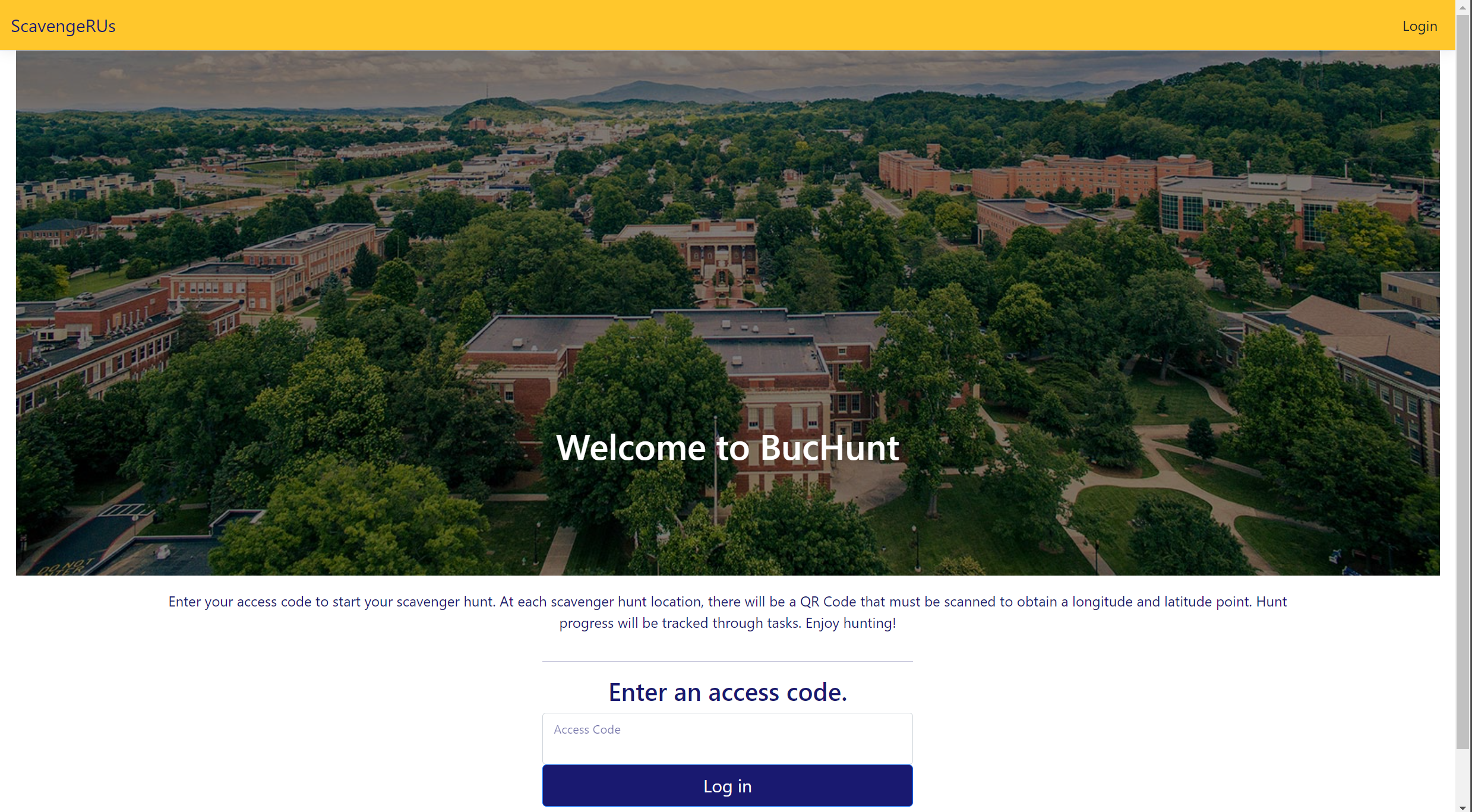
# Adding a User using the Program

## To Start - Background

* This document will show you how you can add a hunt to the database.
* This document will also show you how to add tasks to a hunt.
* This document will show you how to create tasks. Right now, there are only a handful, but you can add more!
* For information on how to add players to the hunt, see the document “Making a User with the Program”.
* This document uses Admin Credentials from Team 1, which are:
  + Email: [waltonca@etsu.edu](mailto:waltonca@etsu.edu)
  + Password: YMXH@9J!72kM6Em

### Steps

1. Run the program. You should be taken to this screen:



1. If you are logged in, logout (unless you are already logged in as an admin… then skip to Step 5). On the top-right of the screen, click “Log Out” to logout.
2. Then click “Login” after you’ve logged out. You should be taken to this screen:

Graphical user interface, application, Teams

Description automatically generated

1. The admin of this project’s credentials are:
   1. Email: [waltonca@etsu.edu](mailto:waltonca@etsu.edu)
   2. Password: YMXH@9J!72kM6Em
2. Press the “Log in” button after you enter the credentials above. You should be taken back to the BucHunt landing page:

Graphical user interface, website

Description automatically generated



1. Click on “Admin Portal” (highlighted in red above). You should be taken to the screen below:

Graphical user interface, table

Description automatically generated

1. Click on “Manage Hunts/Tasks”.

Table

Description automatically generated

1. Click on “Create New Hunt”

Graphical user interface, application

Description automatically generated

1. Fill out the details of the hunt you want to create. When you’re done, your screen should look similar to this:

Graphical user interface, text, application

Description automatically generated

1. Click the “Create Hunt” Button.
2. Congratulations! The Hunt is Active!

Table

Description automatically generated

### Adding Tasks to the Hunt

1. Now that we’ve made our hunt, we must add tasks! At the moment, there is no one participating in the hunt and no tasks.
2. Click on “View Hunt” next to the hunt you created.

