

NORTHWEST NAZARENE UNIVERSITY

NNU Space Research Website Portal and Database

THESIS

Submitted to the Department of Mathematics and Computer Science
in partial fulfillment of the requirements
for the degree of
BACHELOR OF SCIENCE


Jordan Poundstone
2017

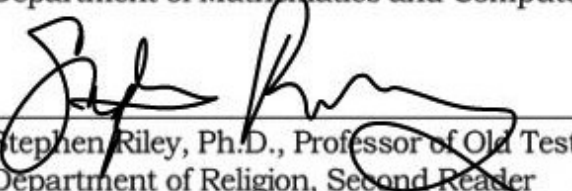
THESIS
Submitted to the Department of Mathematics and Computer Science
in partial fulfillment of the requirements
for the degree of
BACHELOR OF SCIENCE


Jordan Poundstone
2017

NNU Space Research Website Portal and Database

Author: 
Jordan Poundstone

Approved: 
Barry Myers, Ph.D., Professor of Computer Science,
Department of Mathematics and Computer Science, Faculty Advisor

Approved: 
Stephen Riley, Ph.D., Professor of Old Testament Studies,
Department of Religion, Second Reader

Approved: 
Barry Myers, Ph.D., Chair,
Department of Mathematics and Computer Science

Abstract

NNU Space Research Website Portal and Database

POUNDSTONE, JORDAN (Department of Mathematics and Computer Science),
MYERS, DR. BARRY (Department of Mathematics and Computer Science).

NNU currently has two satellite projects: MakerSAT is the first satellite developed in Idaho and RFT-SAT is the second. While work on RFT-SAT was underway, development began on a website to host information about NNU's space research projects and display data sent from the satellites. Information about the state of the projects, related news, contact information, and an image gallery can be accessed on the website without logging in. By logging in, users can gain access to the database to view data sent back to earth from the satellites. The database contains a table with data identifying each of the projects. Connected to this are various tables holding data sent back from the projects. The satellites will send back beacon packets with their locations and diagnostic data and data packets with anything collected during the science experiments. These packets will have different types of data so they require different tables to hold them and different display layouts on the website. The project consists of a website designed using HTML, CSS, JavaScript, and PHP; a database developed with MySQL; and a python application to parse data and upload it to the database.

Acknowledgements

I would like to thank my family for supporting me throughout my education and the sacrifices my parents made to ensure that I got the best education possible. I would also like to thank my wife Janeé for her love and support throughout this project. Without her I'm not sure I would have made it to graduation. Next, I would like to thank the RFT-Sat team: Dr. Dan Lawrence, Dr. Joshua Griffin, Dr. Steve Parke, Cassie Wade, Daniel Slemmer, Lucas Schamber, and Curtis Garner. We had an amazing project and this team was a pleasure to work with and made it a great experience. I would like to thank Dr. Barry Myers for his support before and during the development of this project and Dr. Stephen Riley for being my second reader. Finally, I would like to thank NASA, without their CubeSat Launch Initiative this project would have never existed.

Table of Contents

Title Page.....	i
Signature Page	ii
Abstract	iii
Acknowledgements	iv
Table of Contents	v
List of Figures	viii
Introduction	1
Background	1
Project Overview.....	3
Body.....	4
Database Requirements	4
Website Requirements.....	5
Development Environment	6
Website Implementation.....	7
Database Implementation	14
Future Work.....	16
Conclusion.....	18
Works Cited.....	19
Appendix A – Other Resources.....	20
Appendix B – Screenshots	21

Appendix C – Code	32
Head	32
Navigation Bar	36
Modal Login Form	38
Footer.....	41
Custom Search Engine.....	43
index.php.....	45
data.php	47
satellites.php.....	59
gallery.php	62
contact.php.....	64
contact_rft.php.....	66
whereami.php	72
settings.php	81
Error Handling for Settings	92
login.php.....	99
logout.php.....	102
style.css	103
registration.php.....	124
adduser.php.....	127
deny.php.....	134

changeusername.php.....	136
change pass.php	139
changeemail.php.....	142

List of Figures

Figure 1 – Embedly Configuration	8
Figure 2 – Embedded Twitter Configuration.....	8
Figure 3 – Contact Dropdown Menu	10
Figure 4 – Custom Map Creation	11
Figure 5 – Modal Display of Search Results	12
Figure 6 – Login Table	14
Figure 7 – Registration Table	14
Figure 8 – rftsat_data_packet Table	15
Figure 9 – rftsat_radiation_counter Table	15

Introduction

This project began as the design and production of a CubeSat funded by NASA before turning into the development of a website and database for NNU's Space Research. This section will discuss the background behind the CubeSat and what its purpose is. There is also a short overview of the completed project, what was developed, and its purpose.

Background

In April 2016, NNU received a grant to build a CubeSat from NASA as part of the CubeSat Launch Initiative (CSLI). NASA selected 30 schools to receive CSLI grants of \$200,000 to build a CubeSat or a high-altitude balloon. The schools were selected based on proposals for various science experiments. The projects had to be something useful in the real world and had to be presented in a way to show NASA that the project was worth the time and the money about to be spent on it.

NNU proposed a CubeSat following the 3U form-factor, which gives a volume of 10cm x 10cm x 30cm. This project, known as Radio Frequency Tag Satellite (RFT-Sat) is being developed to test a Radio Frequency (RF) technology in orbit around the earth. NNU proposed to NASA that small passive sensor boards (also known as tags) equipped with energy harvesting and backscatter capabilities could be placed in locations which could not have wires run from a power system. When a reader sends a signal to the tags, the antenna would deliver the power from the radio frequency to the sensors to power them and collect data. After data was collected, the tag would modulate the impedance of the antenna and bounce the signal back to the reader without needing a power system or active antenna to send data.

RFT-Sat will be launched in Summer of 2018 and will be in orbit around the earth at roughly the altitude of the International Space Station (about 400 km) for three to six months. At the end of this period, RFT-Sat will fall back into the earth's atmosphere where it will disintegrate. During its lifespan, the satellite will extend a boom holding a sensor tag which will collect data on space weather. The boom will be nonconducting so that it will not interfere with the radio signal sent between the reader and tag. While this data is not a new discovery, it will demonstrate that the technology is a viable solution in space.

Once this technology has been verified, it will be possible to attach sensors to the outside of space structures such as the Hubble Telescope where it is difficult or expensive to run wires to the outside. These sensors could measure a variety of things including structure health and degradation. Once the sensors are in place, another satellite equipped with a reader could send a signal to the tags as it passes by the telescope or any other satellites equipped with the technology and take a reading. The structure and electrical wiring of the satellite would not have to be changed at all but new sensors could be added.

During its lifecycle, RFT-Sat will collect, packetize, and transmit space weather data back to earth. These packets will be stored as raw data strings in a database managed by Near Space Launch (NSL). The only way to access this data will be to log in to NSL's server using the account provided to NNU. While this could be sufficient for one user, it is not a good solution for many users to access the data or to store data after it has been parsed and processed by NNU.

Project Overview

Once data has been sent back to earth, it will need a home. The purpose behind this project is to house data sent back to earth and develop a portal for individuals interested in NNU Space Research. It started out as a simple database for storing data returned by RFT-Sat and creating an interface to view this data, but the project turned into a database ready to add multiple satellites into it and a website capable of:

- Letting users view data from any satellite with data in the database
- Information about NNU Space Research for individuals interested but not already involved
- Maps showing the satellites current locations and how far each has travelled
- Contact information for the department and individual team members
- A registration/login system

Body

During the development of this project, requirements frequently changed. In this section, the initial requirements will be discussed, followed by the process of implementing and testing everything. After the discussion of what has been done, there will be a list of items not yet completed. The body will end with some concluding remarks before the Appendices. Screenshots of each webpage discussed in this section can be found in Appendix A.

Database Requirements

The database requirements started out simple. It was required to hold three tables, each storing a different type of packet being sent: Health Beacons, Radiation Counter, and Science Data. It was originally planned to only have RFT-Sat data stored in the database, but during development, the design changed to need a database capable of holding data from any satellite or rocket project at NNU. More tables were then needed in order to house the packet types unique to each project and information about each project. The design of the website also required the database to hold tables for the registration and login features of the website.

To hold the packets that are being sent back from RFT-Sat, the database needed to hold a date and time of when the packet was received, the raw data string, a packet type code, and all fields relevant to that packet type. We have sent two types of packets through the radio on the ground to test the data connection: health beacon data and radiation counter data. The health beacon packets are split up into six analog data fields, the bus voltage, internal temperature in degrees Celsius, a packet counter, P1DI, voltage of the six batteries, and the voltage and current of three solar panels. The radiation counter packets are much simpler holding its metadata and a single value corresponding to the amount of radiation measured by the sensor board.

Website Requirements

At the beginning of the development of RFT-Sat, the plan was to have a database storing all data returned and an iOS app to view the data. After reviewing the costs associated with developing an iOS app and the lack of access to Apple devices for several members, a website sounded like the best solution to view data. By developing a website, members can access the database from any device rather than needing an iPad or iPhone.

Once the plan changed to implementing the database connection on a website rather than an iOS app, the requirements expanded as well with the options. Developing a website created the ability for anyone to access the site, not just the members of the team. A login system became necessary to ensure that a user could only access the database if they were given permission. Other features were requested for users who did not have permission to access the sensitive data. Contact information for the team and NNU departments were added in case perspective students view the site. For anyone interested in the location of the satellites, a map was added with a marker that automatically updates based on the last known geographic coordinates of the satellite. An odometer was also added to show how far the satellites have travelled since their launch. An image gallery was created to display images of the teams, development of the projects, and any photographs taken in space and transmitted back to the ground.

Development Environment

This project was developed using several different technologies. The code was written in Sublime Text 3, which is a light IDE for Windows, Mac, and Linux. It supports syntax highlighting and correction for many languages including the languages used for this project: HTML, CSS, JavaScript, and PHP. HTML and CSS were used to create the base of the website, JavaScript was used to implement the features like the map and odometer, and PHP was used to communicate between the website and the database. For the development of the website before database access was implemented, code was tested in Google Chrome. Later, a server was needed to test the PHP and database.

The database was initially created using MySQL Workbench, but once implemented into the website was redone in phpMyAdmin. Originally, MySQL, Microsoft Access, and PostgreSQL were considered. Microsoft Access did not offer the features that the other two had, such as multiple users connecting to the database at once, so it was not used. Between the other two standards, PostgreSQL had far more powerful features, but it was also more complicated to develop the database in. Its biggest claim over MySQL is the superior speed during highly complex transactions. Given that this project only involved very simple tables and queries, MySQL was selected for its more simplistic approach to database development.

Once the database was being implemented, a web browser was no longer sufficient to test the code. To create the code to communicate with the database, WAMP or Windows, Apache, MySQL, PHP was used. This program works as a local web server running Apache with support for MySQL and PHP inside a Windows environment. This is a great tool for developing and testing a website on a local machine, however it is not a full webserver meant to host a deployed website.

Website Implementation

The first thing implemented was the basic website. A frame was created, with a navigation bar at the top and five pages: Home, Data, Satellites, Gallery, and Contact. These were programmed using HTML5 and a CSS file based on the NNU homepage. At the bottom of each page, there is also a footer containing helpful links to pages on the Space Research portal and the NNU website. During the development of the project, as requirements evolved, a page titled “Where Am I?” was added to the navigation bar along with a search box, settings page, and a modal login form.

The navigation bar is set to stay at the top of the browser wherever the user is at on any page to make moving to other pages easier for which the code can be seen in Appendix C under the navigation bar heading. It is a very simple implementation of a navigation bar with a list of links in it. One requirement of the site was to have a contact page for each team rather than having to scroll through searching for the team member that the user is wants to find. This was the most difficult button to implement but in the end is a button which when hovered over changes the CSS of a short list of links to the different contact pages.

When reaching the website, the first page seen will be the Home page, shown under index.php in Appendix C. The left side of the page contains various news articles featuring NNU Space Research. This feature is powered by a site called embed.ly, which gives the ability to embed a variety of media in your website. It is very simple to use, requiring just a link to the content and some quick configuration. Embedly generates code which can be copied and pasted into a website’s code. On the right side of the page is a live Twitter feed. Currently the feed pulls Tweets from NASA’s official Twitter page and automatically displays the newest content. Later, if NNU Space Research creates a Twitter account or another Twitter account’s Tweets

are more desirable, the feed can be altered with just a single line of code. This part of the page is powered by Twitter’s API which can be found on their website as shown in Figure 2.

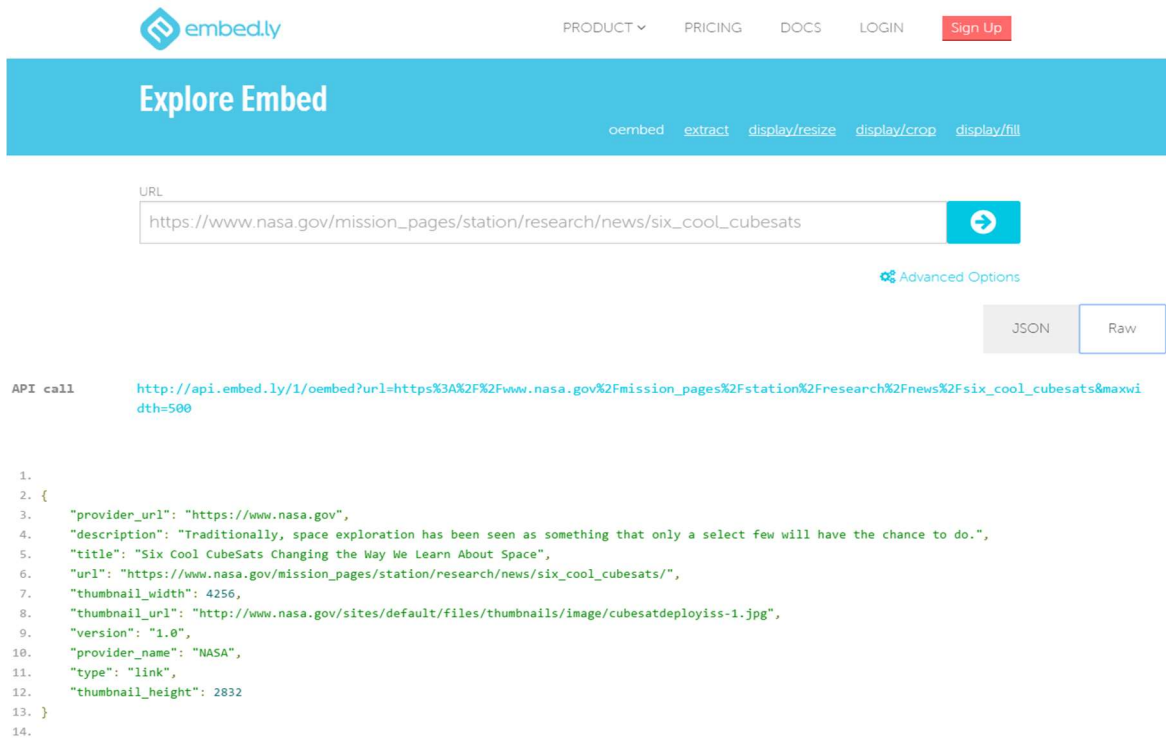


Figure 1 – Embed.ly Configuration

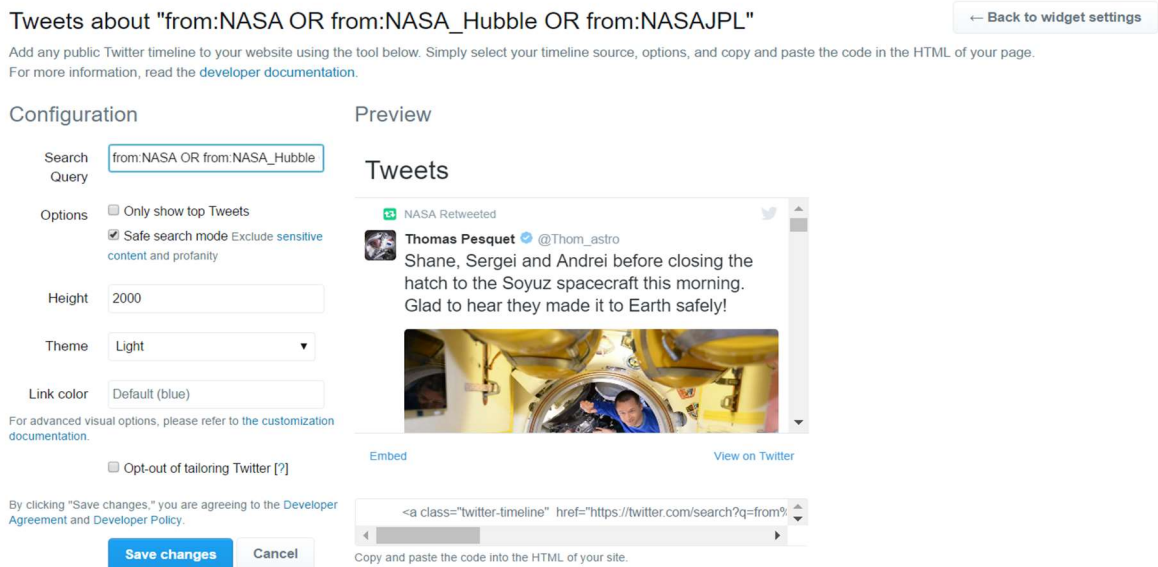


Figure 2 – Embedded Twitter Feed Configuration

The next page is the Data page, which grants access to the database. Its code can be viewed in Appendix C under the name data.php. This page is only accessible if the user is logged in. If the user has not logged in, an error message will pop up blocking all input on the screen except for links to go to another page and the login button. This prevents users from seeing any data sent back from the satellites unless they have explicitly been given permission. Once logged in, the user is presented with a single query asking what satellite they would like to access data from. After selecting a satellite, the appropriate list of packet types will appear. The final option after this is which data fields to display within a packet type. Pressing the submit button will display a table of the requested data.

After this is the Satellites page, which contains information about NNU's current and past Space Research projects. This was the simplest page to create as its content is written entirely in HTML and CSS. The code for this page is in Appendix C in satellites.php. The content of the page is short descriptions of each satellite project that NNU is currently working on followed by descriptions of past projects.

The next page will contain images taken during the development of Space Research projects as well as photographs returned by satellites and is creatively titled Gallery. The image gallery contains 4 columns of images, each of which can be clicked on to open a full resolution copy of the image. The content of this page was again made without using any JavaScript or PHP but was much more complicated to create than the Satellites page.

Hovering over the Contact button on the navigation bar reveals three options: NNU Science Departments, RFT-Sat, and Maker-Sat I as shown in Figure 3. Choosing one of these options brings up a new page with the team members of the selected project (or department). These pages are under contact.php and contact_rft.php in

Appendix C, Maker-Sat I is a placeholder and does not yet have a contact page. The pages have slightly different content based on whether the user is logged in. If the user is not logged in, it will display the names of the team members, their school email address, and the professors contact information which can also be found on the department pages of the NNU website. Once logged in, the page will also display phone numbers for each of the team members. The portal will be used by internal members who may need to contact one another quickly, so having this information in one place would be helpful in the case that someone lost their teammates contact information.

