Digital Backpack User Manual

May 4th, 2022

**Team Logout**

****

Sponsored by Dr. Morgan Vigil-Hayes

Community Aware Networks & Information Systems

**Team Members:**

Volodymyr Saruta – Team Mentor

Nicholas Caporusso – Team Leader

Kevin Kilbourne

Jonathan Nation

Jacob Novin

**Table of Contents**

[**Introduction**](#_7fkplkbgdzxq) **3**

[**Installation**](#_bhgxzbmfb310) **4**

[**Plugging in Key Features**](#_lj7xqwyzczpm) **5**

[**Configuration and Daily Operation**](#_y4rn2brosoey) **8**

[**Maintenance**](#_9h9zxnlhd3m0) **18**

[**Trouble-shooting**](#_mfw1rpnkrwjb) **19**

[**Conclusion**](#_6vl2jj2piek0) **20**

# Introduction

We are pleased that you have chosen the Digital Backpack for your business needs. The Digital Backpack is offered as a web application. It is a helpful system that allows a fluid transition between online and offline learning which has been custom-designed to meet your needs.

User management needed to fully utilize the Digital Backpack:

* Teacher user registration
* Teacher user login
* Student user registration
* Student user login

Some of the key features for the web application includes:

* Fetching and Downloading Algorithm
* A Rating system
* A Flagging system
* An Online Connectivity Heatmap

The purpose of this user manual is to help you, the client, successfully install, administer, and maintain the Digital Backpack product within your actual business context in the future. We hope that the project will aid future generation’s education. The document will talk about the installation process, configuration, maintenance, and troubleshooting.

# Installation

Over time, if it is necessary, you may want to move the product to a new platform or reinstall the product. This section discusses the installation process for the Digital Backpack. There are two major components of installation for the product. This is divided into the Django server installation and the android application installation. The progressive web app is included in the Django server.

**Django Server Setup**

The Django server was originally hosted on a DigitalOcean Droplet, however, it should be easily compatible with other cloud hosting services as well. As previously mentioned, the progressive web app is implemented within the Django server, thus the installation process incorporates both of these components. The server was developed with Django version 4.0.4 and Python version 3.9.10 Steps for both Windows and Linux installation are as follows:

**Windows Installation**

1. SSH into your desired web server - Once this is done you can create your desired directories and drag the Django project files into this directory.
2. Create a Python virtual environment - Ensure you are in the directory with the Django project files. Create a python virtual environment by entering the command:

**> python -m venv [virtualenv name]**

1. Activate your virtual environment - Enter the virtual environment you just created by entering the following command:

**> [virtualenv name]\Scripts\activate.bat**

1. Install requirements - A requirements.txt file is included within the Django server files. Install the requirements in the virtual environment using the following command:

**> pip install -r requirements.txt**

1. Make migrations - The student database is handled using Django models. Make migrations to initialize the database with the following commands:

**> python manage.py makemigrations**

**> python manage.py migrate**

1. Run the server - At this point, the installation process for Django is complete. The final step is to run the server. This can be done by running the following command:

**> python manage.py runserver 127.0.0.1:8000**

**Linux Installation**

1. SSH into your desired web server - Once this is done you can create your desired directories and drag the Django project files into this directory.
2. Create a Python virtual environment - Ensure you are in the directory with the Django project files. Create a python virtual environment by entering the command:

**$ python3 -m venv [virtualenv name]**

1. Enter your virtual environment - Enter the virtual environment you just created by entering the following command:

**$ source [virtualenv name]/bin/activate**

1. Install requirements - A requirements.txt file is included within the Django server files. Install the requirements in the virtual environment using the following command:

**$ pip3 install -r requirements.txt**

1. Make migrations - The student database is handled using Django models. Make migrations to initialize the database with the following commands:

**$ python3 manage.py makemigrations**

**$ python3 manage.py migrate**

1. Run the server - At this point, the installation process for Django is complete. The final step is to run the server. This can be done by running the following command:

**$ python3 manage.py runserver 127.0.0.1:8000**

# Plugging in Key Features

***Note:* More documentation about the specific function’s functionality can be found within the source code.**

Using the Django web framework, modularity is very commonly found. A large point of this project was to provide key features and new tools that can easily be implemented elsewhere on other systems. With that being said, the following directions will navigate you toward plugging in these features into other areas within the Django workspace. There is nothing extra needed to do if your are just cloning the GitHub repository.

