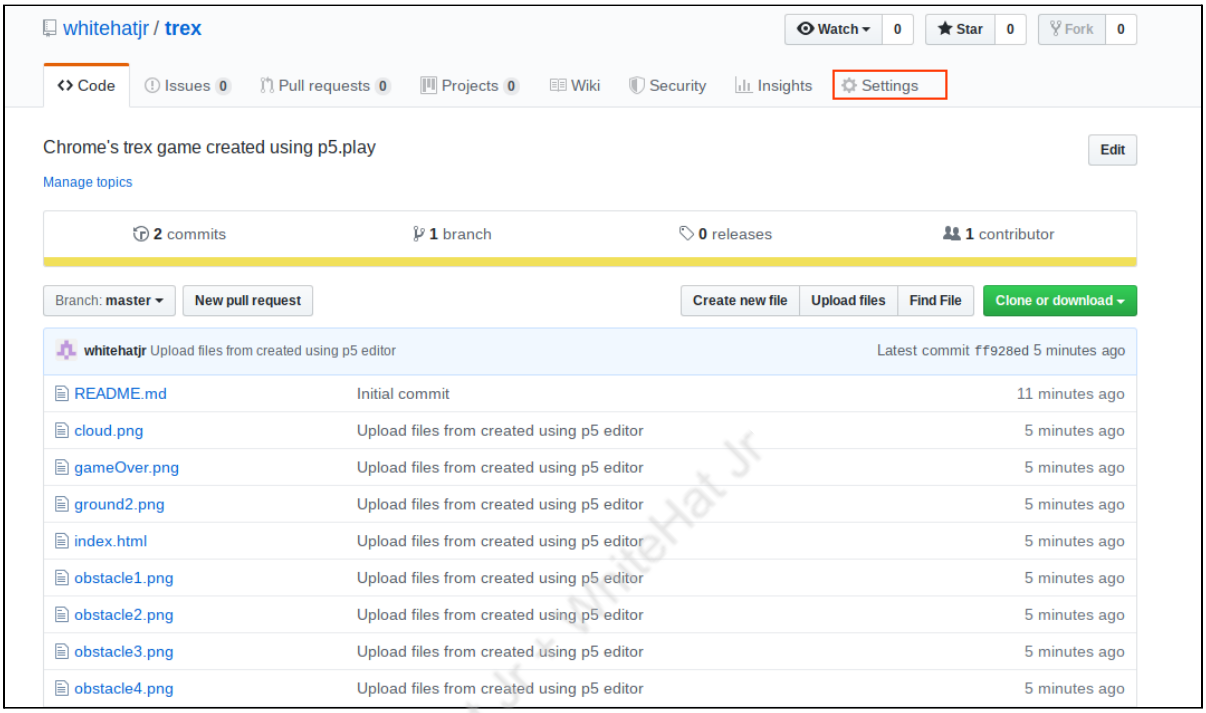
- Click on Choose your file to select the Trex project files on GitHub:



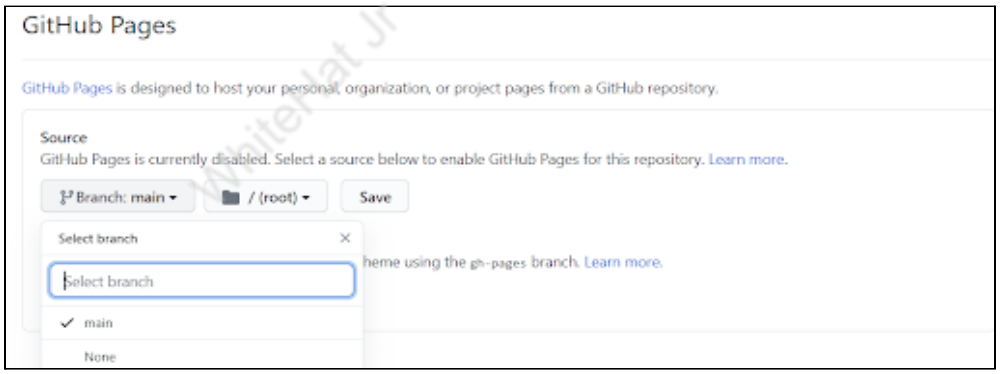
- Click on **Commit changes** once the files are uploaded:



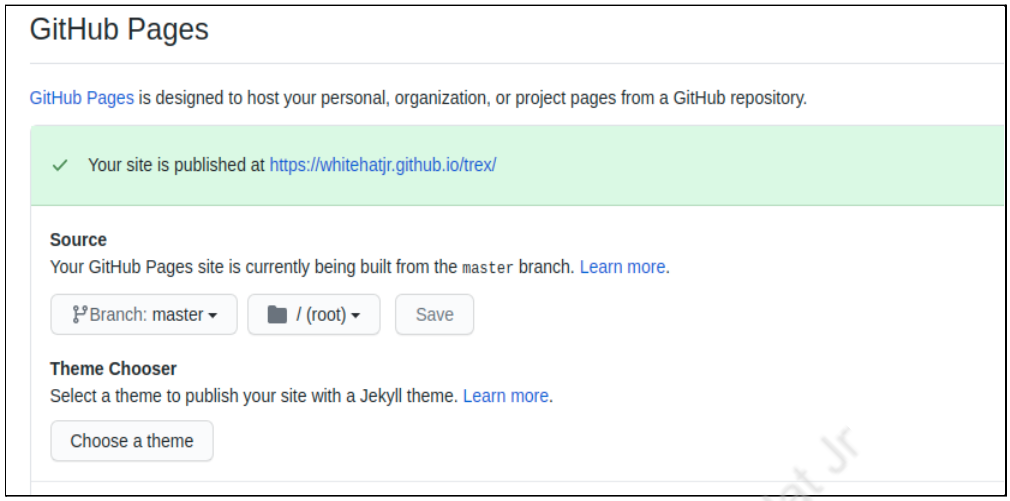
- Go to the **Settings** of the repository:



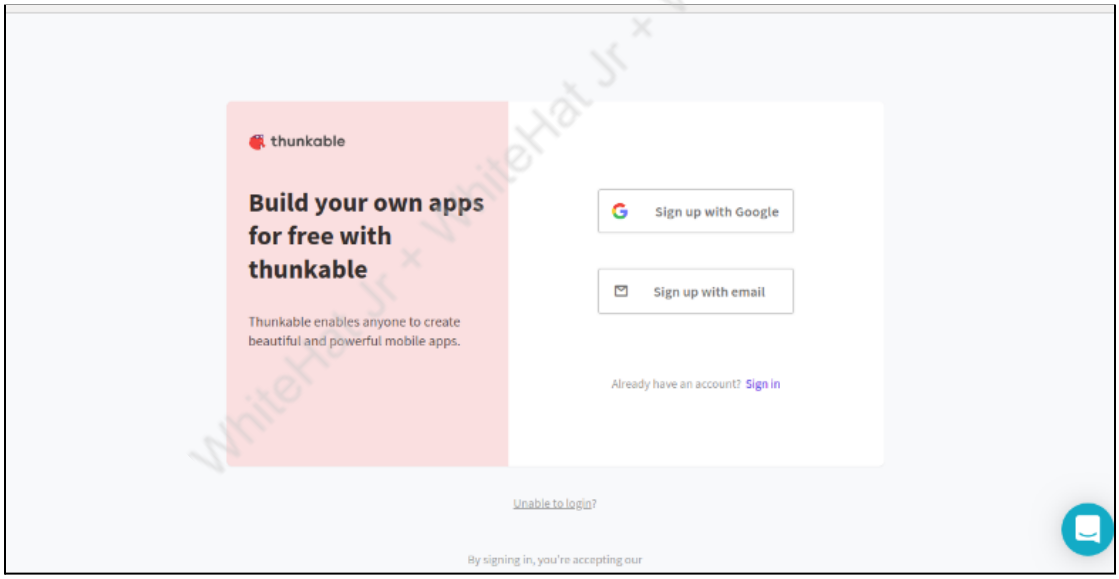
- When activating GitHub pages make sure that the **Branch** is the **main** branch that contains your files:



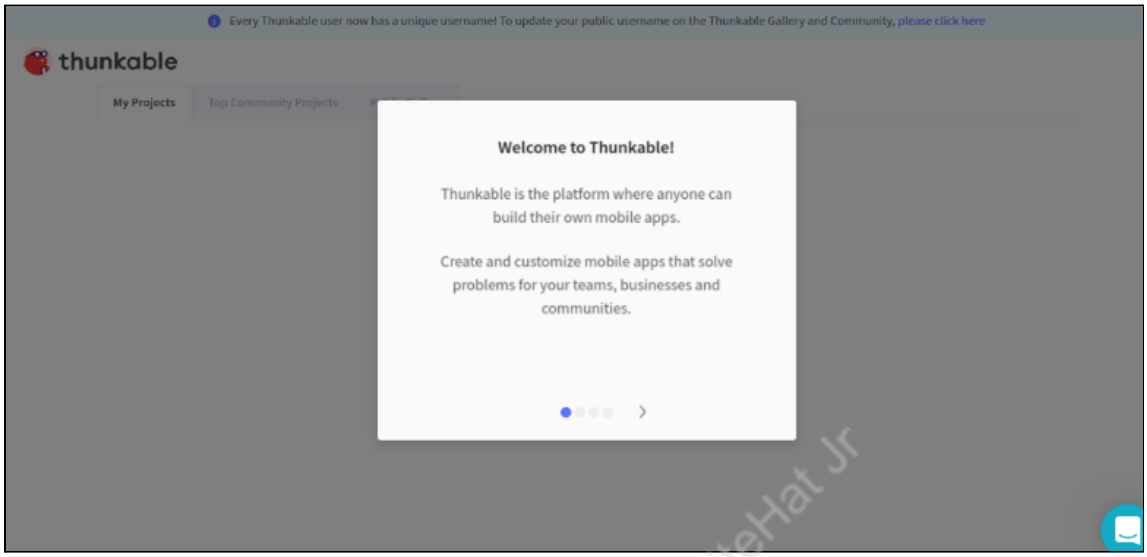
- Click on the URL to view the published page:



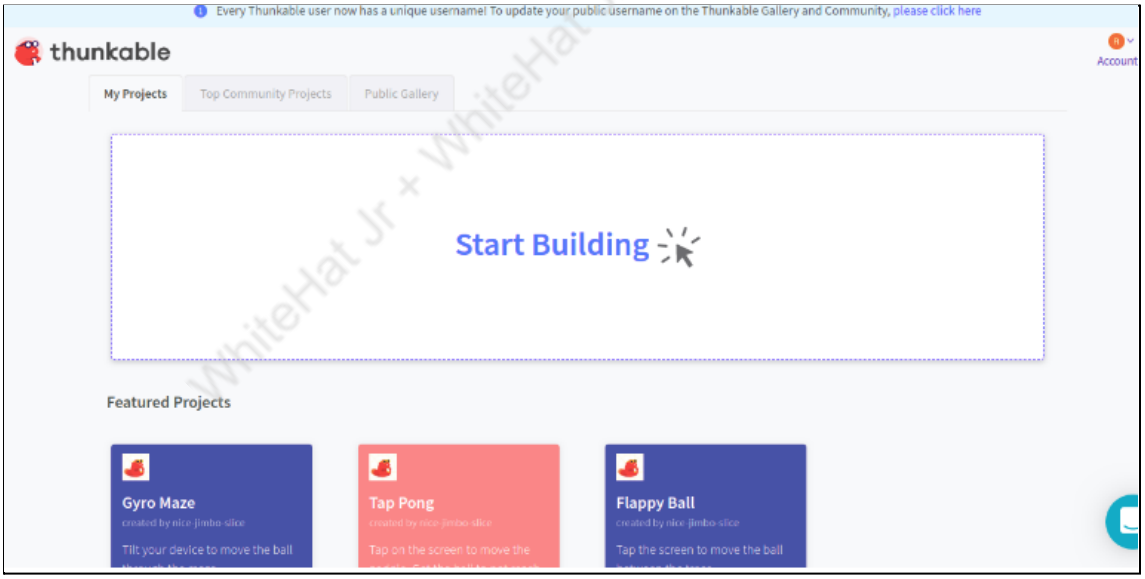
4. Publish the game using **Thunkable**:



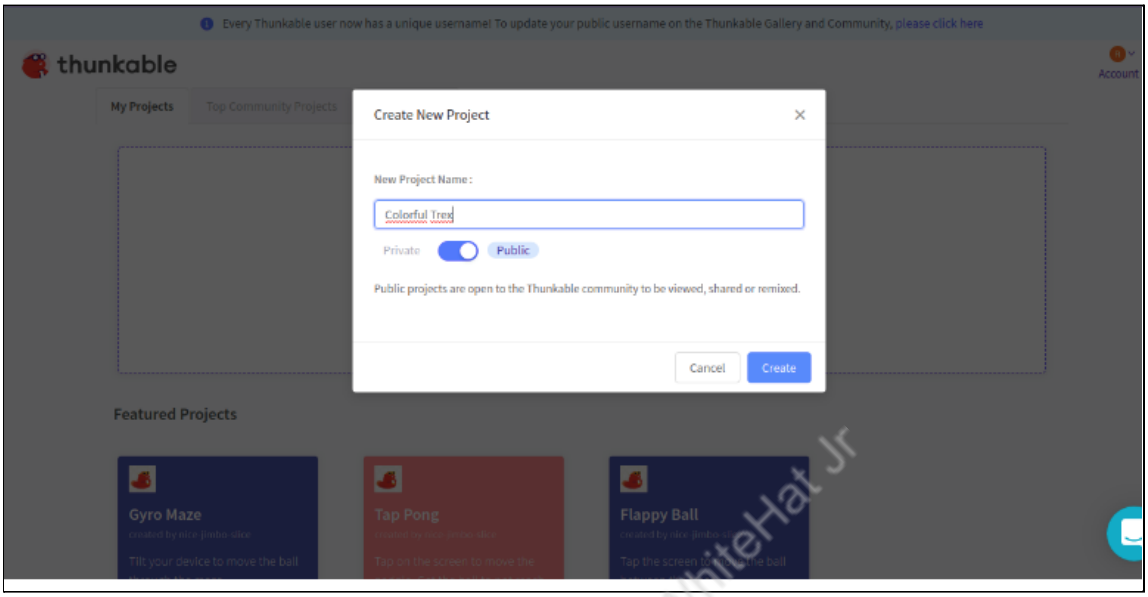
- Sign in to **thunkable.com**



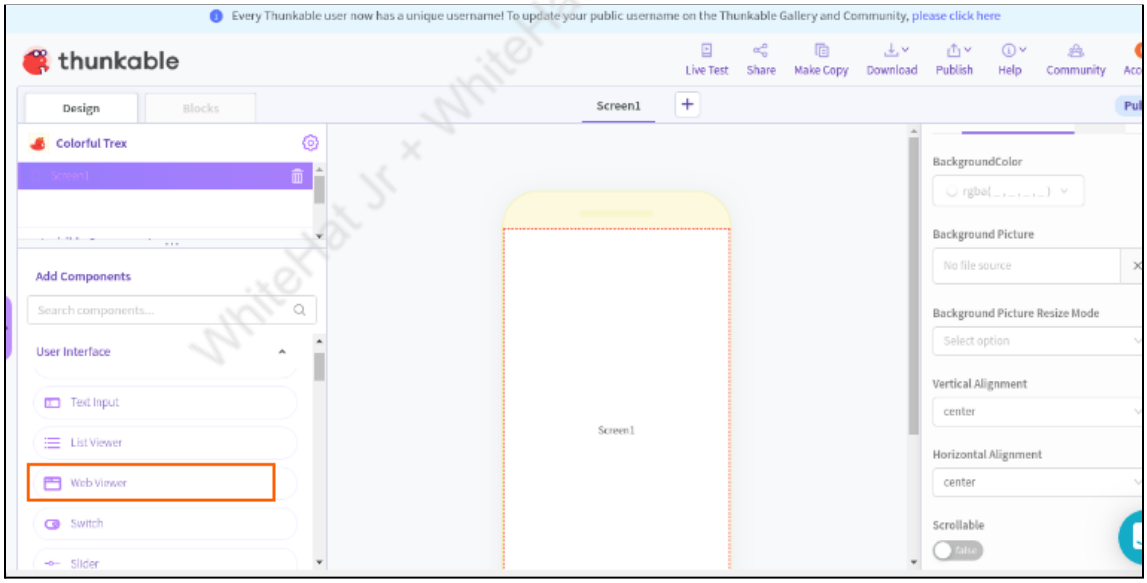
- Start building the project in **Thunkable**