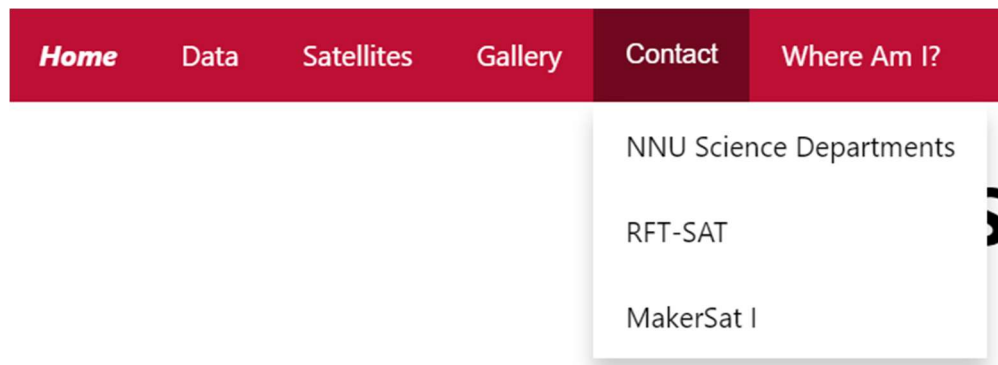


Figure 3 – Contact Dropdown Menu

The final page in the main group on the navigation bar was not in the initial requirements for the website, but was requested by the professors working on the project. Where Am I? holds a map and an odometer. Its code can be found under whereami.php in Appendix C. The map uses a tool from mapsdirections.info, as shown in Figure 4, which generates code using the Google API to embed an iframe into the website that will display a Google Map. The code was modified using PHP script to retrieve the last known geographic coordinates and insert the location into the code for the iframe, which will then place a marker at the location stored on the server. Above the map is also a dial powered by JavaScript which uses an equation based on the speed that the satellite must travel to maintain its orbit and the time since its launch

to gain an estimate of how far it has travelled. It can display the distance in kilometers or miles, chosen by clicking the dropdown menu just after the odometer reading.

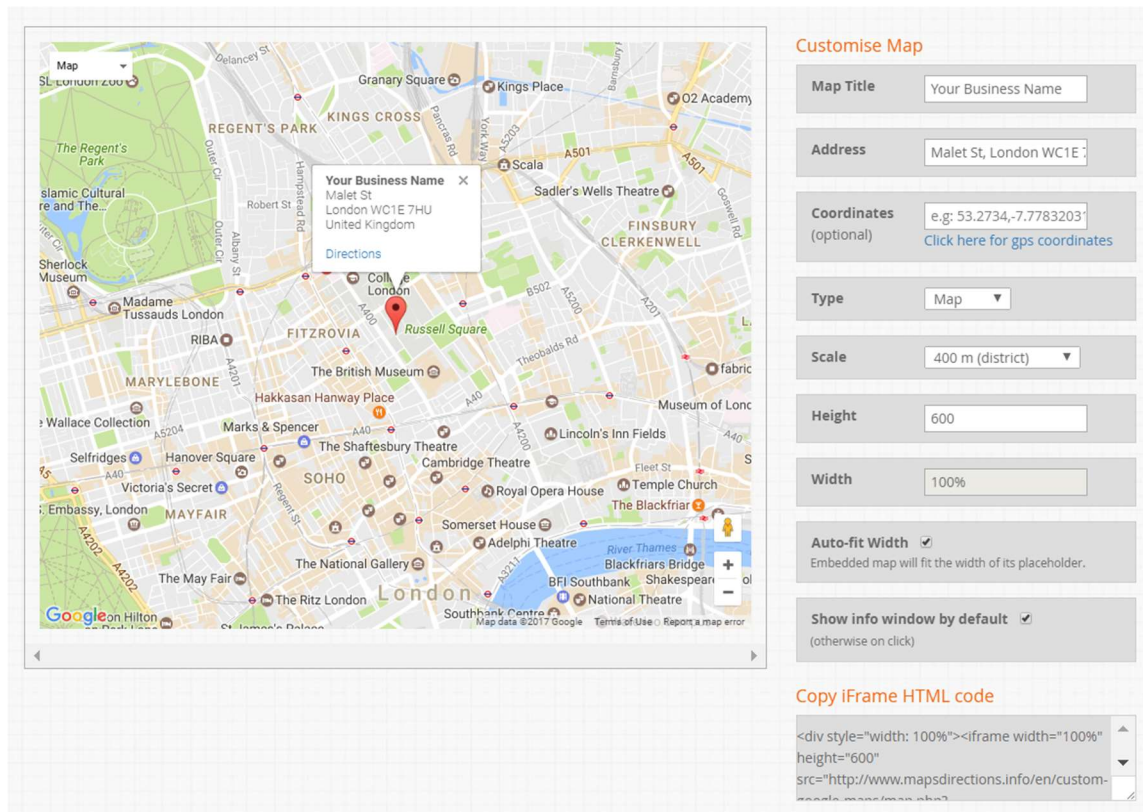


Figure 4 – Custom Map Creation

On the right side of the navigation bar is three more options. Clicking on the magnifying glass icon will open a search box. The code for will also be under navigation bar in Appendix C. This search box is powered by a Google Custom Search Engine (CSE). A CSE can be created on Google’s website by entering criteria for what websites, pages, domains, or tags are relevant. Google then generates code which can be copy and pasted into a website. The CSE searches the NNU website and opens the results in a modal box as shown below in Figure 5. The CSE is a powerful tool when users need to be able to search a website for specific information. The free version was used for this implementation, but there is also a paid version which removes the

Google branding and offers some additional features. The search icon changes the CSS of the CSE box to toggle its visibility.

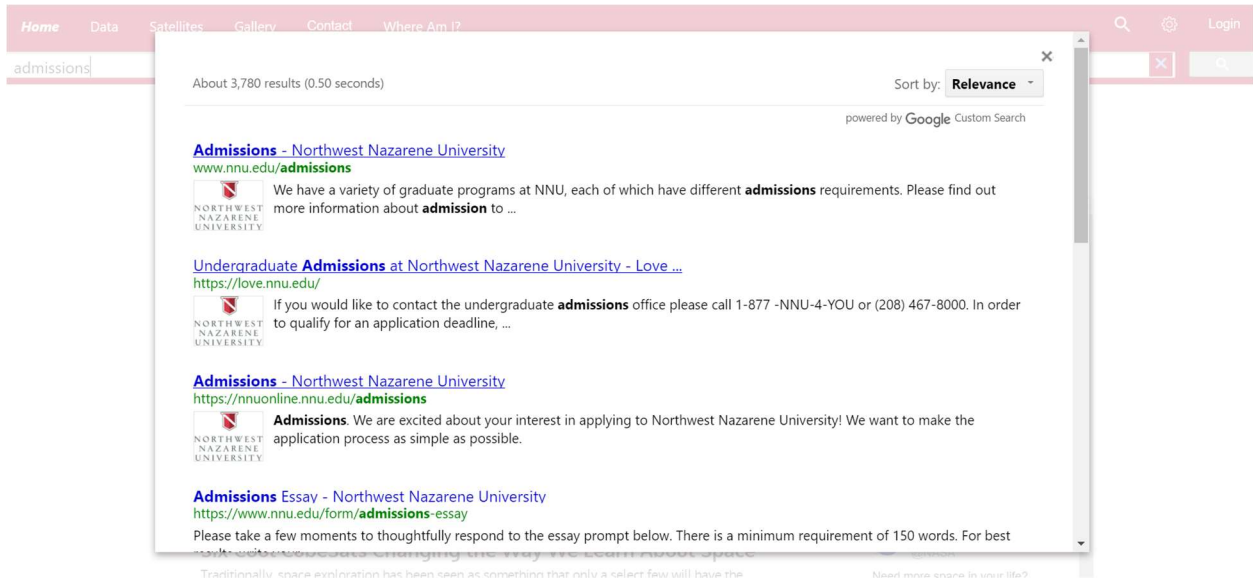


Figure 5 – Modal Display of Search Results

On the right side of the navigation bar, the settings gear is the only button that redirects the user to a new page. By clicking the gear, the user is taken to the Settings page, as shown under settings.php in Appendix C. This is another page which shows different content based on whether the user is logged in. Without being logged in, the user will see a registration form where they can apply for access to the database. By logging in, the page will change and the user will see options to change their password, username, and email. Additionally, if the user is flagged as an administrator in the database, two more sections will appear. The first allows the administrator to add a new user manually by entering an email, first name, and last name. By manually entering the new user, the administrator can also give the new user administrative privileges. Below this option is a table holding a list of any users which have applied for access. The administrator may click deny, which will remove the user from the table and denying access to the database or they may click approve

which will generate a user account for the person. When using the registration table, a new user cannot be given administrative rights. This can only be done by manually entering the user's information as a safeguard against accidentally giving someone that power.

The last button on the navigation bar login/logout button. The text changes based on whether the user is currently logged into the system. The code for the modal login form can be viewed under the heading login and the logout code can be found under logout.php in Appendix B. Clicking login will bring up a modal login form where the user can enter their username and password or select the "Forgot Password" option to have their password reset. Logging in will automatically refresh the page and close the modal form. If an incorrect username or password is entered, an error message will be displayed. Clicking logout will redirect to a PHP script to destroy the session, which will cause the user to need to log in again to view any sensitive content before automatically redirecting to the home page. Once the session has been destroyed, clicking a page link or using the browser's back buttons will not display any information reserved for registered users.

Database Implementation

In order to house a database accessible from the website, MySQL Workbench was used for the initial design. The database was later migrated into the phpMyAdmin system so that it could be used with the website. The database contains several tables, but none have foreign keys as they are not connected. Each table contains a separate type of data and they are each used on their own.

The login table contains information for each user and can be seen in Figure 6. This information includes username, password, real name, email, and a flag to identify the user as an administrator as discussed above. This table contains only approved users, while users waiting for approval are stored in a separate registration table as seen in Figure 7. When a user applies for an account through the settings page, their information is sent to this table, which contains a name, email, optional message, and a flag for denied users. If the flag is set then they will stop showing up for approval by administrators on the settings page. Any users that have registered will not be able to register again, so this denied flag prevents users from repeatedly applying and being denied.



	User_ID	Username	Password	F_Name	L_Name	Email	Administrator
<input type="checkbox"/> Edit Copy Delete	1	jpoundstone	asdf	Jordan	Poundstone	jpoundstone@nnu.edu	1
<input type="checkbox"/> Edit Copy Delete	2	janeepoundstone	jdk;	Janee	Poundstone	janeepoundstone@nnu.edu	0

Figure 6 – Login Table



	id	first_name	last_name	email	message	denied
<input type="checkbox"/> Edit Copy Delete	2	Janee	Poundstone	janeeharris@nnu.edu	Hello!	0

Figure 7 – Registration Table

There are two tables which contain packets sent back from RFT-Sat, rftsat_data_packet and rftsat_radiation_counter. The rftsat_data_packet table contains health beacon packets as shown in Figure 8 and includes a date and time of reception,

Future Work

Both the website and the database have some more features that should be added in the future. Some of these features are things that cannot be fully implemented until more progress has been made, but others are things that simply were not at the top of the requirements list and were left out during this development cycle to avoid feature creep.

The first thing to be addressed is the hosting of the website. Currently it is located on a WAMP server running on a local machine, but this cannot be viewed by any other devices. The goal is to have the site and database stored on NNU's servers as part of the Engineering and Physics department's website. There is currently some content on the department's website with descriptions of the current Space Research projects, but it will be replaced with this site.

After the site is hosted on a real server, it will need an automatic email service. Currently, when a new user account is created, a password for the account is randomly generated, but the user is not notified automatically. Adding a service through PHP will notify the user that they have been given access and allow them to log in, change any necessary information, and access all parts of the site. This feature is not available through WAMP services so it cannot be tested until the project has been migrated to a real server.

There are some small features that will also need to be added such as adding this website to the context search powered by Google. The CSE can only search websites with domain names so it cannot be implemented until it is migrated to a real server and given a URL. A contact page will need to be added for Maker-Sat I, but for the most part can be copied from the RFT-Sat page with the team member list

replaced. Images in the gallery will also need to be updated to pictures relevant to NNU Space Research. Currently the gallery is composed of images retrieved from Google.

The database will also require some modifications later. First, it will need to have the Maker-Sat I tables created. Given that this project was a part of the RFT-Sat team, the data available was also from RFT-Sat. In the future, it is planned for the website to host information about every Space Research project at NNU, not just RFT-Sat. Originally a python program was planned to automatically retrieve data from NSL and upload it to the Space Research database, but this was not at the top of the requirements list as RFT-Sat is over a year from launch at the time of writing.

Conclusion

This project was very beneficial in learning what it is like to develop a larger application than class projects. I had to learn how to manage my time well which was a big struggle for me during this project. I kept waiting for requirements to be more solidified before getting into high gear of development, which did not leave me as much time as I should have had to finish it. I also learned to adapt and add things in on the go as requirements were evolving and new features were requested. At the same time, there was a point where I needed to learn to stop adding in all the extra features and avoid unchecked feature creep.

This project was a great way to use what I have learned from classes throughout my time at NNU. I had to use experience that I was not the fondest of as well as knowledge that I have enjoyed building up. In the end, even the parts that I was dreading having to do become something I enjoyed and hold a lot of pride over what I accomplished. The project ended with a product that I will keep and show people and that will be used and actually serve a purpose.

Works Cited

Coyier, Chris. "Seamless Responsive Photo Grid." *CSS-Tricks*. Chris Coyier, 24 July 2011. Web. 12 Apr. 2017.

"Create a Custom Google Map." *Maps & Directions*. Mapsdirections.info, n.d. Web. 12 Apr. 2017.

"Custom Search." *Custom Search*. Google, n.d. Web. 12 Apr. 2017.

Embedly. "Embed: The Most Powerful Tool Available for Embedding Content into Web Sites and Apps." *Embedly*. Embedly, n.d. Web. 12 Apr. 2017.

"Widgets." *Twitter Widgets*. Twitter, n.d. Web. 12 Apr. 2017.

Appendix A – Other Resources

Adobe Color CC - <https://color.adobe.com/create/color-wheel/>

Stack Overflow - <http://stackoverflow.com/>

w3schools - <https://www.w3schools.com/>

WAMP Server - <http://www.wampserver.com/en/>

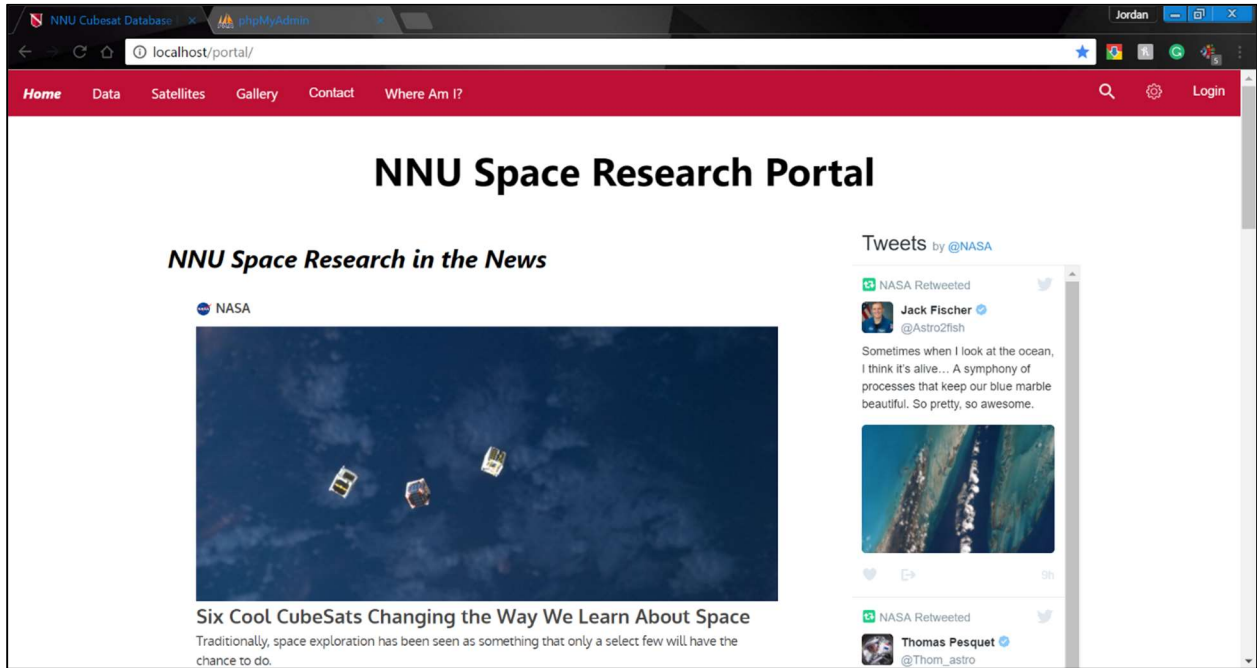
WAMP Server forums - <http://forum.wampserver.com/list.php?2>

Sublime Text 3 - <https://www.sublimetext.com/>

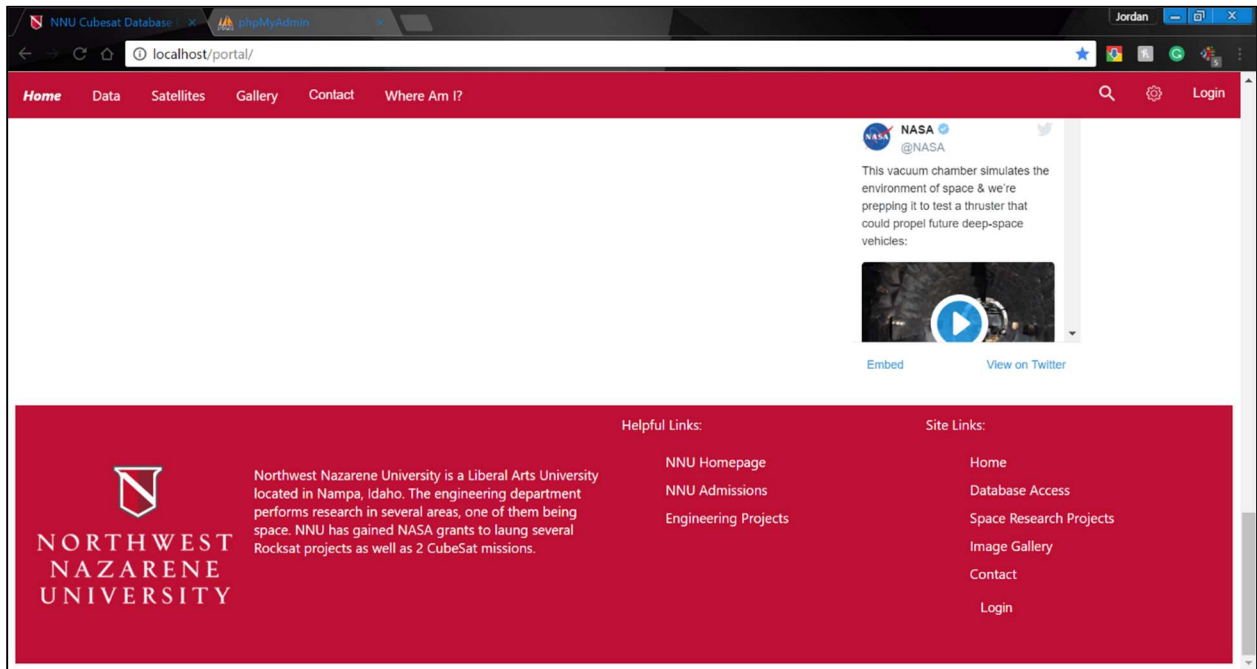
phpMyAdmin - <https://www.phpmyadmin.net/>