**Searching Algorithm**

There are two sections of the Searching Algorithm that need to be covered. The first is the retrieval of the web resources upon assignment creation and the other is downloading the web resources as PDFs to the Student user’s computer.

1. **Retrieving Web Resources -** This is utilized by Teacher users when they create a new assignment for their class. The keywords are grabbed from a text field’s POST request in the New Assignment page, triggering the new\_assignment view function. These keywords are then passed into the function searchingAlgorithm(keywords) found in the utils.py file. Within the utils.py function, the websites are grabbed and a list of the 10 websites are returned.
2. **Downloading Web Resources -** This is utilized by Student users once they submit an assignment with web resources attached. This operation, like the retrieving web resources process, is also located in utils.py called in a function named downloadWebsites(links, path). The function is called within the submit\_new\_assignment() view function.

As you can see, for both of these processes, they are both reliant upon the views functions alongside the utils.py functions.

**Web Resource Rating**

The web resource rating functionality comes from a views.py function, called ratings(), and a model held within models.py named submitRatings(). Upon a user landing on the web resource ratings page, the views function previously mentioned is called. Within that view, the ratings model is instantiated with the data provided from the user’s POST request through the webpage. With all of that being said, the only things needed to keep track of in this accumulation of ratings process are the views.py function and the models.py function. From these, the necessary models and inputs into the database are made accordingly.

**Student Flagging**

Student flagging is not too complex of an operation. The main logic for this task is handled through a function called swapFlag() in a JavaScript file named studentFlag.js. In our Django server, this can be located at ‘DigitalBackpack/static/js/studentFlag.js’. The Student Account webpage, rendered with viewstudent.html, interacts with this JavaScript file with the use of a button. This JavaScript file would just need to be linked into the HTML file using the <script> tag and have a button call the swapFlag() function with its onclick=”swapFlag(event, this)” attribute.

**Online Connectivity Heatmap**

The core of the Online Connectivity Heatmap is primarily based inside of the utils.py file. This is in tandem with the calling of the Heatmap functions residing in the student\_page() views.py function. Within utils.py, the main components can be broken up into 3 different functions:

* **makeHeatmap(username) -** Called from the student\_page() view. This is the overarching ‘main’ function of the Heatmap creation process. This function calls the two other helper functions (seen below) and facilitates the process, gathering time data, opening necessary files, and construction of the Heatmap itself. Does not return any value, but instead just saves the newly created Heatmap in the correct file directory.
  + **‘username’ parameter -** Passed into the function form inside of the student\_page() view, this variable holds the Student model that is currently logged in and using the system.
* **updateWorkingHeatmap() -** Using the current time data collected, this function takes the date and time and inputs it into the corresponding placement within the specific student’s working Heatmap .csv file.
* **transferWorkingData(working\_array, display\_array, result\_array) -** Using the 3 data arrays, this function adds the values recently collected from the working\_array to the more long-term values in the display\_array. The result of this is addition is input into the result\_array which is, ultimately, returned.
  + **‘working\_array’ parameter -** This is the array that actively collects the time information. After one connection is detected, this is reset back to empty.
  + **‘display\_array’ parameter -** This is the array that actually displays on the Student Account page. Its values are collectively growing, not being reset after every connection.
  + **‘result\_array’ parameter -** The sum of working\_array’s data and display\_array’s data is added into this array. This is the array that the function returns.

The only process that is affected when using a different backend is the actual time collection. The system is reliant upon Django’s student\_page() view function, which executes every time a user loads the webpage. Without the Django view function, all that has to be configured is how the system will know when a Student user lands on their Student homepage. Once this is figured out, all that is needed is calling makeHeatmap().

# Configuration and Daily Operation

Once the Django server and all necessary packages have been successfully installed and running smoothly, you can get into actually creating both Student and Teacher accounts to fully use the system! We will be running through the steps to fully utilize the web application in the order that is necessary for successful operation. The operating process of the Digital Backpack begins with setting up an account for teachers.

**Running the Server**

After the installation process has been completed, getting the server to actually start running is not difficult. The following steps outline what is needed to begin running the server.

1. Maneuver to the directory containing all of the server code:
   1. **cd “path to server files”**
2. Start the server using the Django runserver command:
   1. **python3 manage.py runserver**

**General Site Maneuverability**

The Digital Backpack is not out of the ordinary when it comes to making your way around the website. There are a couple of points of access that will be explained here in order to promote successful and fluid maneuverability of the site.

