



Day 3: Team Checkins



<https://hackhpc.github.io/sgx3admi24>

File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat Team_Name.txt
```

```
NLC^2
```

```
root@hack2024:~/NLC^2 $ ls Team_Mentors
```

```
'Teniola Oluwaseyitan'
```

```
root@hack2024:~/NLC^2 $ ls Team_Members
```

```
'Chandler Campbell' 'Christian Johnson' 'Lisha Ramon' 'Nole Stites'
```

```
root@hack2024:~/NLC^2 $ cat Team_Theme_Song.mp3
```

```
'https://uppbeat.io/Melifluous-Mirage'
```





File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat Project_Problem
```

Many institutions don't have an easy way to categorize or present their training resources. Currently, HPC-ED utilizes a command-line interface (CLI) to add training material to and query data from a database which is not at all user friendly or intuitive. Many people don't know how to use a CLI, so they don't get the opportunity to use the institution's training resources.

Furthermore, most people go to Google for their needs which isn't the best way to search. A given Google query might return thousands of results, making it hard to know which ones are worth looking at because the quality of sources vary.

```
:wq
```



File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat HPC-ED_Gateway
```

Our Targeted Science Gateway is the HPC-ED Gateway. HPC-ED (High-Performance Computing - Education) is a project to create and share metadata for HPC educational materials, making it easier to discover, access, and publish these resources through a federated catalog system.

```
:wq
```



File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat Project_Goals
```

1. Create a user-friendly website template for institutions to store and query training resources that don't use a CLI
2. Connect a database to the website that stores the training resources and supports the CRUD operations
3. Allow a user to download a JSON file for a given training resource returned from a search query

```
:wq
```



```
root@hack2024:~/NLC^2 $ cat Potential_Pitfalls_and_Bottlenecks
```

1. Time: only having a week to complete our desired goals can cause stress and anxiety
2. Gold plating: getting too distracted by the finer details at the beginning can prohibit us from getting important work done
3. Test data: getting enough data to test the corner and edge cases might be difficult



File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat Project_Plan_Delivery
```

Deliverables

Github Lead: [Nole Stites](#)

Presentation Lead: [Christian Johnson](#)

Poster Lead: [Lisha Ramon](#)

Code Lead: [Chandler Campbell](#)

:wq



```
root@hack2024:~/NLC^2 $ cat Project_Breakdown_Website
```

Website Pages:

- Home: Initial presentation, connects other pages via links
 - Guides how to utilize template
- Search: User will query the database. May have the option of applying filters for specific results.
- Admin: Accessed by persons with privileges to perform CRUD [Create Read Update Delete] operations on database material.
- About: Provides info about the vision of the mission to achieve

```
:wq
```




File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat Project_Breakdown_Database
```

Database:

- Utilize Global Search API: Stores all data, set privileges on data visibility, as well as retrieve data through search queries.
- Rather than creating a DB, utilize a resource that provides one for us(HPC-ED API)
- (For group) Read up on Github's documentation for open community:
 - github.com/readthedocs/readthedocs.org
 - readthedocs.org
 - docs.globus.org/api/search/

:wq



File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat Project_Breakdown_Backend_Code
```

Backend Code:

- Bridges between the database (DB) and frontend website
- User submits a search query to backend code -> Accesses DB via API
- DB returns query results to backend code -> Displays back on website via API return

:wq



File Edit View Search Terminal Help

```
root@hack2024:~/NLC^2 $ cat Project_Ideas
```

- Relevant and meaningful search results
 - Generate summaries of data and order results in a meaningful priority
- Smart/predefined search features
 - Show results published from a user's institution or about their area of expertise
- Dynamically generated list of search filters
 - Create filter options based on the metadata in the database entries
- Google authentication to allow for admin users and standard users
 - Restrict admin actions like adding/removing database entries to specific users.
- Suggest resources to add to the database
 - AI resource suggestions based on a given text