Code Academy - <https://www.codecademy.com/learn/all>

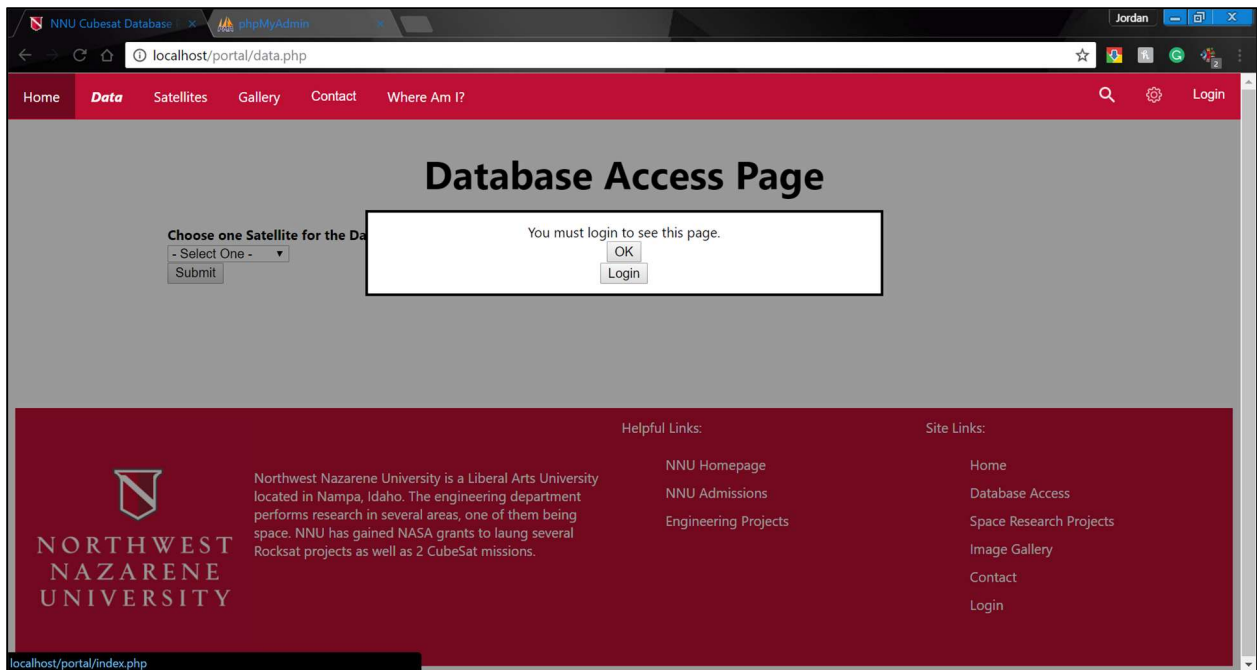
Appendix B – Screenshots



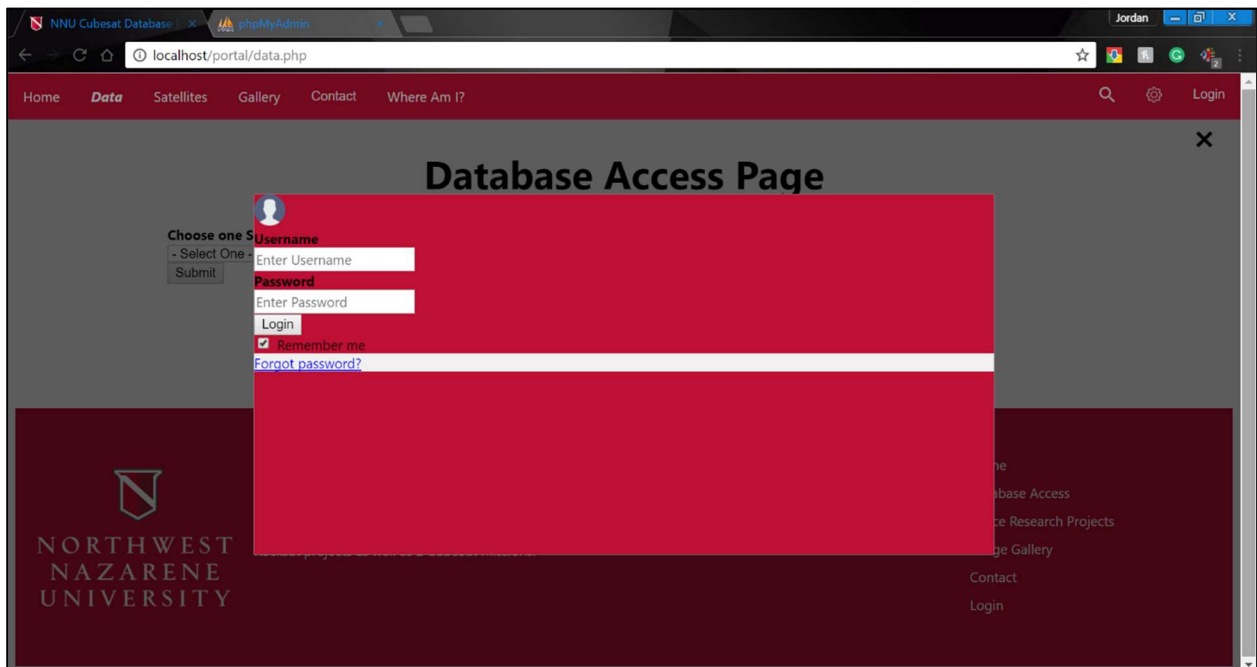
Homepage



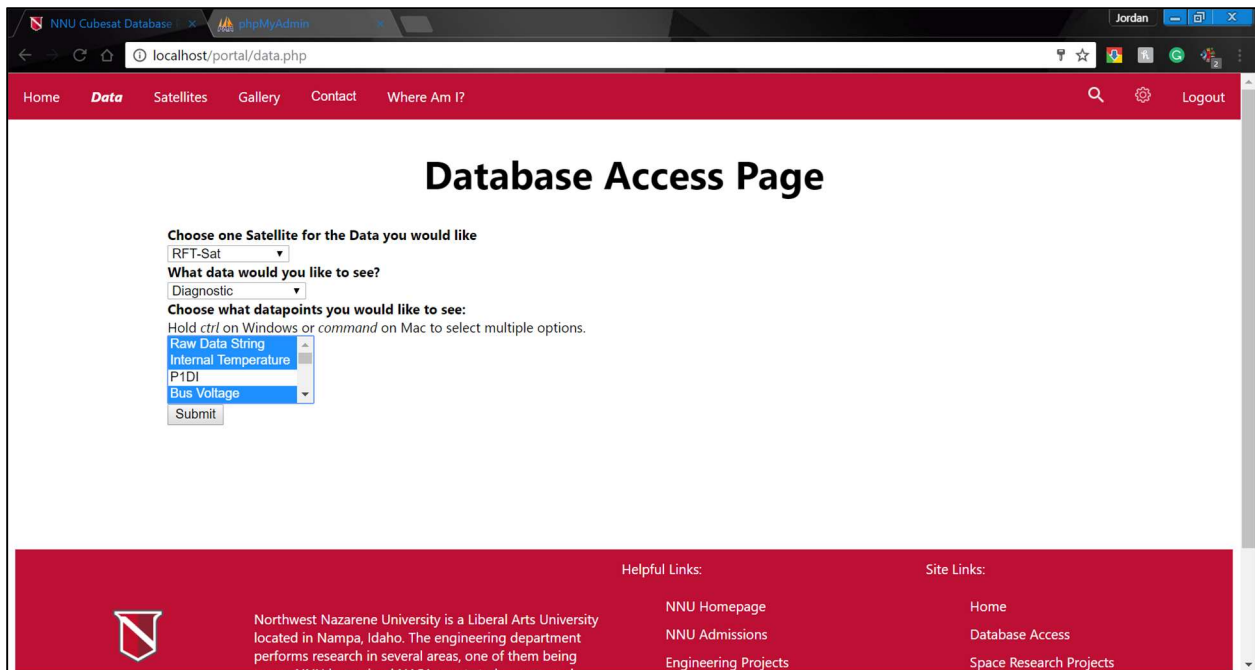
Footer shown at the bottom of every page



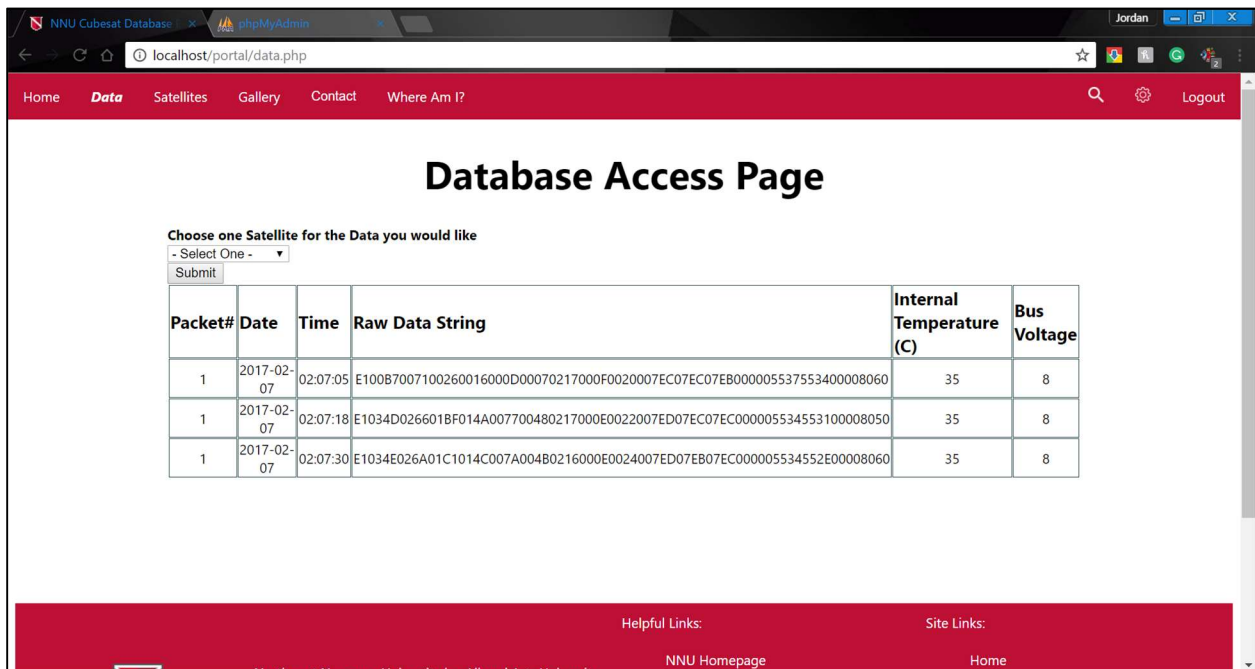
Must Login Error Message on data.php



Modal Login Form



data.php selecting data types



data.php with data table showing

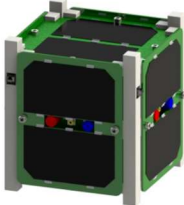
Browser tabs: NNU Cubesat Database, phpMyAdmin. Address bar: localhost/portal/satellites.php. Navigation: Home, Data, Satellites, Gallery, Contact, Where Am I?. Search, Logout.

Current and Past Projects

CubeSats:

A CubeSat (U-class spacecraft) is a type of miniaturized satellite for space research that is made up of multiples of 10×10×11.35 cm cubic units. CubeSats have a mass of no more than 1.33 kilograms per unit, and often use commercial off-the-shelf (COTS) components for their electronics and structure. More information about CubeSats can be found at [the official CubeSat website](#).

Maker-Sat I



MakerSat is a multi-project 1U CubeSat that will 3D printed, assembled, and deployed on the International Space Station (ISS). MakerSat aspires to help pioneer space manufacturing by reducing the time, cost, and complexity needed to create a CubeSat.

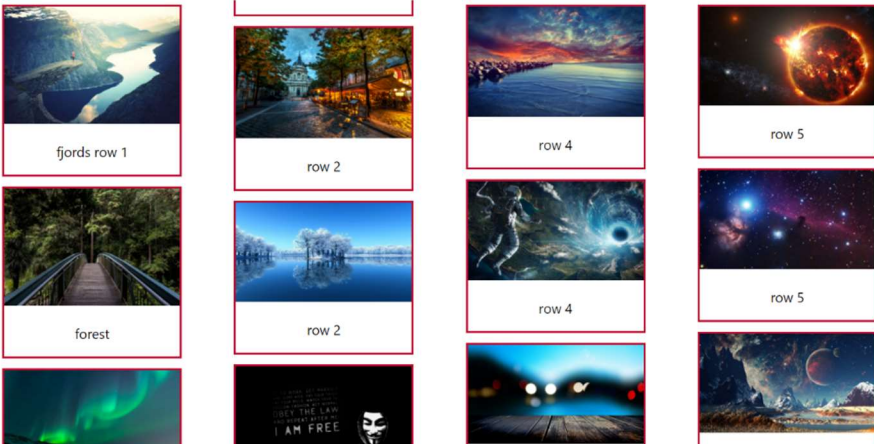
Mission Objectives:

- Demonstrate space additive manufacturing and assembly on the ISS
- Demonstrate multi-user, multi-project satellite architecture

Satellites

Browser tabs: NNU Cubesat Database, phpMyAdmin. Address bar: localhost/portal/gallery.php. Navigation: Home, Data, Satellites, Gallery, Contact, Where Am I?. Search, Logout.

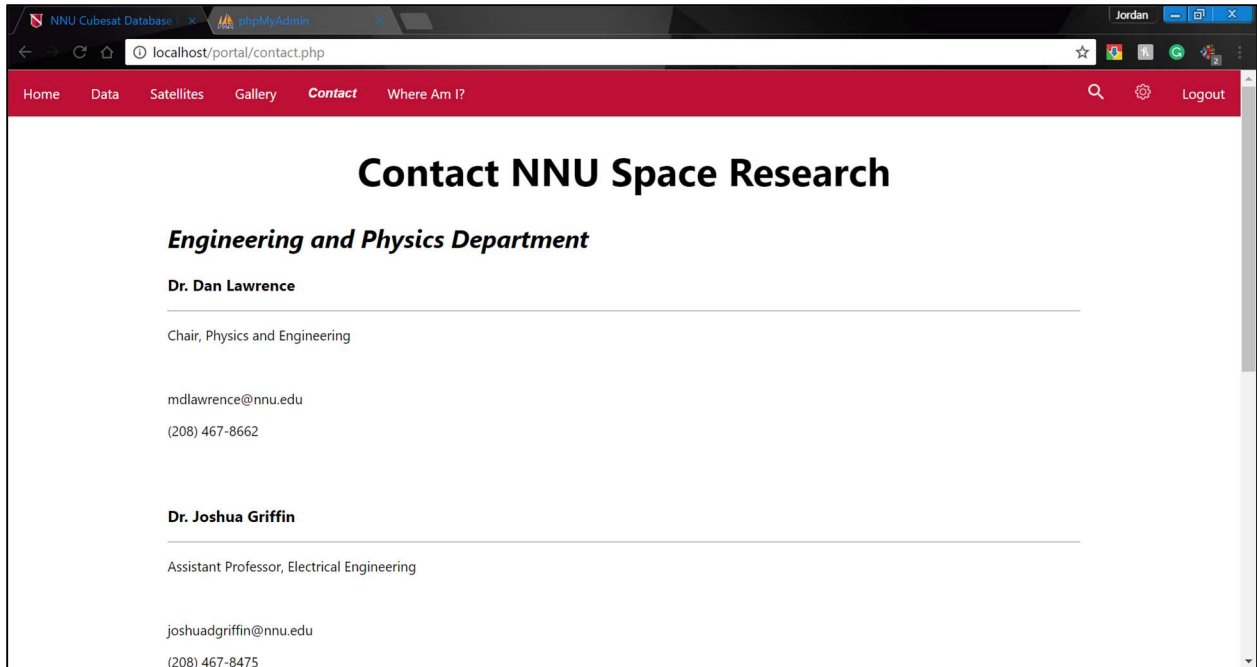
NNU Space Research Image Gallery



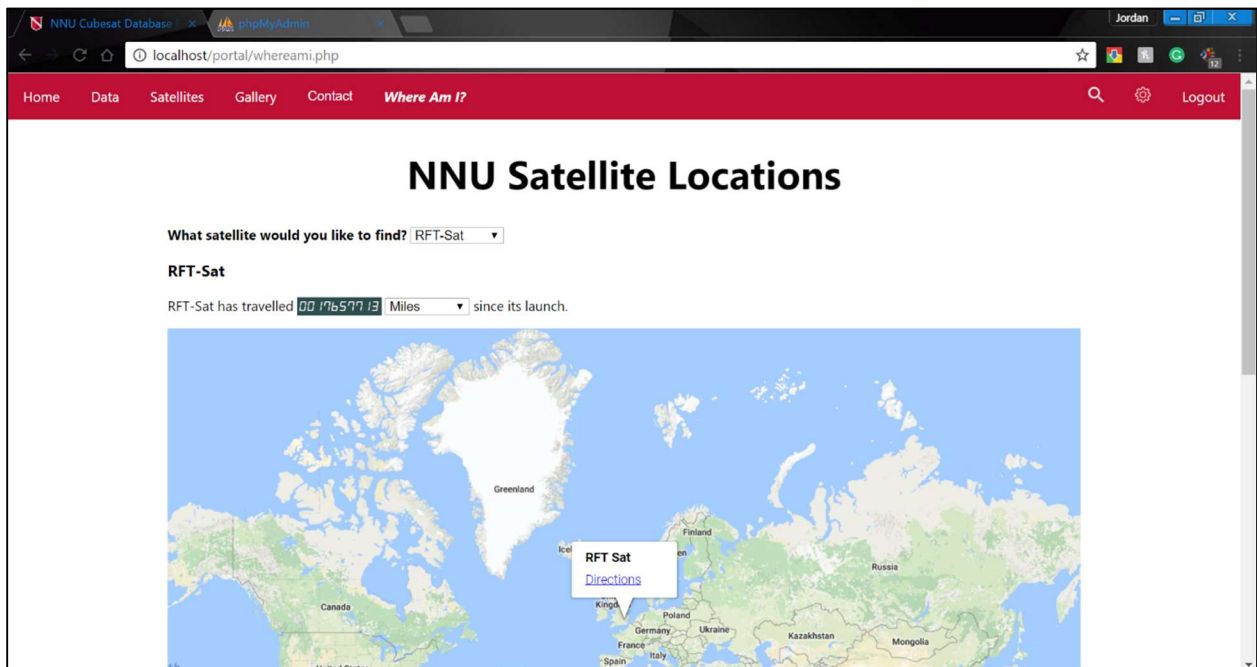
The gallery displays a collection of images including:

- fjords row 1
- row 2
- row 4
- row 5
- forest
- row 2
- row 4
- row 5
- aurora borealis
- I AM FREE
- night street scene
- landscape with moon