* You are able to click “DigitalBackpack” within the website’s header to return to the Digital Backpack landing page. From this landing page, you are able to choose the user you wish to continue with, either login with that user or create a new one.
* You are able to click “Register” within the website’s header to have a drop down menu of which user you would wish to register as. This will redirect to the registration page of the selected user (../register/teacher/ or ../register/student/).
* You are able to click “Login” within the website’s header to have a drop down menu of which user you would wish to login as. This will redirect to the login page of the selected user (../login/teacher/ or ../login/student/).
* You are able to click “Logout” within the website’s header to successfully log out of the user logged in with. The result of this will redirect to the Digital Backpack landing page.

**User Account Management**

**Create New Teacher User Account**

***Note:* With a fresh database with no created accounts (student or teacher) yet, it is imperative to make a Teacher account with at least one class set-up prior to any set-up of student accounts. This is because the Instructor has to set up the class workspace and register the Students they want (via their email) in their class before that specific student’s account is finalized. Further information provided in Step 6.**

1. Upon successful connection to the Digital Backpack landing page (URL with no trailing path), you are given the choice of creating a Student or Teacher account.
2. So, from the Digital Backpack landing page, click on the Teacher button to the right.
3. Upon clicking the Teacher button, you will then be redirected to the Teacher Login page (.../login/teacher/).
   1. If there is already a Teacher account made, enter the correct “Username” and “Password” credentials to the pertaining Teacher account.
   2. However, if you wish to create a new Teacher account, click the “Sign Up Today” link found in blue below username and password boxes. Continue on to step 4.
4. Upon clicking the “Sign Up Today” link, you will then be redirected to the Teacher Registration page (../register/teacher/). Within this page, you will be asked to set a username, email, and password to all be assigned and associated with the created Teacher account. A password confirmation is also required. After entering all of the required account details, click “Create Account” to move forward with the registration process. The account details should be submitted with the characteristics listed below:
   1. **Username:**
      1. 150 characters or fewer.
      2. Only contains letters, number digits, or @ / . / + / - / \_ characters.
   2. **Email Address:**
      1. Standard Email format is needed here.
      2. Ex. [johndoe@company.com](mailto:johndoe@company.com)
   3. **Password:**
      1. Your password can;t be too similar to your other personal information.
      2. Your password must contain at least 8 characters.
      3. Your password should not be a commonly used password.
      4. Your password can not be entirely numeric.
5. Once your chosen username, email, and password are set and you have clicked “Create Account”, you will be redirected to the Class Creation page. This is where your first class associated with your new Teacher account is made. Here, you are provided with 4 boxes to input the class information. These 4 boxes are for the teacher’s username, the teacher’s first name, the teacher’s last name, and lastly, the name of the class you wish to create. After you are finished entering the information you want, click the “Create Class” button to move on to the next step.
   1. **Teacher Username -** Provide the username of the Teacher account that was just created.
   2. **Instructor First Name -** This is the first name of the display name that will be shown to students.
   3. **Instructor Last Name -** This is the last name of the display name that will be shown to students.
   4. **Class Name -** This is the name of the class that you want displayed to all students. There is no specific format to follow here, however, keep in mind that this is the name of the class that students will see.
      1. Ex. “CS126 - Computer Science I”
      2. Ex. “Algebra II”
6. So far we have a Teacher account set-up alongside a Class to go with that Teacher. Now, let’s add some students into this newly created class. Here, you are able to select the created class in which you wish to add students to. Once the class you want is chosen, you can add the email addresses of the students that you want enrolled in the class. Enter a student email in the box provided following the same email format as stated above (Ex. [johndoe@company.com](mailto:johndoe@company.com)), and either click “Add Student” to continue enrolling more students into the class or click to “Submit Student List” to confirm the class roster and finalize the class creation process.
   1. ***Note:* If a student tries to register with an email before this step, they will not be allowed to follow through with their Student account creation until “Submit Student List” is clicked and the class creation is finalized. You are able to tell which student has created their account or not yet by the Student Account value displayed in the Class Roster in the Teacher homepage. If it displays the student’s first and last name in the class roster, they have successfully created their Student Account and nothing is needed. However, if the class roster displays the added student’s email address, that student has yet to set up their account.**

After entering all of the students needed in the class and clicking “Submit Student List”, you will then be redirected to the Teacher homepage of that newly created class. Congratulations! You have now successfully created a Teacher account with a Class and Class Roster associated with that account!