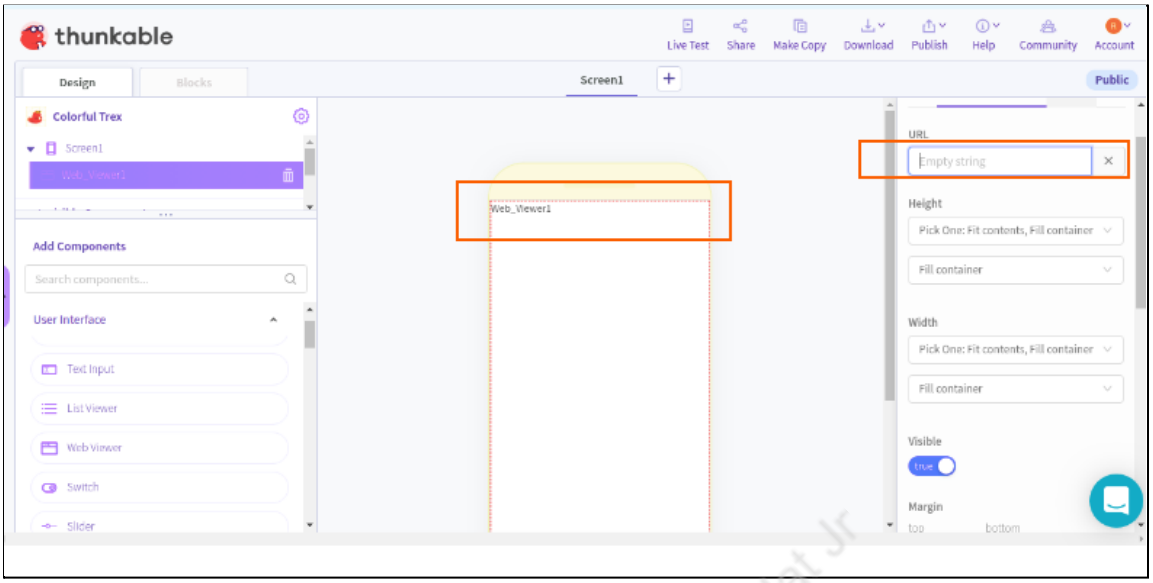


- Add the new project by giving the name.