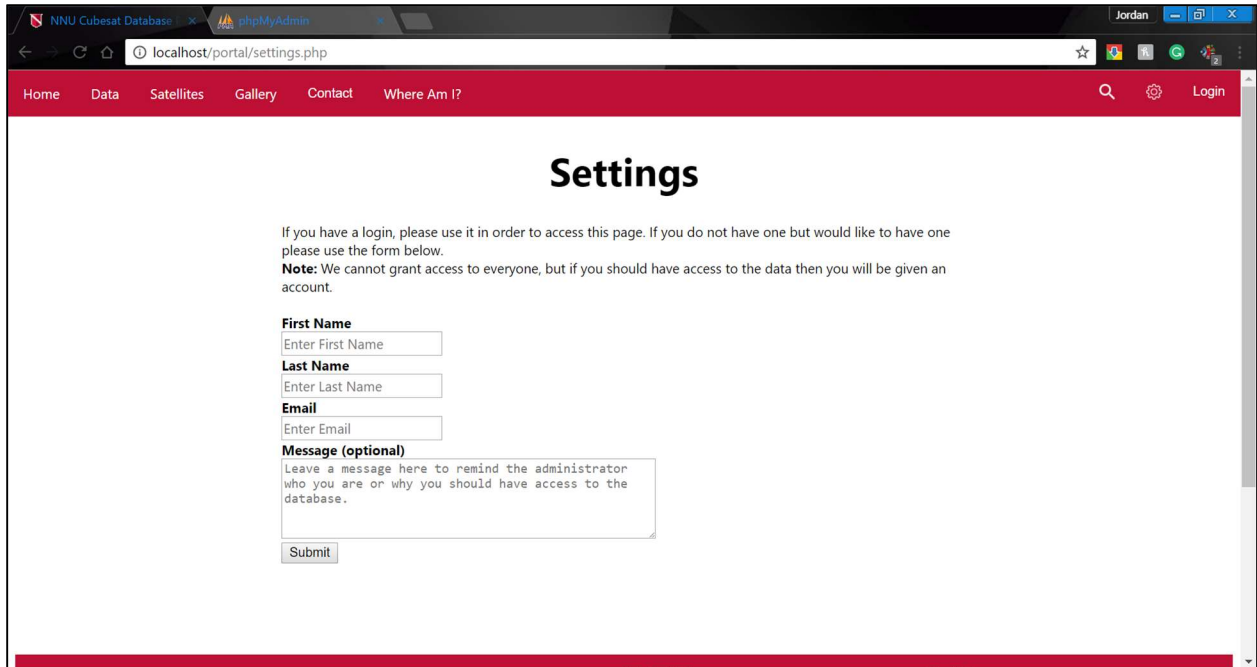
Gallery



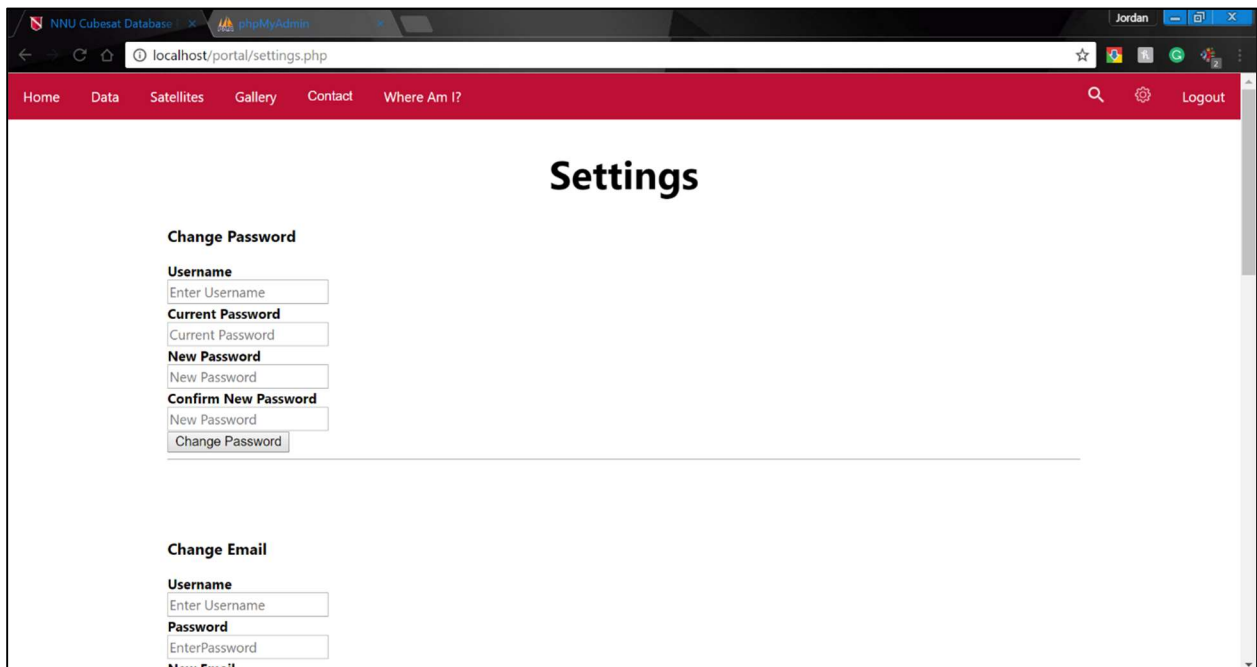
Contact



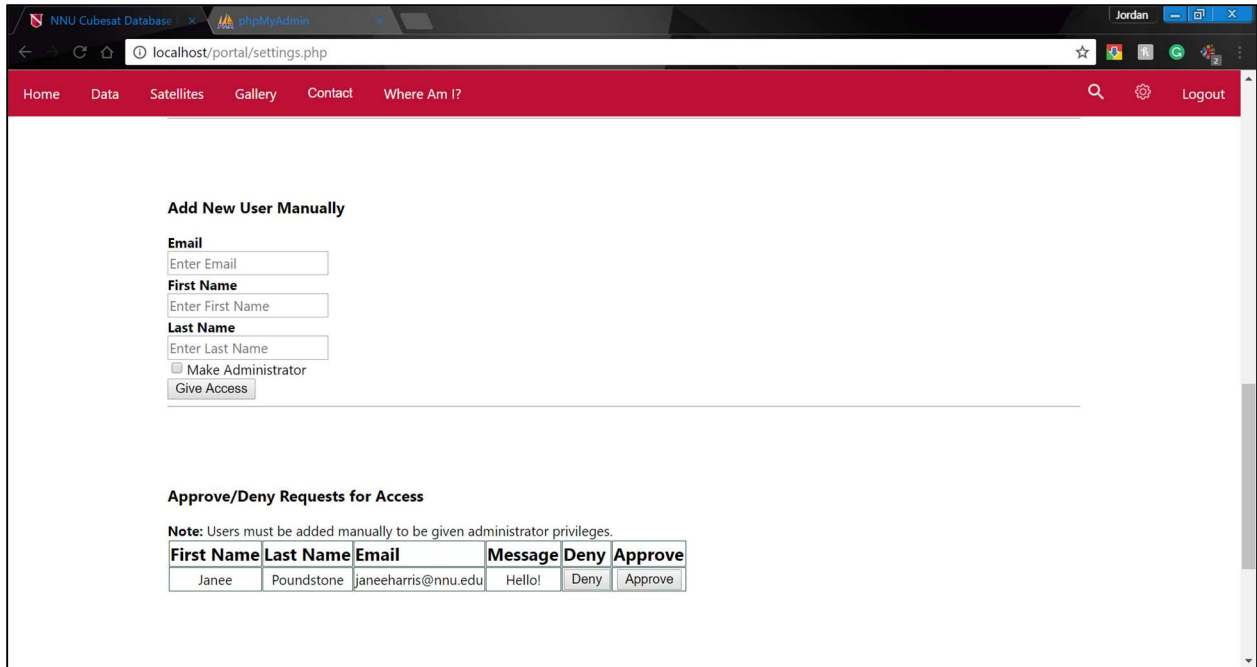
Where Am I?



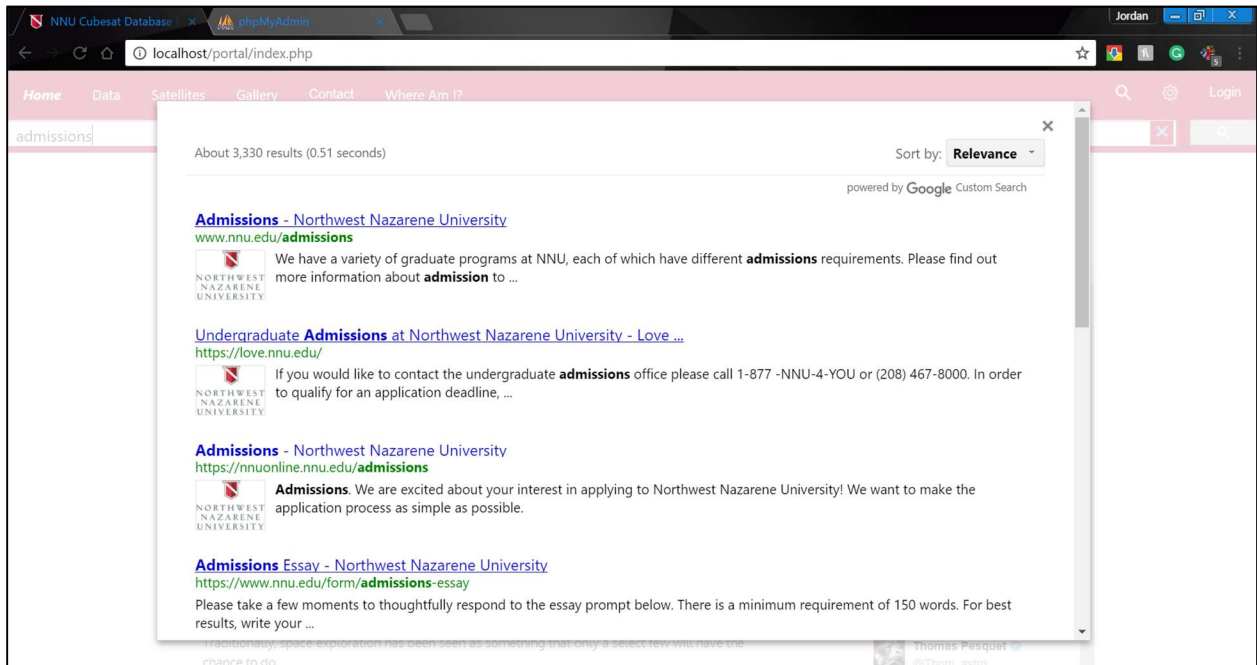
Registration page shown when visiting settings.php without logging in



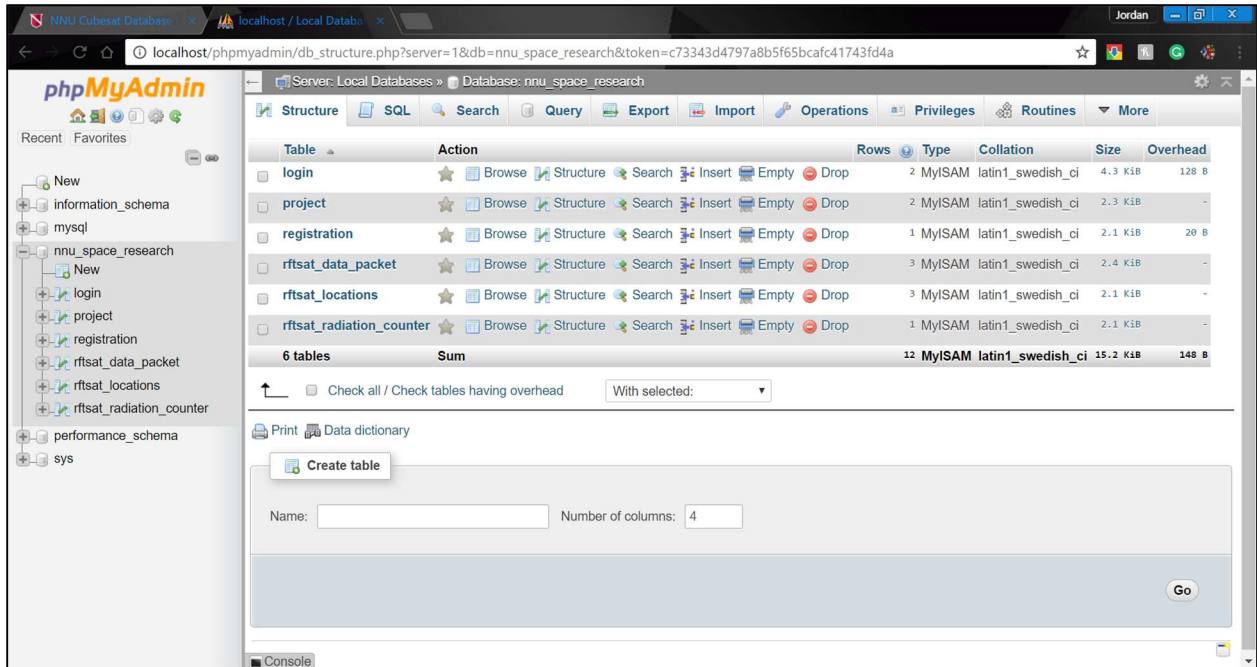
Settings



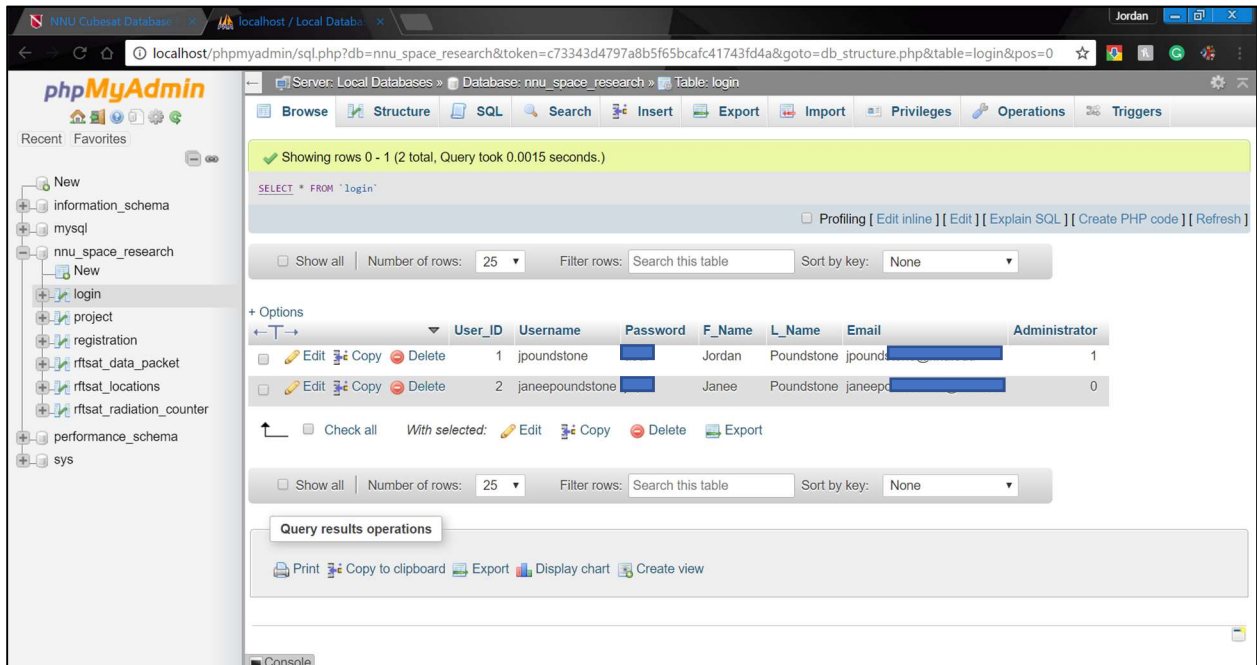
Add User function on Settings page



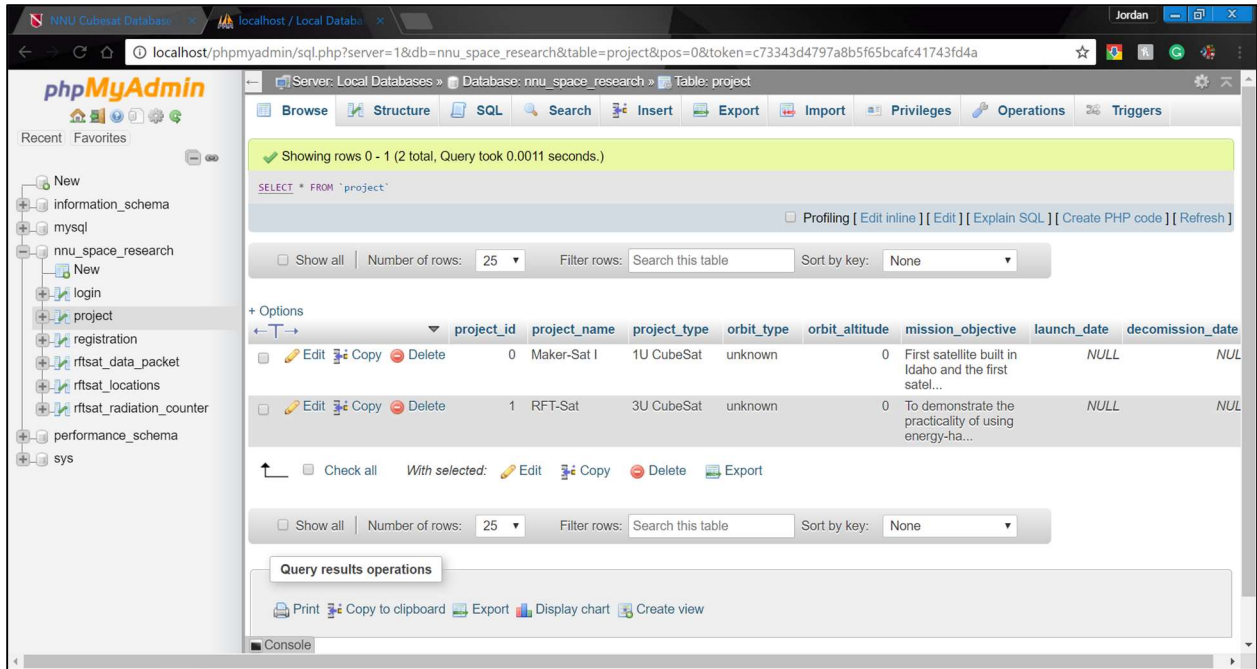
Google Custom Search Engine results



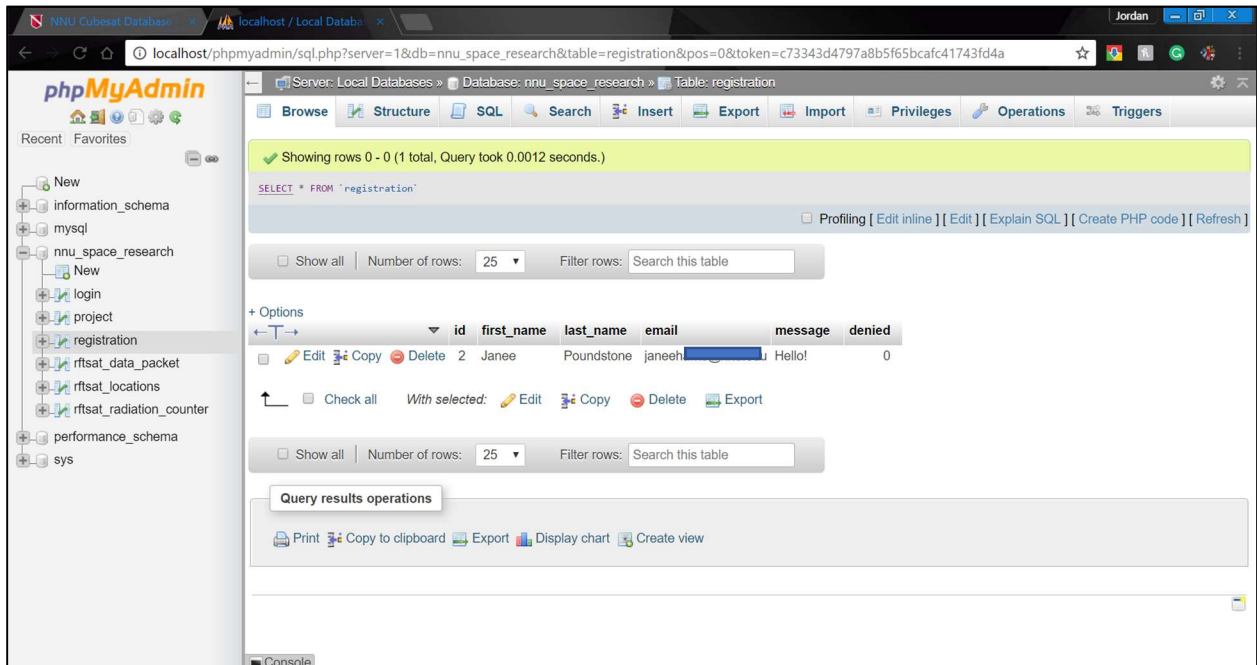
phpMyAdmin used for development of the database