Table

Description automatically generated



1. You are taken to this screen.

Graphical user interface, application, Teams

Description automatically generated

1. As stated, there are no tasks for the hunt. Let’s add some. Click on the “Add Tasks” button. You’ll be taken to this webpage:

Graphical user interface, application

Description automatically generated

1. Note the “Create new task” button. We’ll use it later. For now, just click on “Add” for any task. Notice that when you click on “Add”, it changes the link to “Remove”.

Graphical user interface, text, application, email

Description automatically generated

1. Exit the Tasks menu and go back to Hunt2. The hunt has updated!

Graphical user interface, text, application, email

Description automatically generated

### Creating a Task

1. There are only a handful of places listed here. There’s a ton more places around ETSU (at the time of writing, Lamb Hall is near completion)! We’ll need to create new tasks to keep up with the evolving campus.
2. Navigate to any hunt. This example will use Hunt2:

Graphical user interface, application

Description automatically generated

1. Click on “Manage Tasks”.

Graphical user interface, text, application, email

Description automatically generated



1. Click on “Create new task”. You should see this screen here:

Graphical user interface, text, application

Description automatically generated

1. Be in mind that the picture of the place is also included. Be sure to have the picture of the place ready when you’re creating the task.
2. You will also need the Latitude and Longitude of the place. I used a GPS latitude/longitude finder, like <https://www.gps-coordinates.net/>.
   1. This website will ask you for your location, so hit “yes”. Then, it will find the current latitude/longitude of your current location.
   2. Be sure to be in the specified hall before you hit “yes”. You don’t know where this geolocation data could be stored…
   3. Or, just google “latitude and longitude finder”. There are many other options available.
3. This example will create Davis Hall. It is where the documenter (the person who is writing this document) lives on campus.

Graphical user interface, application, email

Description automatically generated

1. When you have everything in place, press the “Create” button.
2. Congratulations! The Task was added successfully!

Graphical user interface

Description automatically generated

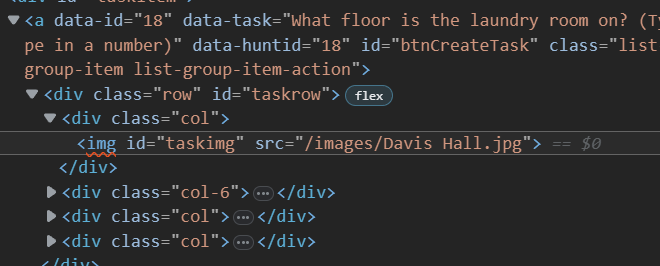


1. Add the task and go back to the Hunt. Oh… Looks like something’s wrong…

Graphical user interface, website

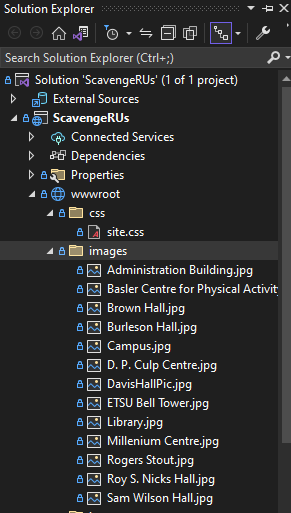
Description automatically generated

1. Pull up Inspect (Right-Click the picture). Note that the src for the image is /images/Davis Hall.jpg.
   1. We need the name of our picture to be exactly Davis Hall.jpg.



* 1. According to src, the webpage is looking in the images folder for a file named Davis Hall.jpg.
  2. Yours may look different. Be sure to inspect the webpage and find the exact name of the file it’s looking for from ‘src’.

1. Also, the image doesn’t display because we didn’t add the image to the code. We can do so by going to \wwwroot\images.
2. Copy and paste the image so it is in the images folder.





1. Then, rename the image to the file based on what you found from using inspect (In my case, I renamed it to “Davis Hall.jpg”).

A screenshot of a computer

Description automatically generated with medium confidence



1. Refresh the webpage. The picture should load up.

Graphical user interface, website

Description automatically generated

1. Lastly, open the ScavengeRUS database. Navigate to the data under the Location Table:

Graphical user interface, application

Description automatically generated

1. Type in “Not completed” for the place of the Task you’ve made (in my case, Davis Hall)

Graphical user interface, text, application

Description automatically generated



1. Hurray! You’re ready to make more tasks!