```
:wq
```



File Edit View Search Terminal Help

root@hack2024:~/NLC^2 \$ cat Day_3_Morning_Check-in

Project Goal(s):

- Create a website to interact the training resources
- Set up Globus Search API to interact with the HPC-ED database
- Make thorough documentation (README, poster, etc.)

Current Status:

- Goal 1 Update:
 - Began designing a wireframe for the website
 - Coded HTML intake forms and implemented Google Authentication (debugging at present)
- Goal 2 Update: Got the Globus Search Python SDK working; we just need to connect it to the backend code
- Goal 3: Update: Began gathering resources for the poster and will begin the README shortly

Next Steps:

- **Have a simple working frontend and get it connected to the HPC-ED database**

Issues/Concerns:

- Time constraint to meet goals, connecting our individual work and have it smoothly function as one piece



Team Name: Data Detectives



- Team Mentor(s): Reggie Kelley
- Team Members: Jean-Dominique Anoh, Paris Coleman, Dickson Acheampong, Jace Lespinasse
- Team Theme Song Name: (FREE FOR PROFIT) Erykah Badu x Jazz x Neo Soul Type Beat - "SPRING"

Link to the theme song:

https://www.youtube.com/watch?v=Jmzk_QqnC3A



Goals

Potential
Pitfalls/Bottlenecks:
Learning/understanding
python coding for data

Target Science Gateway: HPC-ED Metadata

Issue to be addressed: HPC-ED wants to create a database to ensure an easier learning environment for HPC (problem 3)

Project Goal(s):

1. Create flask app to digest information off websites
2. Develop means to transfer information from sites to the app
3. Transfer information from app to database



Project Plan

Deliverables

Github Lead: Jace Lespinasse

Presentation Lead: Jean Anoh

Poster Lead: Paris Coleman

Code Lead: Dickson

Acheampong

Project Plan

Goal 1: Create flask database to digest information off websites

- Task 1: Make new project in Eureka/Jupiter
- Task 2: Develop flask app
- Task 3: Identify which websites to use

Goal 2: Develop means to transfer information from sites to the app

- Task 1: Develop HTML file to connect to websites
- Task 2: Add validation to ensure data quality
- Task 3: Establish the form submission endpoint

Goal 3: Transfer information from app to HPC-ED database

- Task 1: Ensure the data is easily user readable
- Task 2: Handle and store incoming form data
- Task 3: Transmit stored data to the HPC portal



Day 3: Morning Check In Data Detectives

Project Goal(s):

- Create flask database to digest information off websites
- Develop web scraper to strip text from website and store within flask app
- Transfer output to json format

Current Status:

- Goal 1 Update: Finished establishing the app
- Goal 2 Update: Started development and identified website that were going to be using
- Goal 3: Update: Not yet worked on

Next Steps:

- Continuous development of the web scraper as well as establish form submission endpoint

Issues/Concerns:

- Learning new material within constricted time frame



Team Name: Pandora's Programmers

-Team Mentor(s): Charlie

- Team Members: LeahMonet Morgan, Catalina Tovar, Chris Henry, Jesutofarati Ajala

- Team Theme Song Name: Greek Mythology Music - Pandora's Box

Link to the theme song:

<https://youtu.be/s4nt8M85pyE?si=7tXLEFQDYB-vhohD>



Goals

Target Science Gateway: HPC-ED

Issue to be addressed: Choice 3

Project Goal(s):

1. Set Up Flask Project
2. Create the Data Entry Form
3. Use flask to create a webform to store the websites and push them to the HPC-ED website

Potential Pitfalls/Bottlenecks: A lot of our strengths and weaknesses are similar as a group.



Project Plan

Project Plan

Deliverables

Github Lead: Christian Henry

Presentation Lead: Leah Monet

Morgan

Poster Lead: Catalina Tovar

Code Lead: Jesutofarati Ajala

Goal 1: Set up flask project

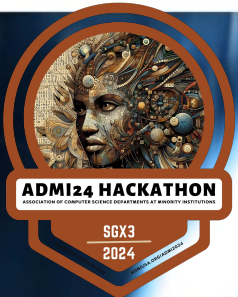
- Task 1- Create project directory
- Task 2- Initialize flask application
- Task 3- Configure routes

Goal 2: Create data entry form

- Task 1- Design HTML form
- Task 2- Style the form
- Task 3- Implement checks to insure data meets criteria

Goal 3: Store and push data to HPC-ED website

- Task 1- Set up form submission route
- Task 2- Process and store form data
- Task 3- Push data to HPC-ED website



Day 3: Morning Checkin Pandora's Programmers

Project Goal(s):

- Goal 1: *Set up flask project*
- Goal 2: *Create data entry form*
- Goal 3: *Store and push data to HPC-ED website*

Current Status:

- Goal 1 Update: We currently are in the process of establishing the functional flask application to ensure that we have a working environment for development.
- Goal 2 Update: Have not assigned anyone goal to at the moment, we will be talking about this task today.
- Goal 3 Update: Have not assigned anyone goal to at the moment, we will be talking about this task today.

Next Steps:

- **Sitting down, really talking through all the task and how we want to go about attacking them.**

Issues/Concerns:

- As I said in our goals slide, we all have similar skills, the task for goal 2 and 3 is still pretty new to all of us. So we just have to talk and figure out a way to go about completing them.



Team Name

- Team Mentor(s)
- Team Members

- Team Theme Song Name
Link to the theme song



Note:

Feel free to change the look of the slide just dont change the theme because that will change everyone else's too!