**Login to Teacher User Account**

1. This login process is simple with not a lot of steps needed. Let’s start from the Digital Backpack landing page. Remember that at any point in the webpage, you can click the “Logout” button to get to the landing page as well. So to start logging in, from the landing page, click the “Teacher” button. This will redirect you to the Teacher login page (../login/teacher/).
2. Now, enter your username and password that you assigned to the account upon account registration.
3. After you are sure that the correct account information was inputted for username and password, click “Log In”. You will then be redirected to the Teacher homepage (../teacher/) with the last class you had used pulled up.

**Create New Student User Account**

1. Upon successful connection to the Digital Backpack landing page (URL with no trailing path), you are given the choice of creating a Student or Teacher account.
2. So, from the Digital Backpack landing page, click on the Student button to the left.
3. Upon clicking the Student button, you will then be redirected to the Student Login page (.../login/student/).
   1. If there is already a Student account made, enter the correct “Username” and “Password” credentials to the pertaining Student account.
   2. However, if you wish to create a new Student account, click the “Sign Up Today” link found in blue below username and password boxes. Continue on to step 4.
4. Upon clicking the “Sign Up Today” link, you will then be redirected to the Student Registration page (../register/student/). Within this page, you will be asked to set a username, email, and password to all be assigned and associated with the created Student account. A password confirmation is also required. After entering all of the required account details, click “Create Account” to move forward with the registration process. The account details should be submitted with the characteristics listed below:
   1. **Username:**
      1. 150 characters or fewer.
      2. Only contains letters, number digits, or @ / . / + / - / \_ characters.
   2. **Email Address:**
      1. Standard Email format is needed here.
      2. Ex. [johndoe@company.com](mailto:johndoe@company.com)
   3. **Password:**
      1. Your password can;t be too similar to your other personal information.
      2. Your password must contain at least 8 characters.
      3. Your password should not be a commonly used password.
      4. Your password can not be entirely numeric.
5. After submitting the username, email, and password that you want associated with your Student account, it is time to put the final touches on the registration process. After clicking “Create Account”, you are redirected to the Student account completion page (../register/complete-student/). Here, enter the username that you had just previously set-up your account with and following that, enter your First Name and Last Name into the respective fields. Your first and last name entered here will be your display name in other areas of the site.
6. When all the account information is set up, click “Finish Account” and you are all set! You will then be redirected to the Student homepage (../student/).

**Login to Student User Account**

1. Again, this login process is simple with not a lot of steps needed. Let’s start from the Digital Backpack landing page. Remember that at any point in the webpage, you can click the “Logout” button to get to the landing page as well. So to start logging in, from the landing page, click the “Student” button. This will redirect you to the Student login page (../login/student/).
2. Now, enter your username and password that you assigned to the account upon account registration.
3. After you are sure that the correct account information was inputted for username and password, click “Log In”. You will then be redirected to the Student homepage (../student/).

**Utilizing User Features**

This section lays out how to access all of the different components available to both Student and teacher users. We will take a look at these features starting with the Teacher user accessible ones, then move on to ones accessible form Student users.

***Teacher Users Features:***

**Create New Class**

1. Since we are going through Teacher accessible features, let’s start out at the Teacher homepage (../teacher/). This is assuming that you have already created a Teacher account and log in or finish the Teacher account creation process and land here.
2. From the Teacher homepage, within the “Current Active Classes box”, you have the ability to click “Add New Class” to introduce a new class with associated class name and configured class roster.
3. From here, this is essentially the same process that a teacher user would go through after their initial account creation. With that being said, refer back to **User Account Management -> Create New Teacher User Account -> Step 5** as seen above. Follow these steps to successfully create your class and be redirected back to where you started, the Teacher homepage (../teacher/).