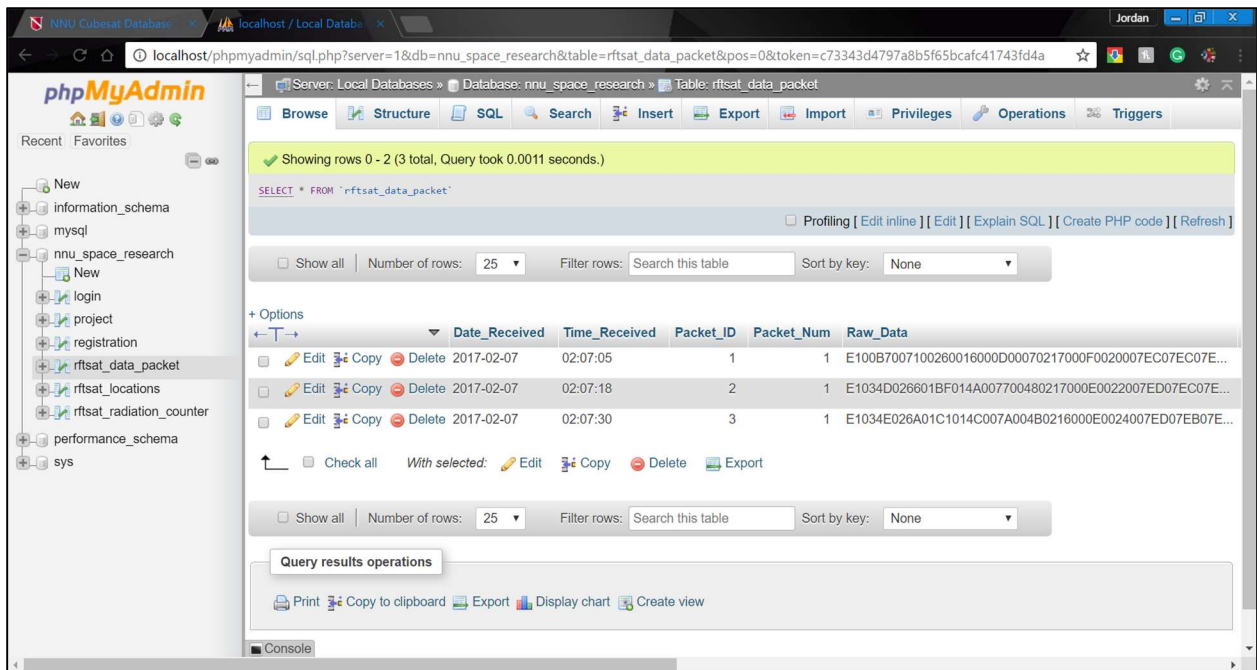
Login Table in phpMyAdmin



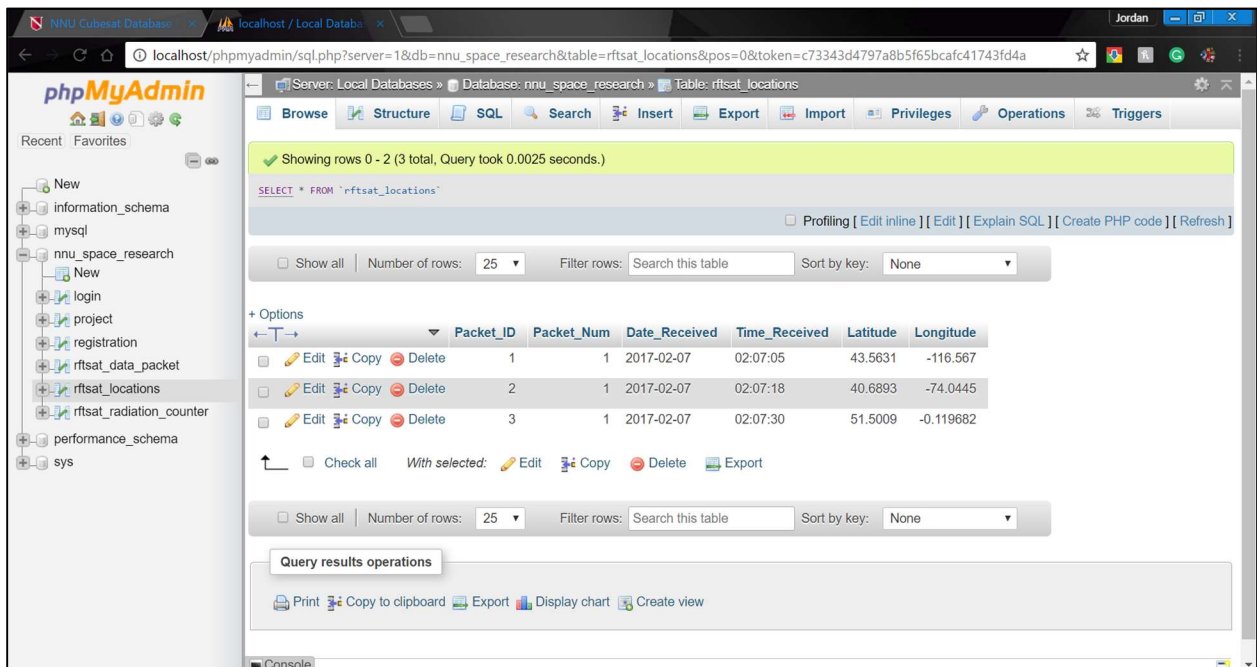
Project Table in phpMyAdmin



Registration Table in phpMyAdmin



RFT-Sat Data Packet Table in phpMyAdmin



Geographic Locations of RFT-Sat Table in phpMyAdmin

Appendix C – Code

The remainder of this document contains the actual code which was written to complete the project. Sections with a label that ends in .php or .css are the actual files which comprise the website, while others such as “Navigation Bar” that do not have a file type contain code that appeared in multiple files and was removed in the interest of a more concise appendix.

Head

The heading shows up at the top of every page, it includes the title, links to the css file and browser icon, and a php script. The php script is used to handle errors with the login system. It hides or displays blocks which alert the users to incorrect usernames or passwords. If an error is thrown, it also displays the login form when the page reloads.

```
<head>
  <title>NNU Cubesat Database Portal</title>
  <link href="style.css" type="text/css" rel="stylesheet">
  <link rel="shortcut icon" type="image/x-icon" href="favicon.ico"
  />

<?php
  if (session_status() == PHP_SESSION_NONE) {
    session_start();
  }

  if (isset($_SESSION['errors'])) {
    if($_SESSION['errors'] == "login-user") {
```



```

echo "
<style>
    .uname_error_message {
        display: block;
    }
    .pass_error_message {
        display: none;
    }
    .modal {
        display: block;
    }
</style>";
}
else if($_SESSION['errors'] == "login-pass") {
echo "
<style>
    .uname_error_message {
        display: none;
    }
    .pass_error_message {
        display: block;
    }
    .modal {
        display: block;
    }
</style>";
}
else {
echo "

```

```

    <style>
        .uname_error_message {
            display: none;
        }
        .pass_error_message {
            display: none;
        }
        .modal {
            display: none;
        }
    </style>";
}
}

if(isset($_SESSION['username'])) {
    echo "
    <style>
        .loginbtn {
            display: none;
        }
        .logoutbtn {
            display: block;
        }
    </style>";
}
else
{
    echo "
    <style>

```

```
.loginbtn {  
    display: block;  
}  
.logoutbtn {  
    display: none;  
}  
</style>" ;  
}  
?>  
</head>
```

Navigation Bar

The navigation bar is another piece of code that shows up at the top of every file. This section is in the body tag and is displayed at the top of every page on the website. This code creates links to all the pages of the website and is given better visual properties by the css file.

```
<ul id="NavBar">
  <li class="NavItem">
    <a class="active" href="index.php">Home</a>
  </li>
  <li class="NavItem"><a href="data.php">Data</a></li>
  <li class="NavItem"><a href="satellites.php">Satellites</a></li>
  <li class="NavItem"><a href="gallery.php">Gallery</a></li>
  <li class="NavItem dropdownnav">
    <div class="dropdown">
      <button class="dropbtn">Contact</button>
      <div class="drop-content">
        <a href="contact.php">NNU Science Departments</a>
        <a href="contact_rft.php">RFT-SAT</a>
        <a href="#">MakerSat I</a>
      </div>
    </div>
  </li>
  <li class="NavItem"><a href="whereami.php">Where Am I?</a></li>
  <li class="NavItemRight">
    <button onclick="document.getElementById('id01').style.display='block'" class="loginbtn">Login</button>
  </li>
</ul>
```

```
<li class="NavItemRight">
  <a href="logout.php" class="logoutbtn">Logout</a>
</li>
<li class="NavItemRight">
  <a href="settings.php"></a>
</li>
<li class="NavItemRight">
  <button onclick="toggleSearch();" id="opensearch">
    
  </button>
</li>
<li class="NavItemRight">
  <div id="searchbar" style="display:none;">
    <gcse:searchbox></gcse:searchbox>
  </div>
</li>
</ul>
```

Modal Login Form

The following code is the modal login form. This is for the form itself, creating the container and populating it with the fields to collect data. The login button on the navigation bar will display the form, while the `` on the second line creates an x button to close the form.

```
<div id="id01" class="modal">
  <span
    onclick="document.getElementById('id01').style.display='none'"
    class="close" title="Close Modal">&times;</span>

  <form class="modal-content animate" action="login.php"
    method="post">
    <div class="imgcontainer">
      
    </div>

    <div class="container">
      <label><b>Username</b></label><br>
      <input type="text" placeholder="Enter Username" name="uname"
        required><br>

      <label><b>Password</b></label><br>
      <input type="password" placeholder="Enter Password"
        name="pass" required><br>
```

```

<input type="text" name="page" value="index.php"
      style="display:none">

<button type="submit">Login</button><br>
<input type="checkbox" checked="checked"> Remember me
</div>

<div class="container" style="background-color:#F1F1F1">
  <span class="pass"><a href="#">Forgot password?</a></span>
</div>
</form>
</div>

<div class="pass_error_message">
  Username or password is incorrect.
  <button onclick="this.parentElement.style.display='none';">
    OK</button>
</div>

```

The code below this is an expansion of the modal login form. This JavaScript allows the form to be closed by clicking anywhere outside the borders of the form.

```

<script>
  var modal = document.getElementById('id01');

  window.onclick = function(event)
  {

```

```
    if (event.target == modal)
    {
        modal.style.display = "none";
    }
}
</script>
```


Footer

The footer section appears at the bottom of every page on the website. This contains helpful links to navigate within the website as well as relevant pages on NNU's website.

```
<div class="Footer">
  <div class="About">
    
    <br><br>
    <p>Northwest Nazarene University is a Liberal Arts University
      located in Nampa, Idaho. The engineering department performs
      research in several areas, one of them being space. NNU has
      gained NASA grants to launch several Rocksats projects as well
      as 2 CubeSat missions.</p>
  </div>

  <div class="Links">
    Helpful Links:
    <ul>
      <li><a href="http://nnu.edu">NNU Homepage</a></li>
      <li><a href="http://love.nnu.edu">NNU Admissions</a></li>
      <li><a href="https://engineering.nnu.edu/research/">
        Engineering Projects</a></li>
    </ul>
  </div>

  <div class="Links">
    Site Links:
```

```
<ul>
  <li><a href="home.html">Home</a></li>
  <li><a href="data.html">Database Access</a></li>
  <li><a href="satellites.html">Space Research
    Projects</a></li>
  <li><a href="gallery.html">Image Gallery</a></li>
  <li><a href="contact.html">Contact</a></li>
  <li><button
    onclick="document.getElementById('id01').style.display=
    'block'" class="loginbtn">Login</button></li>
</ul>
</div>
</div>
```

Custom Search Engine

The Google Custom Search Engine code in this website was generated automatically using Google's CSE API. The code is configured and then copy and pasted into the website. The first two items are the sections of the navigation bar (shown above under the Navigation Bar heading) to display the search icon and search box. Below these two `li` tags is the javascript that is used to implement the actual search functions.

```
<li class="NavItemRight"><button onclick="toggleSearch();"
    id="opensearch"></button></li>
<li class="NavItemRight"><div id="searchbar"
    style="display:none;"><gcse:searchbox></gcse:searchbox></div></li>

<script>
    (function() {
        var cx = '006264526096789928836:qgpjinwxowu';
        var gcse = document.createElement('script');
        gcse.type = 'text/javascript';
        gcse.async = true;
        gcse.src = 'https://cse.google.com/cse.js?cx=' + cx;
        var s = document.getElementsByTagName('script')[0];
        s.parentNode.insertBefore(gcse, s);
    })();
</script>
<gcse:searchresults></gcse:searchresults>
```

```
<script>
  function toggleSearch()
  {
    var searchbox = document.getElementById('searchbar');
    if(searchbox.style.display === "none")
    {
      searchbox.style.display = "block";
    }
    else
    {
      searchbox.style.display = "none";
    }
  }
</script>
```

index.php

index.php is the homepage of the website. This code implements a homepage with information about NNU Space Research, embedded news articles, and a live Twitter feed.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
```

```
<h1>NNU Space Research Portal</h1>
```

```
<div class="Content">
```

```
  <div class="MainPageInfo">
```

```
    <p><!-- Contains a description of NNU Space Research --> </p>
```

```
    <h2>NNU Space Research in the News</h2>
```

```
    <blockquote class="embedly-card"><h4><a
```

```
      href="https://www.nasa.gov/mission_pages/station/research/new
      s/six_cool_cubesats">Six Cool CubeSats Changing the Way We
      Learn About Space</a></h4><p>Traditionally, space exploration
      has been seen as something that only a select few will have
      the chance to do.</p></blockquote>
```

```
    <script async src="//cdn.embedly.com/widgets/platform.js"
      charset="UTF-8"></script>
```

```
    <a class="embedly-card"
```

```
      href="http://kboi2.com/news/local/local-university-builds-
      idahos-first-satellite">Local university builds Idaho's first
      satellite</a>
```

```
    <script async src="http://cdn.embedly.com/widgets/platform.js"
      charset="UTF-8"></script>
```

```
<a class="embedly-card"
  href="https://www.idahoednews.org/features/caldwell-teens-
  ready-space-exploration/">Caldwell teens ready for space
  exploration</a>
<script async src="//cdn.embedly.com/widgets/platform.js"
  charset="UTF-8"></script>
</div>
```

```
<div class="MainPageFeed">
  <a class="twitter-timeline" data-height="2000"
    href="https://twitter.com/NASA">Tweets by NASA</a>
  <script async src="https://platform.twitter.com/widgets.js"
    charset="utf-8"></script>
</div>
```

```
<!-- Insert Footer, Google CSE, close modal, and toggleSearch
  JavaScript here-->
```

data.php

This page is really the heart of the website. The original goal was to create a database and a way to access it. This page uses php to retrieve desired data from the database and display it on the page by using echo. It will only echo the data that was requested while leaving any other data hidden from view. The session is checked to ensure that anyone attempting to view the data has used valid credentials to do so.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
<div class="you_must_login">
  <div class="you_must_login_content">
    You must login to see this page.
    <form action="index.php">
      <button type="submit">OK</button>
    </form>
    <button onclick="document.getElementById('id01').style.display=
      'block'">Login</button>
  </div>
</div>

<h1>Database Access Page</h1>

<div class="Content">
  <form class="dataselection" action="data.php" method="post">
    <label><b>Choose one Satellite for the Data you would
      like</b></label><br>
    <select id="satellite" name="satellite" required>
      <option value="none">- Select One -</option>
      <option value="rftsat">RFT-Sat</option>
      <option value="mk1sat">Maker-Sat I</option>
```

```

    <option value="rsat16">Rock-Sat X 2016</option>
</select><br>

<div id="rftdatadesired" style="display: none;">
    <label><b>What data would you like to see?</b></label><br>
    <select id="rft_packettypes" name="data" required>
        <option value="none">- Select One -</option>
        <option value="sci">Science Experiment</option>
        <option value="diag">Diagnostic</option>
    </select><br>
</div>

<div id="rftdiagdatatypes" style="display: none;">
    <label><b>Choose what datapoints you would like to
        see:</b><br>Hold <i>ctrl</i> on Windows or <i>command</i>
        on Mac to select multiple options.</label><br>
    <select multiple id="rft_datatypes" name="tuples[]">
        <option value="raw">Raw Data String</option>
        <option value="intemp">Internal Temperature</option>
        <option value="p1di">P1DI</option>
        <option value="busv">Bus Voltage</option>
        <option value="bat1v">Batteries 1&4 Voltage</option>
        <option value="bat2v">Batteries 2&5 Voltage</option>
        <option value="bat3v">Batteries 3&6 Voltage</option>
        <option value="sol1v">Solar Panel 1 Voltage</option>
        <option value="sol2v">Solar Panel 2 Voltage</option>
        <option value="sol3v">Solar Panel 3 Voltage</option>
        <option value="sol1a">Solar Panel 1 Current</option>
        <option value="sol2a">Solar Panel 2 Current</option>

```



```

        <option value="sol3a">Solar Panel 3 Current</option>
    </select><br>
</div>

<button type="submit">Submit</button>

<script
    src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.
    js" ></script>
<script
    src="//ajax.googleapis.com/ajax/libs/jqueryui/1.10.2/jquery-
    ui.min.js"></script>
<script>
    $('#satellite').on('change',function(){
        if( $(this).val()=== "rftsat")
        {
            document.getElementById('rftdatadesired').style.display
                = "block";
        }
        else
        {
            document.getElementById('rftdatadesired').style.display
                = "none";
        }
    });

    $('#rft_packettypes').on('change',function(){
        if( $(this).val()=== "diag")
        {

```

```

        document.getElementById('rftdiagdatatypes').style.
            display = "block";
    }
    else
    {
        document.getElementById('rftdiagdatatypes').style.
            display = "none";
    }
});
</script>
</form>

```

```

<div style="margin-bottom: 100px;">
    <?php
        if (session_status() == PHP_SESSION_NONE) {
            session_start();
        }

        $satellite = "none";
        $packettype = "none";

        if(isset($_POST['satellite']))
        {
            $satellite = $_POST['satellite'];
        }

        if(isset($_POST['data']))
        {
            $packettype = $_POST['data'];
        }
    }