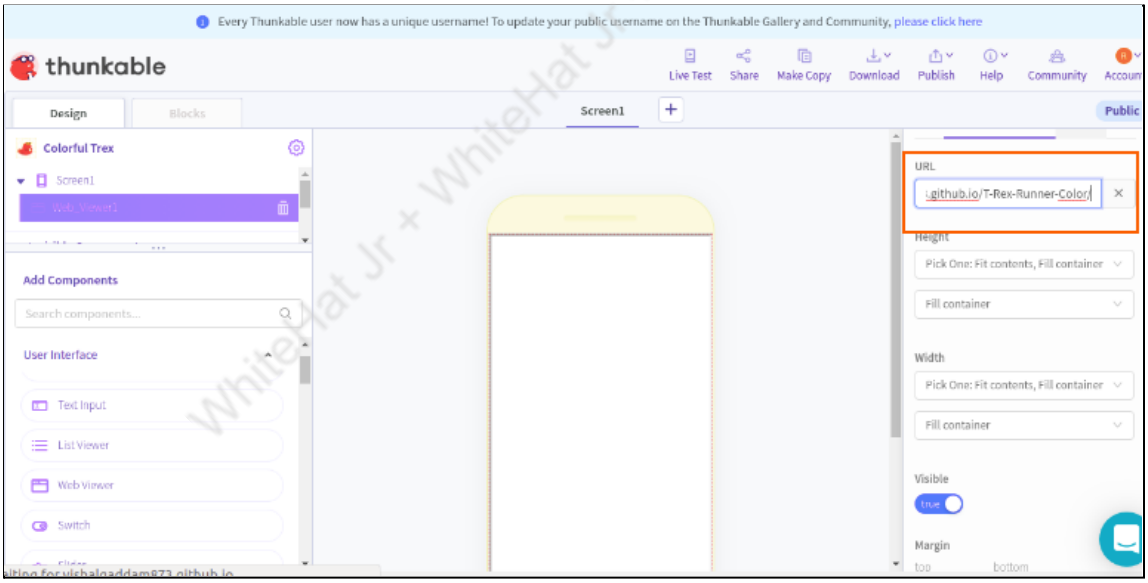


5. Locate and add the **Web Viewer** component to the screen:

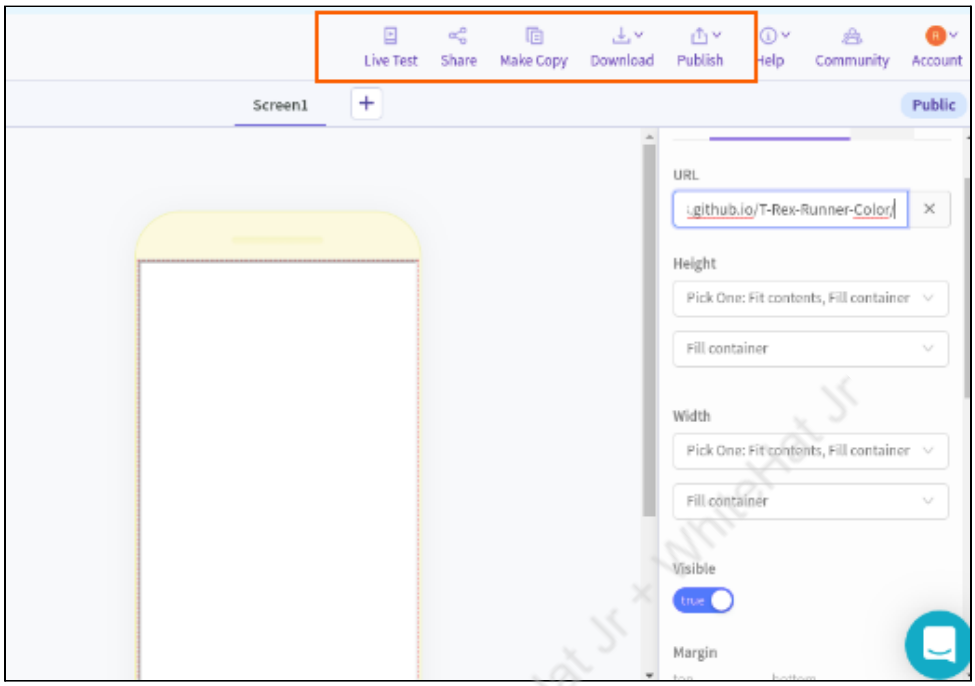




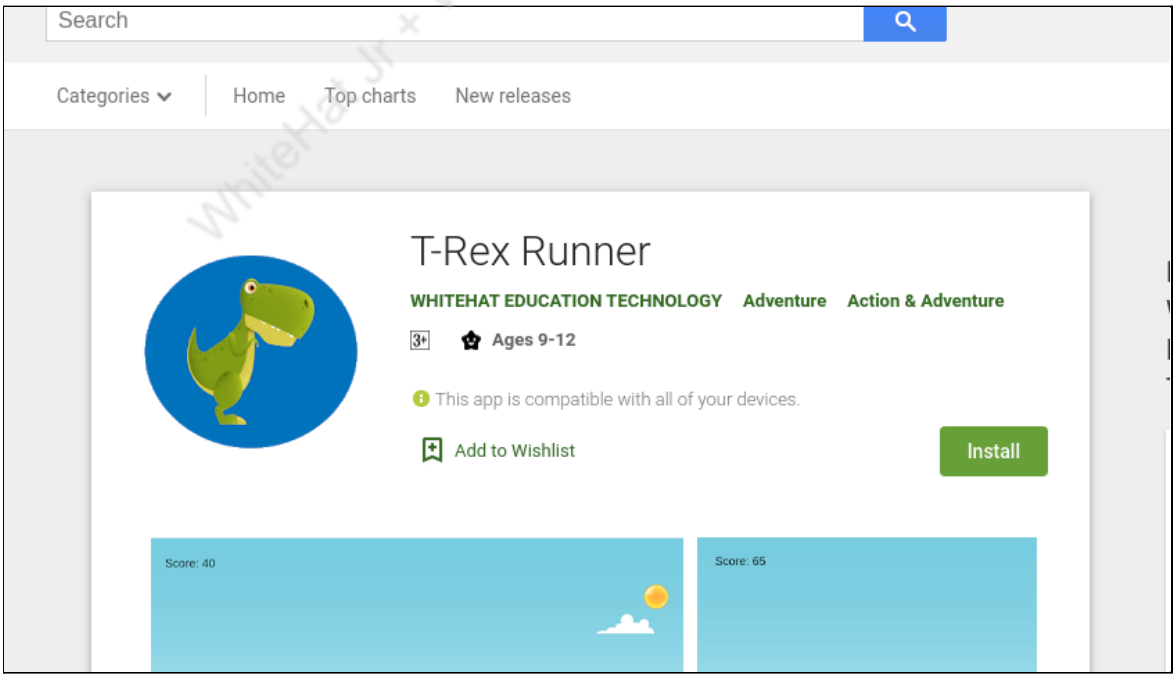
6. Paste the game link which is hosted in GitHub in the URL part:



7. Check the **Live Test** on the app, and also download the APK to be installed on iOS and Android:



8. Download the file from the Play Store:



What's next?

We will learn how to build a different infinite running game - a ghost running on the tower - which you will be coding on your own!

Extend Your Knowledge:

1. Learn & Experiment with more components of Thunkable: https://thunkable.com/#/