**Create New Assignment**

1. In the teacher homepage (../teacher/) you have the ability to create new assignments for your students. Again, this is assuming that you have already created a Teacher account and log in or finish the Teacher account creation process and land here.
2. From the Teacher homepage, within the “Active Assignments box”, you have the ability to click “Add New Assignment” to introduce a new assignment for the class you are located in.
3. After clicking “Add New Assignment”, you will then be redirected to the Create New Assignment page (../teacher/newAssignment/). Upon landing here, you will notice different fields that make up the assignment as a whole. These fields for assignment information are defined as such:
   1. **Class Choice -** This is a dropdown menu which displays all of the active classes that the logged in Teacher user oversees. Simply select which class you would like the assignment to fall under.
   2. **Assignment Title -** Every assignment needs a name! This is pretty self explanatory so just provide a meaningful name that pertains to the assignment.
   3. **Assignment Instructions -** This box allows for a general description or specific assignment instructions to be conveyed to the student.
   4. **Assignment Due Date -** When would you like this specific assignment to be due by? Input the date here in a MM/DD/YYYY format.
   5. **Assignment Attachment -** Any extraneous files that you would like to be contained with the assignment should be added here. Click “Choose File” in order to choose a file within your own computer’s file directory.
   6. **Search Keywords (Searching Algorithm) -** The final field here is utilized in order to search Google for any supplemental web resources that could help out the student completing the assignment. This section is completely optional, but has the potential to enrich the educational experience of students that much more. The values that should be input into the text box are your keywords which pertain to the assignment being created. For example, if the assignment wanting to be created is about learning Python for loops, the keyword input should look like this: “Python, for, loop, syntax, iterate”. The format for these keywords, as seen in the example, is all the keywords you wish to search Google with, separated by commas. Once the desired keywords are inputted and comma-seperated, click “Fetch Resources”. The system will begin to work, searching Google with the entered keywords, displaying the links to the first 10 websites found (excluding YouTube links). From here, you are free to click the links and make sure they are worthy, delete the specific links that you do not want included, or start up another search with another set of keywords.
4. After all of the assignment information is entered to your liking, click on the “Create Assignment” button to finalize the assignment creation process. Upon clicking “Create Assignment”, you will then be redirected back to the Teacher homepage.

**View Student Account Page**

1. In the teacher homepage (../teacher/) you have the ability to check out information about the students within your class.
2. Looking to the left of the page, you will notice a class roster of the pertaining class, holding all of the students who are enrolled in that class. In the class roster, each student’s name is able to be clicked. Upon clicking on a specific student's name, you will then be redirected to the Student Account page (../teacher/viewstudent/) for that selected student.
3. Once on the Student Account page, information pertaining to the student who was clicked is then shown to the Teacher user. The displayed information includes the student’s name, student’s email address, and the student’s flagged status.
4. Considering the student’s flagged status, this is actually something the teacher is able to manipulate. The ability to swap a student’s flag is provided by a button under the flagged status called “Swap Flag”. When clicked, it changes the boolean status of the student’s flagged status.
5. One other thing also able to be seen here is the student’s Online Connectivity heatmap. No interaction is needed with the Heatmap as it just gives a good indication of patterns within the selected student’s recent website connectivity.
6. Anytime you wish to return back to the Teacher homepage, refer down to the bottom-left of the webpage. Clicking the button “Return to Teacher Page” will then redirect you back to the Teacher homepage.

**Swap Student Flag**

1. Now that we have been to the Student Account page (../teacher/viewstudent), we can look at swapping the individual student’s swap flag. This is not a complex operation and only requires minimal actions.
2. Locate the “Swap Flag” button within the “Student Flagged Status” box. Clicking this button will switch the boolean value of the specific student’s flagged status, with flagging or unflagging them. The student model’s attribute within the database is then changed accordingly.
3. Again, click “Return to Teacher Page” when ready to go back to the Teacher homepage.

***Student Users Features:***

**View Myself Page**

1. Since we are going through Student accessible features, let’s start out at the Student homepage (../student/). This is assuming that you have already created a Student account and login.
2. From the Student homepage, you will notice a large button inside of the “My Connection Timeframes” box near the bottom of the page. Upon clicking this button, it will then redirect the Student user to the mentioned above Student Account page (../student/viewmyself/).
3. Once landed on the Student Account page from the Student homepage, you will notice that it is similar to the same page accessible by the Teacher as it displays the student’s name, student’s email address, and the student’s flagged status. (Student’s do not have the privilege to change their own flagged status).
4. Also on this page, the student’s Online Connectivity Heatmap is also available for viewing.
5. When you are all done with this page and ready to head back to the Student homepage, click “Return to Student Page” in the bottom-left of the web page to be redirected back.