```

```

}

if($satellite == 'rftsat' and $packettype == 'diag')
{
    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "rftsat_data_packet";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username",
        "$password") or die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot
        select DB");

    $sql = "SELECT * FROM $tbl_name";
    $result = mysqli_query($connection, $sql);

    echo "<table>";
    echo "<tr><td class='data_title'>Packet#</td><td
        class='data_title'>Date</td><td
        class='data_title'>Time</td>";
    if(in_array('raw', $_POST['tuples']))
    {
        echo "<td class='data_title'>Raw Data String</td>";
    }
    if(in_array('intemp', $_POST['tuples']))
    {

```

```

        echo "<td class='data_title'>Internal Temperature
            (C)</td>";
    }
    if(in_array('p1di', $_POST['tuples']))
    {
        echo "<td class='data_title'>P1DI</td>";
    }
    if(in_array('busv', $_POST['tuples']))
    {
        echo "<td class='data_title'>Bus Voltage</td>";
    }
    if(in_array('bat1v', $_POST['tuples']))
    {
        echo "<td class='data_title'>Batteries 1&4
            Voltage</td>";
    }
    if(in_array('bat2v', $_POST['tuples']))
    {
        echo "<td class='data_title'>Batteries 2&5
            Voltage</td>";
    }
    if(in_array('bat3v', $_POST['tuples']))
    {
        echo "<td class='data_title'>Batteries 3&6
            Voltage</td>";
    }
    if(in_array('sol1v', $_POST['tuples']))
    {

```

```

        echo "<td class='data_title'>Solar Panel 1
            Voltage</td>";
    }
    if(in_array('sol2v', $_POST['tuples']))
    {
        echo "<td class='data_title'>Solar Panel 2
            Voltage</td>";
    }
    if(in_array('sol3v', $_POST['tuples']))
    {
        echo "<td class='data_title'>Solar Panel 3
            Voltage</td>";
    }
    if(in_array('sol1a', $_POST['tuples']))
    {
        echo "<td class='data_title'>Solar Panel 1
            Current</td>";
    }
    if(in_array('sol2a', $_POST['tuples']))
    {
        echo "<td class='data_title'>Solar Panel 2
            Current</td>";
    }
    if(in_array('sol3a', $_POST['tuples']))
    {
        echo "<td class='data_title'>Solar Panel 3
            Current</td>";
    }
    echo "</tr>";

```

```

while ($row = mysqli_fetch_array($result, MYSQLI_ASSOC))
{
    echo "<tr><td class='data_table'>".$row['Packet_Num'].
        "</td><td class='data_table'>".$row['Date_Received'].
        "</td><td class='data_table'>".$row['Time_Received'].
        "</td>";

    if(in_array('raw', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Raw_Data'].
            "</td>";
    }

    if(in_array('intemp', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Internal_Temp'].
            "</td>";
    }

    if(in_array('pldi', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['P1DI']."</td>";
    }

    if(in_array('busv', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Bus_Voltage'].
            "</td>";
    }

    if(in_array('bat1v', $_POST['tuples']))
    {

```

```

        echo "<td class='data_table'>".$row['Batt14_Volt'].
            "</td>";
    }
    if(in_array('bat2v', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Batt25_Volt'].
            "</td>";
    }
    if(in_array('bat3v', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Batt36_Volt'].
            "</td>";
    }
    if(in_array('sol1v', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Solar1_Volt'].
            "</td>";
    }
    if(in_array('sol2v', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Solar2_Volt'].
            "</td>";
    }
    if(in_array('sol3v', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Solar3_Volt'].
            "</td>";
    }
    if(in_array('sol1a', $_POST['tuples']))

```

```

    {
        echo "<td class='data_table'>".$row['Solar1_Amp'].
            "</td>";
    }
    if(in_array('sol2a', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Solar2_Amp'].
            "</td>";
    }
    if(in_array('sol3a', $_POST['tuples']))
    {
        echo "<td class='data_table'>".$row['Solar3_Amp'].
            "</td>";
    }
    echo "</tr>";
}

echo "</table>";

mysqli_close($connection);
}

else if ($satellite == 'rftsat' and $packettype == 'sci')
{
    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "rftsat_radiation_counter";
}

```



```

// Connect to server and select database
$connection = mysqli_connect("$host", "$username",
    "$password") or die("Cannot connect");
mysqli_select_db($connection, "$db_name") or die("cannot
    select DB");

$sql = "SELECT * FROM $tbl_name";
$result = mysqli_query($connection, $sql);

echo "<table>";
echo "<tr><td class='data_title'>Packet#</td><td
    class='data_title'>Date</td><td
    class='data_title'>Time</td><td class='data_title'>Raw
    Data String</td>";

while ($row = mysqli_fetch_array($result, MYSQLI_ASSOC))
{
    echo "<tr><td
        class='data_table'>".$row['Packet_Num']. "</td><td
        class='data_table'>".$row['Date_Received']. "</td><td
        class='data_table'>".$row['Time_Received']. "</td><td
        class='data_table'>".$row['Raw_Data']. "</td></tr>";
}

echo "</table>";

mysqli_close($connection);
}

```

```
    ?>
  </div>
</div>

<!-- Insert Footer, Google CSE, close modal, and toggleSearch
      JavaScript here-->
```

This page was a very basic one. It has short descriptions of the types of projects at NNU and the projects both completed and under development.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
```

```
<h1>Current and Past Projects</h1>
```

```
<div class="Content">
```

```
<h2>CubeSats:</h2>
```

```
<p>A CubeSat (U-class spacecraft) is a type of miniaturized satellite for space research that is made up of multiples of 10×10×11.35 cm cubic units. CubeSats have a mass of no more than 1.33 kilograms per unit, and often use commercial off-the-shelf (COTS) components for their electronics and structure. More information about CubeSats can be found at <a href="http://www.cubesat.org" target="_blank">the official CubeSat website</a>.</p>
```

```
<h3>Maker-Sat I</h3>
```

```

```

```
<p>MakerSat is a multi-project 1U CubeSat that will 3D printed, assembled, and deployed on the International Space Station (ISS). MakerSat aspires to help pioneer space manufacturing by reducing the time, cost, and complexity needed to create a CubeSat. </p>
```

```
<br style="clear: both"><p style="clear: both">Mission Objectives:</p>
```

```
<ul>
```

- Demonstrate space additive manufacturing and assembly on the ISS
- Demonstrate multi-user, multi-project satellite architecture
- Determine the rate of decay in 3D printed polymers in space in outer space
- Capture multiple photographs of Earth from orbit

<h3>RFT-Sat</h3>

<p>NNU's Radio Frequency Tag Satellite (RFTSat) is a 3U CubeSat that will perform a technology demonstration of wireless, radio frequency (RF) sensor tags that harvest RF energy and communicate with the spacecraft using backscatter. The RF tags can be configured to sense many phenomena including radiation, temperature, acceleration, electric field strength and magnetic field strength. During the mission, the tags will be deployed to demonstrate the range and effectiveness of the RF system.</p>

<br style="clear: both"><p style="clear: both">Mission

Objectives:</p>

- Collect space weather data wirelessly from an energy harvesting RF tag
- Demonstrate data collection with different distances between reader and tag
- Provide sustained data throughout the satellite's life

<h2>RockSat:</h2>

<p>The RockSat X program is sponsored by the Colorado Space Grant Consortium and allows students to build their own experimental payloads and launch them on a NASA sounding rocket.</p>

<h3>RockSat X 2016</h3>

<p>NNU's RockSat-X engineering team, in collaboration with Boise-based American Semiconductor, Inc. (ASI), launched an experimental payload on a sub-orbital sounding rocket to 100 miles above the Atlantic Ocean on a half-hour flight on Wednesday, August 12. The payload parachuted down into the Atlantic where it was recovered by ship and returned to the NNU research lab for post-flight data analysis by the team and their collaborators. The team was sponsored by ASI and the NASA Idaho Space Grant Consortium. Engineering Manager Dale Wilson served as the team's mentor from ASI.</p>

<p>The team's goal was to test flexible analog-to-digital converter ICs developed by ASI, called Flexform-ADC. This team, led by senior engineering major Jameson Krueger (Star, Idaho), also included senior physics major Paul Zukowski (Salem, Ore.) and sophomore engineering major Scott Thatcher (Nampa, Idaho). The team also tested electronics developed by another NNU engineering team, which is working on CubeSat designs with Made in Space, Inc.</p>

</div>

<!-- Insert Footer, Google CSE, close modal, and toggleSearch JavaScript here-->

gallery.php

This page uses divs to create a gallery to display any team pictures, pictures of the satellites, or any images returned from space.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
```

```
<h1>NNU Space Research Image Gallery</h1>
```

```
<div class="Content">
```

```
  <div class="gallery">
```

```
    <div class="galleryImg">
```

```
      <a target="_blank" href="images/fjords.jpg">
```

```
        
```

```
      </a>
```

```
      <div class="desc">fjords row 1</div>
```

```
    </div>
```

```
    <div class="galleryImg">
```

```
      <a target="_blank" href="images/forest.jpg">
```

```
        
```

```
      </a>
```

```
      <div class="desc">forest</div>
```

```
    </div>
```

```
    <div class="galleryImg">
```

```
      <a target="_blank" href="images/lights.jpg">
```

```
        
```

```
</a>

<div class="desc">lights</div>
</div>

<div class="galleryImg">
  <a target="_blank" href="images/mountains.jpg">
    
  </a>
  <div class="desc">mountains</div>
</div>

<!-- the galleryImg div is repeated for each additional image
      in the gallery -->
</div>
</div>
<!-- Insert Footer, Google CSE, close modal, and toggleSearch
      JavaScript here-->
```

contact.php

This is a general contact page with information for the departments pulled from NNU's website.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
```

```
<h1>Contact NNU Space Research</h1>
```

```
<div class="Content">
```

```
  <h2>Engineering and Physics Department</h2>
```

```
  <h3>Dr. Dan Lawrence</h3><hr>
```

```
  <p>Chair, Physics and Engineering</p><br>
```

```
  <p>mdlawrence@nnu.edu</p>
```

```
  <p>(208) 467-8662</p>
```

```
  <br><br>
```

```
  <h3>Dr. Joshua Griffin</h3><hr>
```

```
  <p>Assistant Professor, Electrical Engineering</p><br>
```

```
  <p>joshuadgriffin@nnu.edu</p>
```

```
  <p>(208) 467-8475</p>
```

```
  <br><br>
```

```
  <h3>Dr. Steve Parke</h3><hr>
```

```
  <p>Professor, Electrical Engineering</p><br>
```

```
  <p>sparke@nnu.edu</p>
```

```
  <p>(208) 467-8475</p>
```

```
  <br><br><br>
```

```
  <h2>Mathematics and Computer Science</h2>
```


<h3>Dr. Barry Myers</h3><hr>

**<p>Chair, Mathematics and Computer Science</p>
**

<p>blmyers@nnu.edu</p>

<p>(208) 467-8670</p>

**

**

<h3>Professor Dale Hamilton</h3><hr>

**<p>Professor, Computer Science</p>
**

<p>joshuadgriffin@nnu.edu</p>

<p>(208) 467-8475</p>

**

**

</div>

**<!-- Insert Footer, Google CSE, close modal, and toggleSearch
JavaScript here-->**

contact_rft.php

This is a more specific contact page than contact.php in the previous section. Contact_rft.php contains information for the professors and the students on the RFT-Sat team.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
```

```
<h1>Contact the RFT-Sat Team</h1>
```

```
<div class="Content">
```

```
  <h2>Professors</h2>
```

```
  <h3>Dr. Dan Lawrence - Principal Investigator</h3><hr>
```

```
  <p>Chair, Physics and Engineering</p><br>
```

```
  <p>mdlawrence@nnu.edu</p>
```

```
  <p>(208) 467-8662</p>
```

```
  <br><br>
```

```
  <h3>Dr. Joshua Griffin - Co-PI</h3><hr>
```

```
  <p>Assistant Professor, Electrical Engineering</p><br>
```

```
  <p>joshuadgriffin@nnu.edu</p>
```

```
  <p>(208) 467-8475</p>
```

```
  <br><br>
```

```
  <h3>Dr. Steve Parke - Co-PI</h3><hr>
```

```
  <p>Professor, Electrical Engineering</p><br>
```

```
  <p>sparke@nnu.edu</p>
```

```
  <p>(208) 467-8475</p>
```

```
  <br><br><br>
```

```
<h2>Students</h2>
```

```
<h3>Cassie Wade - Team Leader</h3><hr>
```

```
<p>Electrical Engineering</p>
```

```
<p>cassandrawade@nnu.edu</p>
```

```
<?php
```

```
    if (session_status() == PHP_SESSION_NONE) {  
        session_start();  
    }
```

```
    if (isset($_SESSION['username']))  
    {  
        echo "<p>(***) ***-****</p>";  
    }
```

```
?>
```

```
<br><br>
```

```
<h3>Daniel Slemmer</h3><hr>
```

```
<p>Electrical Engineering</p>
```

```
<p>dslemmer@nnu.edu</p>
```

```
<?php
```

```
    if (session_status() == PHP_SESSION_NONE) {  
        session_start();  
    }
```

```
    if (isset($_SESSION['username']))  
    {  
        echo "<p>(***) ***-****</p>";  
    }
```

?>

<h3>Curtis Garner</h3><hr>

<p>Electrical Engineering</p>

<p>madisongarner@nnu.edu</p>

<?php

```
    if (session_status() == PHP_SESSION_NONE) {  
        session_start();  
    }
```

```
    if (isset($_SESSION['username']))  
    {  
        echo "<p>(***) ***-****</p>";  
    }
```

?>

<h3>Lucas Schamber</h3><hr>

<p>Physics</p>

<p>lschamber@nnu.edu</p>

<?php

```
    if (session_status() == PHP_SESSION_NONE) {  
        session_start();  
    }
```

```
    if (isset($_SESSION['username']))  
    {  
        echo "<p>(***) ***-****</p>";  
    }
```

```

    }
?>
<br><br>

<h3>Jordan Poundstone</h3><hr>
<p>Computer Science</p>
<p>jpoundstone@nnu.edu</p>
<?php
    if (session_status() == PHP_SESSION_NONE) {
        session_start();
    }

    if (isset($_SESSION['username']))
    {
        echo "<p>(***) ***-****</p>";
    }
?>
<br><br>

<h3>Brandon Pankey</h3><hr>
<p>Electrical Engineering</p>
<p>bpankey@nnu.edu</p>
<?php
    if (session_status() == PHP_SESSION_NONE) {
        session_start();
    }

    if (isset($_SESSION['username']))
    {

```

```

        echo "<p>(***) ***-****</p>";
    }
?>
<br><br>

<h3>Brittany McCarty</h3><hr>
<p>Graphic Design</p>
<p>bmccarty@nnu.edu</p>
<?php
    if (session_status() == PHP_SESSION_NONE) {
        session_start();
    }

    if (isset($_SESSION['username']))
    {
        echo "<p>(***) ***-****</p>";
    }
?>
<br><br>

<h3>Zachary Mason</h3><hr>
<p>Film and Marketing</p>
<p>zmason@nnu.edu</p>
<?php
    if (session_status() == PHP_SESSION_NONE) {
        session_start();
    }

    if (isset($_SESSION['username']))

```

```
    {
      echo "<p>(***) ***-****</p>";
    }
?>
<br><br><br>
</div>

<!-- Insert Footer, Google CSE, close modal, and toggleSearch
JavaScript here-->
```

[whereami.php](#)

This page is more for prospective students or kids. It will show the last known geographic location of the satellite along with the estimated distance that it has travelled since launch. The maps are generated by the Google Maps API.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
```

```
<h1>NNU Satellite Locations</h1>
```

```
<div class="Content">
```

```
  <label><b>What satellite would you like to find?</b></label>
```

```
  <select id="selectsatellitelocation" default="rft">
```

```
    <option value="rft">RFT-Sat</option>
```

```
    <option value="mk1">MakerSat I</option>
```

```
    <option value="all">All Satellites</option>
```

```
</select>
```

```
<div id="rftsatlocationcontainer">
```

```
  <div style="width: 100%" id="rftsatlocation">
```

```
    <h3>RFT-Sat</h3>
```

```
    <p>RFT-Sat has travelled
```

```
      <span id="rftdistancemiles" class="odometer"></span>
```

```
      <span id="rftdistancekilometers" class="odometer"
```

```
        style="display:none"></span>
```

```
      <select id="rftunits" default="mi">
```

```
        <option value="mi">Miles</option>
```

```
        <option value="km">Kilometers</option>
```

```
      </select> since its launch.
```

```
</p>
```



```

<?php

    if (session_status() == PHP_SESSION_NONE) {
        session_start();
    }

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "rftsat_locations";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username",
        "$password") or die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot
        select DB");

    $sql = "SELECT * FROM $tbl_name ORDER BY Packet_ID DESC
        LIMIT 1";

    $result = mysqli_query($connection, $sql);

    $row = mysqli_fetch_array($result, MYSQLI_ASSOC);

    $src = "http://www.mapsdirections.info/en/custom-google-
        maps/map.php?width=100&height=600&hl=ru&co
        ord=" . $row['Latitude'] . ", " . $row['Longitude'] . "&q=%2
        0+(RFT%20Sat) &ie=UTF8&t=&z=2&iwloc=A&am
        p;output=embed";

```

```

        echo "<iframe width='100%' height='600' src=".$src."
            frameborder='0' scrolling='no' marginheight='0'
            marginwidth='0'></iframe>";

    ?>

</div>
</div><br>

<div id="mksatlocationcontainer" style="display:none">
    <div style="width: 100%" id="mksatlocation">
        <h3>MakerSat I</h3>

        <p>MakerSat I has travelled
        <span id="mkldistancemiles" class="odometer"></span>
        <span id="mkldistancekilometers" class="odometer"
            style="display:none"></span>
        <select id="mklunits">
            <option value="mi">Miles</option>
            <option value="km">Kilometers</option>
        </select> since its launch.
        </p>

        <iframe width="100%" height="600"
            src="http://www.mapsdirections.info/en/custom-google-maps/
            map.php?width=100%&height=600&hl=ru&coord=38.991496034,-
            76.842329964&q=%20+(MakerSat%20I)&ie=UTF8&t=&z=2&iwloc=A&o
            utput=embed" frameborder="0" scrolling="no"
            marginheight="0" marginwidth="0">
        </iframe>
    </div>
</div>

```

```
</div><br>
```

```
<script
```

```
  src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js  
  " ></script>
```

```
<script
```

```
  src="//ajax.googleapis.com/ajax/libs/jqueryui/1.10.2/jquery-  
  ui.min.js"></script>
```

```
<script>
```

```
$('#selectsatellitelocation').on('change',function()  
{  
  if( $(this).val()=== "rft")  
  {  
    document.getElementById('rftsatlocationcontainer').style.  
      display = "block";  
    document.getElementById('mksatllocationcontainer').style.  
      display = "none";  
  }  
  else if( $(this).val()=== "mk1")  
  {  
    document.getElementById('rftsatlocationcontainer').style.  
      display = "none";  
    document.getElementById('mksatllocationcontainer').style.  
      display = "block";  
  }  
  else if( $(this).val()=== "all")  
  {
```

```

        document.getElementById('rftsatlocationcontainer').style.
            display = "block";
        document.getElementById('mksatllocationcontainer').style.
            display = "block";
    }
});

```

```

$('#rftunits').on('change', function()
{
    if( $(this).val() === "mi" )
    {
        document.getElementById('rftdistancemiles').style.display
            = "block";
        document.getElementById('rftdistancekilometers').style.
            display = "none";
    }
    else
    {
        document.getElementById('rftdistancemiles').style.display
            = "none";
        .getElementById('rftdistancekilometers').style.display =
            "block";
    }
});

```

```

$('#mklunits').on('change', function()
{
    if( $(this).val() === "mi" )
    {

```

```

        document.getElementById('mk1distancemiles').style.display
            = "block";
        document.getElementById('mk1distancekilometers').style.
            display = "none";
    }
    else
    {
        document.getElementById('mk1distancemiles').style.display
            = "none";
        document.getElementById('mk1distancekilometers').style.
            display = "block";
    }
});
</script>

```

```

<script>
    function startTime()
    {
        startTimeRFT();
        startTimeMK1();
    }

    function startTimeRFT()
    {
        var time = Date.now() - 1490000000000;
        var distancemi = time*4.76/1000;
        var distancekm = time*7.66/1000;
        distancemi = distancemi.toFixed(0);
        distancekm = distancekm.toFixed(0);
    }

```

```

    distancemi = addZeroes(distancemi);
    distancekm = addZeroes(distancekm);

    document.getElementById('rftdistancemiles').innerHTML =
        distancemi;
    document.getElementById('rftdistancekilometers').innerHTML =
        distancekm;

    t = setTimeout(function(){startTime()}, 50);
}

function startTimeMK1()
{
    var time = Date.now() - 1480500000000;
    var distancemi = time*4.76/1000;
    var distancekm = time*7.66/1000;
    distancemi = distancemi.toFixed(0);
    distancekm = distancekm.toFixed(0);
    distancemi = addZeroes(distancemi);
    distancekm = addZeroes(distancekm);

    document.getElementById('mk1distancemiles').innerHTML =
        distancemi;
    document.getElementById('mk1distancekilometers').innerHTML =
        distancekm;

    t = setTimeout(function(){startTime()}, 50);
}

```

```
function addZeroes(distance)
{
  if (distance < 10)
  {
    distance = "000000000" + distance;
  }
  else if (distance < 100)
  {
    distance = "00000000" + distance;
  }
  else if (distance < 1000)
  {
    distance = "0000000" + distance;
  }
  else if (distance < 10000)
  {
    distance = "000000" + distance;
  }
  else if (distance < 100000)
  {
    distance = "00000" + distance;
  }
  else if (distance < 1000000)
  {
    distance = "0000" + distance;
  }
  else if (distance < 10000000)
  {
    distance = "000" + distance;
  }
}
```

```
    }  
  
    else if (distance < 100000000)  
    {  
        distance = "00" + distance;  
    }  
  
    else if (distance < 1000000000)  
    {  
        distance = "0" + distance;  
    }  
  
    return distance;  
}  
  
    startTime();  
</script>  
</div>  
  
<!-- Insert Footer, Google CSE, close modal, and toggleSearch  
    JavaScript here-->
```


settings.php

This page uses sessions to check if the user is logged in. If they have not then a registration form is displayed, but if they are then settings to change login information is displayed. The page also checks for administrator privileges from the database and displays the add user functions if appropriate.

```
<!-- Insert code from Head, Navigation Bar, and Modal Login Form -->
```

```
<h1>Settings</h1>
```

```
<div class="Content">
```

```
  <div class="loggedin">
```

```
    <h3>Change Password</h3>
```

```
    <form action="changepass.php" method="post">
```

```
      <label><b>Username</b></label><br>
```

```
      <input type="text" placeholder="Enter Username" name="uname"
        required><br>
```

```
      <label><b>Current Password</b></label><br>
```

```
      <input type="password" placeholder="Current Password"
        name="cpass" required><br>
```

```
      <label><b>New Password</b></label><br>
```

```
      <input type="password" placeholder="New Password" name="npass"
        required><br>
```

```
      <label><b>Confirm New Password</b></label><br>
```

```
      <input type="password" placeholder="New Password" name="rpass"
        required><br>
```

```

<button type="submit">Change Password</button><br>

<div class="passwordsnomatch">
    Error: Passwords did not match.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>

<div class="passsuccess">
    Password successfully changed.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>
</form><hr><br><br><br>

<h3>Change Email</h3>
<form action="changeemail.php" method="post">
    <label><b>Username</b></label><br>
    <input type="text" placeholder="Enter Username" name="uname"
        required><br>

    <label><b>Password</b></label><br>
    <input type="password" placeholder="EnterPassword" name="pass"
        required><br>

    <label><b>New Email</b></label><br>
    <input type="text" placeholder="New Email" name="nmail"
        required><br>

```

```

<label><b>Confirm New Email</b></label><br>
<input type="text" placeholder="New Email" name="rmail"
    required><br>

<button type="submit">Change Email</button><br>

<div class="emailsnomatch">
    Error: Emails did not match.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>

<div class="emailsuccess">
    Email successfully changed.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>
</form><hr><br><br><br>

<h3>Change Username</h3>
<form action="changeusername.php" method="post">
    <label><b>Username</b></label><br>
    <input type="text" placeholder="Enter Username" name="uname"
        required><br>

    <label><b>Password</b></label><br>
    <input type="password" placeholder="EnterPassword" name="pass"
        required><br>

```

```

<label><b>New Username</b></label><br>
<input type="text" placeholder="New Username" name="nuser"
    required><br>

<label><b>Confirm New Username</b></label><br>
<input type="text" placeholder="New Username" name="ruser"
    required><br>

<button type="submit">Change Username</button><br>

<div class="unamesnomatch">
    Error: Usernames did not match.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>

<div class="unamesuccess">
    Username successfully changed.
    <button
        onclick="this.parentElement.style.display='none';">OK</button
    >
</div>
</form><hr><br><br><br>

<div class="administrator">
    <h3>Add New User Manually</h3>
    <form action="adduser.php" method="post">
        <label><b>Email</b></label><br>

```

```

<input type="text" placeholder="Enter Email" name="email"
    required><br>

<label><b>First Name</b></label><br>
<input type="text" placeholder="Enter First Name"
    name="fname" required><br>

<label><b>Last Name</b></label><br>
<input type="text" placeholder="Enter Last Name" name="lname"
    required><br>

<input type="checkbox" name="admin" value="yes">Make
    Administrator<br>
<button type="submit">Give Access</button><br>
</form>

<div class="waitingusers">
<?php
    if (session_status() == PHP_SESSION_NONE) {
        session_start();
    }

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "registration";

    // Connect to server and select database

```

```

$connection = mysqli_connect("$host", "$username",
    "$password") or die("Cannot connect");
mysqli_select_db($connection, "$db_name") or die("cannot
    select DB");

$sql = "SELECT * FROM $tbl_name WHERE denied=0";
$result = mysqli_query($connection, $sql);
$count = mysqli_num_rows($result);

if($count > 0)
{
    echo "<hr><br><br><br><h3>Approve/Deny Requests for
        Access</h3>";
    echo "<b>Note:</b> Users must be added manually to be
        given administrator privileges.";
    echo "<table>";
    echo "<tr><td class='data_title'>First Name</td>
        <td class='data_title'>Last Name</td>
        <td class='data_title'>Email</td>
        <td class='data_title'>Message</td>
        <td class='data_title'>Deny</td>
        <td class='data_title'>Approve</td></tr>";
    while ($row = mysqli_fetch_array($result, MYSQLI_ASSOC))
    {
        if($row['denied'] == 0)
        {
            echo "<tr>";
            echo "<td class='data_table'>";
            echo $row['first_name'];

```

```

echo "</td><td class='data_table'>";
echo $row['last_name'];
echo "</td><td class='data_table'>";
echo $row['email'];
echo "</td><td class='data_table'>";
echo $row['message'];
echo "</td><td class='data_table'>
    <form action='deny.php' method='post'>
    <input type='text' name='fname' value=";
echo $row['first_name'];
echo " style='display:none'>
    <input type='text' name='lname' value=";
echo $row['last_name'];
echo " style='display:none'>
    <input type='text' name='email' value=";
echo $row['email'];
echo " style='display:none'><button type='submit'>
    Deny</button></form></td>";
echo "</td><td class='data_table'>
    <form action='adduser.php' method='post'>
    <input type='text' name='fname' value=";
echo $row['first_name'];
echo " style='display:none'>
    <input type='text' name='lname' value=";
echo $row['last_name'];
echo " style='display:none'>
    <input type='text' name='email' value=";
echo $row['email'];

```

```

        echo " style='display:none'><button type='submit'>
            Approve</button></form></td>";
        echo "</tr>";
    }
}
echo "</table>";
}
?>
</div>

<div class="registrationdenied">
    You have successfully denied <?php
        if(isset($_SESSION['denieduser'])) echo
            $_SESSION['denieduser']; ?>'s request for access.
    <button
        onclick="this.parentElement.style.display='none';">OK</but
        ton>
</div>

<div class="registrationapproved">
    <?php if(isset($_SESSION['approveduser'])) echo
        $_SESSION['approveduser']; ?>'s request for access has
        been approved.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>

<div class="useralreadyhasaccess">

```



```

Error: <?php if(isset($_SESSION['denieduser'])) echo
    $_SESSION['denieduser']; ?> already has access.
<button onclick="this.parentElement.style.display='none';">
    OK</button>
</div>
<br><br><br>
</div>

<div class="something">
    Error: Something went wrong. Please try again later.<br>
    If error persists, please contact the administrator.<br>
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>
</div>

<div class="altContent">
    If you have a login, please use it in order to access this page.
    If you do not have one but would like to have one please use
    the form below.<br>
    <b>Note:</b> We cannot grant access to everyone, but if you should
    have access to the data then you will be given an
    account.<br><br>
    <form action="registration.php" method="post">
        <label><b>First Name</b></label><br>
        <input type="text" placeholder="Enter First Name" name="fname"
            required><br>

```

```

<label><b>Last Name</b></label><br>
<input type="text" placeholder="Enter Last Name" name="lname"
    required><br>

<label><b>Email</b></label><br>
<input type="text" placeholder="Enter Email" name="email"
    required><br>

<label><b>Message (optional)</b></label><br>
<textarea rows="5" cols="50" name="message" placeholder="Leave
    a message here to remind the administrator who you are or why
    you should have access to the database."></textarea><br>

<button type="submit">Submit</button><br><br><br><br><br><br>
</form>
<div class="requestalreadySENT">
    Error: You have already submitted a request for access.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>

<div class="registrationsuccess">
    You have successfully submitted a request for access.
    <button onclick="this.parentElement.style.display='none';">
        OK</button>
</div>
</div>

```

```
<!-- Insert Footer, Google CSE, close modal, and toggleSearch  
JavaScript here-->
```

Error Handling for Settings

This code is part of the settings.php file on the website, however it was moved to its own section of the Appendix to draw a line between it and the rest of the file to keep the settings.php section more concise. Much of this section of the file is very repetitive, so most of it has been cut out, but enough has been left in to give the general idea of what was done.

```
<?php
    if (session_status() == PHP_SESSION_NONE) {
        session_start();
    }

    if (isset($_SESSION['errors']))
    {
        if($_SESSION['errors'] == "login-user")
        {
            echo "
            <style>
                .uname_error_message {
                    display: block;
                }
                .pass_error_message {
                    display: none;
                }
                .modal {
                    display: block;
                }
            </style>";
        }
    }
}
```

```

else if($_SESSION['errors'] == "login-pass")
{
    echo "
<style>
    .uname_error_message {
        display: none;
    }
    .pass_error_message {
        display: block;
    }
    .modal {
        display: block;
    }
</style>";
}
else if($_SESSION['errors'] == "passwordsnomatch")
{
    echo "
<style>
    .uname_error_message {
        display: none;
    }
    .pass_error_message {
        display: none;
    }
    .modal {
        display: none;
    }
    .passwordsnomatch {

```

```

        display: block;
    }

    .emailsnomatch {display: none;}
    .unamesnomatch {display: none;}
    .unamesuccess {display: none;}
    .passsuccess {display: none;}
    .emailsuccess {display: none;}
    .something {display: none;}
    .requestalreadysent {display: none;}
    .registrationdenied {display: none;}
    .registrationapproved {display: none;}
    .registrationsuccess {display: none;}
    .useralreadyhasaccess {display: none;}
</style>";
}

else if($_SESSION['errors'] == "passsuccess")
{
    echo "
<style>
        .uname_error_message {
            display: none;
        }

        .pass_error_message {
            display: none;
        }

        .modal {
            display: none;
        }

        .passsuccess {

```

```

        display: block;
    }

    .passwordsnomatch {display: none;}
    .emailsnomatch {display: none;}
    .unamesnomatch {display: none;}
    .unamesuccess {display: none;}
    .emailsuccess {display: none;}
    .something {display: none;}
    .requestalreadysent {display: none;}
    .registrationdenied {display: none;}
    .registrationapproved {display: none;}
    .registrationsuccess {display: none;}
    .useralreadyhasaccess {display: none;}
</style>";
}

<!-- This code repeats for each possible error -->
}
else
{
    echo "
<style>

    .uname_error_message {display: none;}
    .pass_error_message {display: none;}
    .modal {display: none;}
    .passwordsnomatch {display: none;}
    .emailsnomatch {display: none;}
    .unamesnomatch {display: none;}
    .unamesuccess {display: none;}
    .passsuccess {display: none;}

```

```
.emailsuccess {display: none;}

.something {display: none;}

.registrationapproved {display: none;}

.registrationdenied {display: none;}

.requestalreadysent {display: none;}

.registrationsuccess {display: none;}

</style>";
}

```

```
if(isset($_SESSION['username']))
```

```
{
    echo "
<style>
    .loginbtn {
        display: none;
    }
    .logoutbtn {
        display: block;
    }
    .loggedin {
        display: block;
    }
    .altContent {
        display: none;
    }
</style>";
}

```

```
else
```

```
{
```



```

echo "

<style>

    .uname_error_message {display: none;}

    .pass_error_message {display: none;}

    .modal {display: none;}

    .passwordsnomatch {display: none;}

    .emailsnomatch {display: none;}

    .unamesnomatch {display: none;}

    .unamesuccess {display: none;}

    .passsuccess {display: none;}

    .emailsuccess {display: none;}

    .something {display: none;}

    .registrationapproved {display: none;}

    .registrationdenied {display: none;}

    .requestalreadysent {display: none;}

    .registrationsuccess {display: none;}

</style>";
}

if(isset($_SESSION['admin']))
{
    echo "

    <style>

        .administrator {

            display: block;

        }

    </style>";
}

else
{

```

```
echo "  
  <style>  
    .administrator {  
      display: none;  
    }  
  </style>";  
}  
?>
```

login.php

When the login form is used, the user is redirected to this page. It does not display anything, but it checks for the credentials in the database and if the username and password are correct, a session is registered.

```
<?php
```

```
    session_start();

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "login";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username", "$password") or
        die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot select
        DB");

    // get information from $_POST
    $username=$_POST["uname"];
    $password=$_POST["pass"];
    if(isset($_POST["page"]))
    {
        $_SESSION['redirectto'] = $_POST["page"];
    }
    else
    {
```

```

    $_SESSION['redirectto'] = "index.html";
}

// To protect MySQL injection
$username = stripslashes($username);
$password = stripslashes($password);
$username = mysqli_real_escape_string($connection, $username);
$password = mysqli_real_escape_string($connection, $password);

// Create Query
$count = 0;

$sql = "SELECT * FROM $tbl_name WHERE username='$username' and
        password='$password'";
$result = mysqli_query($connection, $sql);

$count = mysqli_num_rows($result);

if($count == 1)
{
    $_SESSION['username'] = $username;
    $_SESSION['password'] = $password;
    $_SESSION['errors'] = "none";

    $sql2 = "SELECT * FROM $tbl_name WHERE username='$username' and
            Administrator=1";
    $result = mysqli_query($connection, $sql2);
    $count = mysqli_num_rows($result);
}

```

```
if($count == 1)
{
    $_SESSION['admin'] = 'yes';
}

header("location:".$_SESSION['redirectto']);
}

else
{
    $_SESSION['errors'] = "login-pass";
    header("location:".$_SESSION['redirectto']);
}
?>
```

logout.php

This is a very simple file to just destroy the session when the user is done so that restricted parts of the website cannot be accessed with their credentials.

```
<?php
    session_start();
    session_destroy();

    header("location:index.php");
?>
```

style.css

CSS is what gives websites their design. This CSS file is based on the NNU website's CSS file, but many pieces were modified to fit the purposes of the Space Research Portal.

```
/*    Navigation Bar    *****/
#NavBar {
    list-style-type: none;
    margin: 0;
    padding: 0;
    background-color: #BE0F34;
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    z-index: 2;
}

.NavItem {
    float: left;
    text-align: center;
}

.NavItemRight {
    float: right;
    text-align: center;
}

li a {
```

```
display: block;

color: white;

padding: 14px 16px;

text-decoration: none;

background-color: #BE0F34;
}

li a:hover {

background-color: #70091F;

color: white;
}

.active {

font-weight: 700;

font-style: italic;

color: white;
}

.mobilemenu {

list-style-type: none;

margin: 0;

padding: 0;

background-color: #BE0F34;

position: fixed;

top: 0;

left: 0;

width: 100%;

z-index: 1;
}
```



```

.mobilemenubtn, #mobilemenuclose {
    display: none;
    border: none;
    background-color: #BE0F34;
    padding: 10px;
    margin: auto;
    width: 100%;
    text-align: center;
}

.mobilemenubtn img{
    display: none;
    margin: auto;
}

.mobilecontact {
    text-align: center;
    display: none;
}

@media screen and (max-width: 1200px) {
    .NavItem, .NavItemRight {float: none;}
    #NavBar {position: absolute; z-index: 2;}
    .mobilemenu, .mobilemenubtn, .mobilemenubtn img, #mobilemenuclose
    {display:block;}
    .mobilecontact {display:block;}
    .dropdownnav {display:none;}
}

```

```

li button {
    display: block;
    color: white;
    padding: 14px 16px;
    text-decoration: none;
    background-color: #BE0F34;
    border: none;
}

li button:hover {
    background-color: #70091F;
    color: white;
}

/*      Dropdown Content      *****/
.dropdown {
    float: left;
    overflow: hidden;
}

.dropdown .dropbtn {
    border: none;
    outline: none;
    color: white;
    padding: 16px;
    background-color: inherit;
}

```

```

.dropdown:hover .dropbtn {
    background-color: #70091F;
}

.drop-content {
    display: none;
    position: absolute;
    min-width: 160px;
    box-shadow: 0px 8px 16px 0px rgba(0,0,0,0.2);
    z-index: 1;
}

.drop-content a {
    float: none;
    background-color: white;
    color: black;
    padding: 12px 16px;
    text-decoration: none;
    display: block;
    text-align: left;
}

.drop-content a:hover {
    background-color: #70091F;
    color: white;
}

.dropdown:hover .drop-content {
    display: block;
}

```

```
}

/* Footer *****/

.Footer {
    background-color: #BE0F34;
    overflow: hidden;
    width: 100%;
    height: 20%;
    color: white;
    margin: auto;
    padding: 10px 0;
    bottom: 0;
    left: 0;
    z-index: 10;
}

.Footer div {
    color: white;
}

.Footer ul {
    list-style-type: none;
}

.Footer li a {
    display: block;
    color: white;
    padding: 5px;
    text-decoration: none;
}
```