**Rate Web Resources**

1. To utilize the resource rating feature, navigate to the ratings page (../ratings/). For most purposes, a student user will be redirected to the ratings page upon submission of an assignment, with the resources passed along. For purposes of modular use, pass a list of sites to the page via a GET request.
2. Once the page loads, you will be prompted with the provided list of web resources and a choice of rating between 1 and 5 stars for each
3. Rate each resource that you utilized with how helpful you feel it was in completion of your assignment. If you did not use a certain resource, you may simply skip it.
4. Once you’ve completed your ratings, click the submit resources button and be redirected back to your homepage.

# Maintenance

In order to keep the Digital Backpack up and successfully running, it is imperative to have access to the Digital Backpack GitHub repository. With the source code available to you, any under the hood changes can be easily manipulated and addressed. Below is link to the repository:

**GitHub Repository:** <https://github.com/jakenovin3/DigitalBackpack>

If for any reason the system’s database needs to be wiped clean of users, classes, etc. then the “db.sqlite3” file will have to be deleted. This will require the original migrations of the server’s database models to be done again. This process is outlined below.

**Removing the current database:**

* Run the following code in the terminal:

**rm db.sqlite3**

**Setting up a new database:**

* Run the following code in the terminal

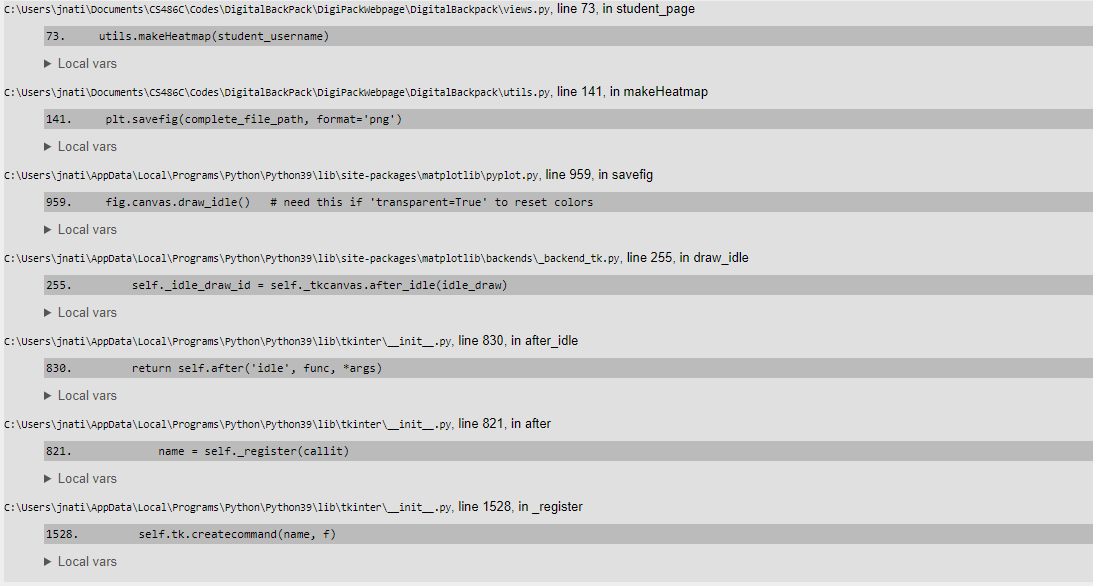
**python3 manage.py makemigrations**

**python3 manage.py migrate**

# Trouble-shooting

There was a single error that had popped up after we restarted the server. This was a “main thread is not in main loop” error upon a student user clicking on the “View my Connection” button to be taken to the Student Account page. Once on this page, quickly after clicking on “Return to Student Page” and repeating this many times, until the message pops up. The amount of times and conditions in which the error occurs are mildly inconsistent. This error does not show up too often and after leaving the page, everything works as it should.





# Conclusion

With best wishes from your Digital Backpack developers: Nick Caporusso, Jake Novin, Kevin Kilbourne, and Jon Nation. We thank you for allowing us to help the Digital Backpack vision come to life. It has been a great experience for the team to develop the application and work with you. While we are moving on to professional careers, we would be happy to answer short questions in the coming months to help you get the product deployed and operating optimally in your organization.

If you would like to contact us, our email addresses are:

Nick Caporusso: [nmc288@nau.edu](mailto:nmc288@nau.edu)

Jake Novin: [jon23@nau.edu](mailto:jon23@nau.edu)

Kevin Kilbourne: [kpk54@nau.edu](mailto:kpk54@nau.edu)

Jon Nation: [jdn238@nau.edu](mailto:jdn238@nau.edu)