```

    background-color: #BE0F34;
}

.Footer li a:hover {
    background-color: #70091F;
    color: white;
}

.About {
    width: 49%;
    float: left;
    padding-right: 10px;
}

.Links {
    width: 25%;
    float: left;
    /*padding-left: 10px;*/
}

/*    Fonts    *****/
body, table, td, th, div, input, h1, h2, h3, h4, h5, h6 {
    font-family : Segoe UI, Segoe, SegoeUI-Regular-final, Tahoma,
    Helvetica, Arial, sans-serif;
    color: #000000;
}

body, table, td, th, div, input {
    font-size: 14px;
}

```

```
}

h1, h2, h3, h4, h5, h6 {
    font-size : 18px;
    text-decoration: none;
}

h1 {
    text-align: center;
    font-size: 40px;
}

h2 {
    text-align: left;
    font-size: 25px;
    font-style: italic;
}

h3 {
    text-align: left;
    font-size: 16px;
}

h4 {
    font-style: bold;
}

div#page_footer {
    font-size: 11px;
```

```

        width: 100%;

        text-align: left;
    }

    @font-face {
        font-family: digital;
        src: url('fonts/alarmclock.ttf');
    }

    /*      Content      *****/
    body {
        margin-top: 75px;
        background-color: white;
    }

    .Content {
        width: 75%;
        margin: 0 auto;
    }

    .altContent {
        width: 75%;
        margin: 0 auto;
    }

    .MainPageInfo {
        width: 70%;
        margin: 0;
        float: left;
    }

```

```

}

.MainPageFeed {
    width: 25%;
    margin: 0;
    float: right;
}

.MissionImg {
    float: left;
    width: 200px;
    padding-right: 20px;
}

@media screen and (max-width: 600px) {
    .Content {width: 100%;}
    body {margin-top: 400px;}
    .MainPageFeed {display: none;}
    .MainPageInfo {width: 90%; margin: 0 auto;}
}

/*    Gallery    *****/
div.gallery {
    column-count: 4;
}

div.galleryImg {
    margin: 5px;
    border: 2px solid #BE0F34;
}

```



```

    float: left;

    width: 180px;
}

div.galleryImg:hover {
    border: 2px solid #70091F;
}

div.galleryImg img {
    width: 100%;
    height: auto;
}

div.desc {
    padding: 15px;
    text-align: center;
}

/*    Modal Login Form    *****/
.modal {
    display: none;
    position: fixed;
    z-index: 2;
    left: 0;
    top: 0;
    width: 100%;
    height: 100%;
    overflow: auto;
    background-color: rgb(0,0,0);
}

```

```
    background-color: rgba(0,0,0,0.4);
    padding-top: 60px;
}

.modal-content {
    background-color: #BE0F34;
    margin: 5% auto;
    border: 1px solid #888;
    width: 60%;
    height: 60%;
}

.close {
    position: absolute;
    right: 25px;
    top: 40px;
    color: #000;
    font-size: 35px;
    font-weight: bold;
}

.close:hover, .close:focus {
    color: red;
    cursor: pointer;
}

.animate {
    -webkit-animation: animatezoom 0.6s;
    animation: animatezoom 0.6s;
```

```

}

@-webkit-keyframes animatezoom {
    from {-webkit-transform: scale(0)}
    to {-webkit-transform: scale(1)}
}

@keyframes animatezoom {
    from {transform: scale(0)}
    to {transform: scale(1)}
}

/*      Error Handling      *****/
.pass_error_message {
    display: none;
    position: fixed;
    z-index: 2;
    color: black;
    background-color: white;
    text-align: center;
    left: 50%;
    top: 30%;
    width: 40%;
    border: 3px solid black;
    padding: 10px;
    transform: translate(-50%, -50%);
}

.passwordsnomatch {

```

```
display: none;
position: fixed;
z-index: 2;
color: black;
background-color: white;
text-align: center;
left: 50%;
top: 30%;
width: 40%;
border: 3px solid black;
padding: 10px;
transform: translate(-50%, -50%);
}
```

```
.passsuccess {
display: none;
position: fixed;
z-index: 2;
color: black;
background-color: white;
text-align: center;
left: 50%;
top: 30%;
width: 40%;
border: 3px solid black;
padding: 10px;
transform: translate(-50%, -50%);
}
```

```
.emailsnomatch {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;  
    padding: 10px;  
    transform: translate(-50%, -50%);  
}
```

```
.emailsuccess {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;  
    padding: 10px;  
    transform: translate(-50%, -50%);  
}
```

```
.unamesnomatch {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;  
    padding: 10px;  
    transform: translate(-50%, -50%);  
}
```

```
.unamesuccess {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;  
    padding: 10px;  
    transform: translate(-50%, -50%);
```

```
}
```

```
.requestalreadysent {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;  
    padding: 10px;  
    transform: translate(-50%, -50%);  
}
```

```
.registrationapproved {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;  
    padding: 10px;
```

```
    transform: translate(-50%, -50%);  
}
```

```
.registrationdenied {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;  
    padding: 10px;  
    transform: translate(-50%, -50%);  
}
```

```
.useralreadyhasaccess {  
    display: none;  
    position: fixed;  
    z-index: 2;  
    color: black;  
    background-color: white;  
    text-align: center;  
    left: 50%;  
    top: 30%;  
    width: 40%;  
    border: 3px solid black;
```



```
padding: 10px;
transform: translate(-50%, -50%);
}
```

```
.something {
display: none;
position: fixed;
z-index: 2;
color: black;
background-color: white;
text-align: center;
left: 50%;
top: 30%;
width: 40%;
border: 3px solid black;
padding: 10px;
transform: translate(-50%, -50%);
}
```

```
.you_must_login {
display: none;
position: fixed;
z-index: 1;
left: 0;
top: 0;
width: 100%;
height: 100%;
overflow: auto;
background-color: rgb(0,0,0);
}
```

```

    background-color: rgba(0,0,0,0.4);
    padding-top: 60px;
}

.you_must_login_content {
    display: none;
    position: fixed;
    z-index: 2;
    color: black;
    background-color: white;
    text-align: center;
    left: 50%;
    top: 30%;
    width: 40%;
    border: 3px solid black;
    padding: 10px;
    transform: translate(-50%, -50%);
}

/* Database Table *****/
.data_table {
    outline: 1px solid darkslategrey;
    text-align: center;
    font-size: 14px;
}

.data_title {
    font-size: 18px;
    text-align: left;
}

```

```

    font-weight: bold;

    outline: 1px solid darkslategrey;
}

/*    Location Page    *****/
.odometer {
    font-family: digital;
    color: white;
    background-color: darkslategrey;
    padding: 2px;
}

/*    Custom Search Box    *****/
input.gsc-input {
    border-color: #004220;
    width:100px;
}

input.gsc-search-button {
    border-color: #004220;
    background-color: #004220;
    width:10px;
}

aa {
    padding: 12px;
    margin: 20px;
    width: 80%;
}

```

registration.php

When a user registers for access, the browser redirects to this page, which checks for the user information in the database before storing the information for review by an administrator.

```
<?php
    session_start();

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "registration";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username", "$password") or
die("Cannot connect");
mysqli_select_db($connection, "$db_name") or die("cannot select
DB");

    // get information from $_POST
    $firstname=$_POST["fname"];
    $lastname=$_POST["lname"];
    $email=$_POST["email"];
    $message=$_POST["message"];

    // To protect MySQL injection
    $firstname = stripslashes($firstname);
    $lastname = stripslashes($lastname);
```

```

$email = stripslashes($email);
$message = stripslashes($message);

$firstname = mysqli_real_escape_string($connection, $firstname);
$lastname = mysqli_real_escape_string($connection, $lastname);
$email = mysqli_real_escape_string($connection, $email);
$message = mysqli_real_escape_string($connection, $message);

// Create Query
$count = 0;

$sql2 = "SELECT * from $tbl_name WHERE email='$email'";
$result = mysqli_query($connection, $sql2);
$count = mysqli_num_rows($result);

if ($count > 0)
{
    $_SESSION['errors'] = "requestalready sent";
    header("location: settings.php");
    exit;
}

$sql = "INSERT INTO $tbl_name SET first_name='$firstname',
    last_name='$lastname', email='$email', message='$message',
    denied=0";

$result = mysqli_query($connection, $sql);

$sql2 = "SELECT * from $tbl_name WHERE first_name='$firstname' and
    last_name='$lastname' and email='$email'";

```

```
$result = mysqli_query($connection, $sql2);  
$count = mysqli_num_rows($result);  
  
if ($count == 1)  
{  
    $_SESSION['errors'] = "registrationsuccess";  
    header("location:settings.php");  
    exit;  
}  
else  
{  
    $_SESSION['errors'] = "somethingwentwrong";  
    header("location:settings.php");  
    exit;  
}  
?>
```

adduser.php

When an administrator adds a new user, whether through the approve/deny buttons or by manually entering them in, the browser redirects to this page. This moves the user from the registration page (if they existed there) to the login page and gives them access to the portal.

```
<?php
```

```
    session_start();

    // Database connection information
    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "login";
    $registration = "registration";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username", "$password") or
        die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot select
        DB");

    // get information from $_POST
    $email=$_POST["email"];
    $fname=$_POST["fname"];
    $lname=$_POST["lname"];

    // To protect MySQL injection
```

```

$email = stripslashes($email);
$fname = stripslashes($fname);
$lname = stripslashes($lname);
$email = mysqli_real_escape_string($connection, $email);
$fname = mysqli_real_escape_string($connection, $fname);
$lname = mysqli_real_escape_string($connection, $lname);

// Create Query
$count = 0;

// Check if database contains the user being added in the
registration table
$sql2 = "SELECT * from $registration WHERE email='$email' and
        denied=0";
$result = mysqli_query($connection, $sql2);
$count = mysqli_num_rows($result);

// If the user does exist, change denied to 1 so they won't show
up on the settings page
if($count == 1)
{
    $sql = "UPDATE $registration SET denied=1 WHERE
            email='$email'";
    $result = mysqli_query($connection, $sql);

    $sql2 = "SELECT * FROM $registration WHERE email='$email' and
            denied=1";
    $result = mysqli_query($connection, $sql2);
    $count = mysqli_num_rows($result);
}

```



```

// If the user was found, but the denied field could not be
edited, throw an error.
if($count != 1)
{
    $_SESSION['errors'] = "somethingwentwrong";
    header("location:settings.php");
    exit;
}
}

// If the user was not found add them to the registration table so
they cannot register again.
else
{
    $sql = "INSERT INTO $registration SET email='$email',
        first_name='$fname', last_name='$lname', denied=1";
    $result = mysqli_query($connection, $sql);

    $sql2 = "SELECT * FROM $registration WHERE email='$email' and
        denied=1";
    $result = mysqli_query($connection, $sql2);
    $count = mysqli_num_rows($result);

    if($count != 1)
    {
        $_SESSION['errors'] = "somethingwentwrong";
        header("location:settings.php");
        exit;
    }
}

```

```

    }
}

// Check to make sure the user does not already have access to the
database
$sql2 = "SELECT * from $tbl_name WHERE email='$email'";
$result = mysqli_query($connection, $sql2);
$count = mysqli_num_rows($result);

// If the user already has access, throw an error.
if($count == 1)
{
    $_SESSION['errors'] = "useralreadyhasaccess";
    $_SESSION['denieduser'] = $fname." ".$lname;
    header("location:settings.php");
    exit;
}

// Create a username and password for new user
$username = $email;
$password = RandomString();
$admin = 0;

// Set Administrator field
if(isset($_POST['admin']))
{
    if($_POST['admin'] == "yes")
    {
        $admin = 1;
    }
}

```

```

    }
}

// Create and run SQL code to add user to the login table
$sql = "INSERT INTO $tbl_name SET username='$username',
    password='$password', f_name='$fname', l_name='$lname',
    email='$email', administrator='$admin'";
$result = mysqli_query($connection, $sql);

// Verify that the user has been added
$sql2 = "SELECT * FROM $tbl_name WHERE username='$username'";
$result = mysqli_query($connection, $sql2);
$count = mysqli_num_rows($result);

// If the user was successfully added display a message on the
Settings page
if($count > 0)
{
    $_SESSION['errors'] = "registrationapproved";
    $_SESSION['approveduser'] = $fname." ".$lname;

// the email function has been finished but does not yet work
because I can't get an email server set up on this computer.
once the system has been migrated to a real server, it will be
reinstated

// $message = "<html><body>Congratulations ".$fname.",\n\nYou have
been granted access to the <a href='localhost/portal'>NNU Space
Research Portal</a>. Use your credentials below to log into the
system. Click <a href='localhost/portal/settings.php'>Here</a>

```

```

to change your username and password.\n\nUsername:
".$username."\nPassword: ".$password."\n\nIf you have any
questions, please contact jpoundstone@nnu.edu</body></html>";

// $headers = "MIME-Version: 1.0"."Content-
    type:text/html;charset=UTF-8"."\\r\\n"."\\r\\n"."From:
    jpoundstone@nnu.edu"."\\r\\n"."CC: jpoundstone@nnu.edu";

// mail($email, "Access to NNU Space Research Portal",
    $message, $headers);

header('location:settings.php');
exit;
}
else //If the user was not added, throw an error
{
    $_SESSION['errors'] = "somethingwentwrong";
    header("location:settings.php");
    exit;
}

// Random string function used to create a random password for new
users
function RandomString()
{
    $characters =
        '0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ
        YZ';
    $randstring = '';

```

```
for ($i = 0; $i < 10; $i++) {  
    $randstring .= $characters[rand(0, strlen($characters))];  
}  
return $randstring;  
}  
?>
```

deny.php

If a user is denied, this page sets a denied flag so that their information does not continue to show on the settings page for administrators.

```
<?php
```

```
    session_start();

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "registration";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username", "$password") or
        die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot select
        DB");

    // get information from $_POST
    $email=$_POST["email"];
    $fname=$_POST["fname"];
    $lname=$_POST["lname"];

    // To protect MySQL injection
    $email = stripslashes($email);
    $fname = stripslashes($fname);
    $lname = stripslashes($lname);
    $email = mysqli_real_escape_string($connection, $email);
```

```

$fname = mysqli_real_escape_string($connection, $fname);
$lname = mysqli_real_escape_string($connection, $lname);

// Create Query
$count = 0;

$sql = "UPDATE $tbl_name SET denied=1 WHERE email='$email'";
$result = mysqli_query($connection, $sql);

$sql2 = "SELECT * from $tbl_name WHERE email='$email' and
        denied=1";
$result = mysqli_query($connection, $sql2);
$count = mysqli_num_rows($result);

if($count == 1)
{
    $_SESSION['errors'] = "registrationdenied";
    $_SESSION['denieduser'] = $fname." ".$lname;
    header("location:settings.php");
    exit;
}

else
{
    $_SESSION['errors'] = "somethingwentwrong";
    header("location:settings.php");
    exit;
}
?>

```

changeusername.php

Using the change username function on the settings page redirects to this page which connects to the database and updates the username field in the login table.

```
<?php
    session_start();

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "login";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username", "$password") or
        die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot select
        DB");

    // get information from $_POST
    $username=$_POST["uname"];
    $password=$_POST["pass"];
    $newuser =$_POST["nuser"];
    $confuser=$_POST["ruser"];

    // To protect MySQL injection
    $username = stripslashes($username);
    $password = stripslashes($password);
    $newuser = stripslashes($newuser );
```



```

$confuser = stripslashes($confuser);

if ($newemail !== $confmail)
{
    $_SESSION['errors'] = "unamesnomatch";
    header("location:settings.php");
    exit;
}

$username = mysqli_real_escape_string($connection, $username);
$password = mysqli_real_escape_string($connection, $password);
$newuser = mysqli_real_escape_string($connection, $newuser );

// Create Query
$count = 0;

$sql = "SELECT * FROM $tbl_name WHERE username='$username' and
        password='$password'";
$result = mysqli_query($connection, $sql);

$count = mysqli_num_rows($result);

if($count == 1)
{
    $sql2 = "UPDATE $tbl_name SET username='$newuser' WHERE
            username='$username'";
    $result = mysqli_query($connection, $sql2);
}

```

```
$sql = "SELECT * FROM $tbl_name WHERE username='$newuser' and
        password='$password'";
$result = mysqli_query($connection, $sql);
$count = mysqli_num_rows($result);

if($count == 1)
{
    $_SESSION['errors'] = "unamesuccess";
    header("location:settings.php");
    exit;
}

else
{
    $_SESSION['errors'] = "somethingwentwrong";
    header("location:settings.php");
    exit;
}
?>
```

changePASS.php

This page works exactly the same as changeusername.php except it updates the password field rather than the username.

```
<?php
```

```
    session_start();

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "login";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username", "$password") or
        die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot select
        DB");

    // get information from $_POST
    $username=$_POST["uname"];
    $password=$_POST["cpass"];
    $newpass =$_POST["npass"];
    $confpass=$_POST["rpass"];

    // To protect MySQL injection
    $username = stripslashes($username);
    $password = stripslashes($password);
    $newpass = stripslashes($newpass);
```

```

$confpass = stripslashes($confpass);

if ($newpass !== $confpass)
{
    $_SESSION['errors'] = "passwordsnomatch";
    header("location:settings.php");
    exit;
}

$username = mysqli_real_escape_string($connection, $username);
$password = mysqli_real_escape_string($connection, $password);
$newpass  = mysqli_real_escape_string($connection, $newpass );

// Create Query
$count = 0;

$sql = "SELECT * FROM $tbl_name WHERE username='$username' and
        password='$password'";
$result = mysqli_query($connection, $sql);

$count = mysqli_num_rows($result);

if($count == 1)
{
    $sql2 = "UPDATE $tbl_name SET password='$newpass' WHERE
            username='$username'";
    $result = mysqli_query($connection, $sql2);
}

```

```
$sql = "SELECT * FROM $tbl_name WHERE username='$username' and
        password='$newpass'";
$result = mysqli_query($connection, $sql);

if($count == 1)
{
    $_SESSION['errors'] = "passsuccess";
    header("location:settings.php");
    exit;
}

else
{
    $_SESSION['errors'] = "somethingwentwrong";
    header("location:settings.php");
    exit;
}
?>
```

changeemail.php

This page again works the same as changeusername.php and changepass.php, but updates the email field.

```
<?php
```

```
    session_start();

    $host = "localhost";
    $username = "root";
    $password = "";
    $db_name = "nnu_space_research";
    $tbl_name = "login";

    // Connect to server and select database
    $connection = mysqli_connect("$host", "$username", "$password") or
        die("Cannot connect");
    mysqli_select_db($connection, "$db_name") or die("cannot select
        DB");

    // get information from $_POST
    $username=$_POST["uname"];
    $password=$_POST["pass"];
    $newemail=$_POST["nmail"];
    $confmail=$_POST["rmail"];

    // To protect MySQL injection
    $username = stripslashes($username);
    $password = stripslashes($password);
    $newemail = stripslashes($newemail);
```

```

$confmail = stripslashes($confmail);

if ($newemail !== $confmail)
{
    $_SESSION['errors'] = "emailsnomatch";
    header("location:settings.php");
    exit;
}

$username = mysqli_real_escape_string($connection, $username);
$password = mysqli_real_escape_string($connection, $password);
$newemail = mysqli_real_escape_string($connection, $newemail);

// Create Query
$count = 0;

$sql = "SELECT * FROM $tbl_name WHERE username='$username' and
        password='$password'";
$result = mysqli_query($connection, $sql);

$count = mysqli_num_rows($result);

if($count == 1)
{
    $sql2 = "UPDATE $tbl_name SET email='$newemail' WHERE
            username='$username'";
    $result = mysqli_query($connection, $sql2);
}

```

```
$sql = "SELECT * FROM $tbl_name WHERE username='$username' and
        password='$password'";
$result = mysqli_query($connection, $sql);

if($count == 1)
{
    $_SESSION['errors'] = "emailsucces";
    header("location:settings.php");
    exit;
}

else
{
    $_SESSION['errors'] = "somethingwentwrong";
    header("location:settings.php");
    exit;
}
?>
```