

# Permission Number Distribution System

(Under Guidance - Haim Levkowitz, Acting Department Chair, Associate Professor  
Richard A. Miner School of Computer & Information Sciences, Kennedy College of Sciences, [haim@cs.uml.edu](mailto:haim@cs.uml.edu) )

Soham Prasad Kulkarni  
ID: 01985953

Department of Computer Science  
Richard A. Miner School of Computer &  
Information Sciences  
University of Massachusetts Lowell  
[SohamPrasad\\_Kulkarni@student.uml.edu](mailto:SohamPrasad_Kulkarni@student.uml.edu)

Akshay Kolli  
ID: 02069804

Department of Computer Science  
Richard A. Miner School of  
Computer & Information Sciences  
University of Massachusetts Lowell  
[Akshay\\_Kolli@student.uml.edu](mailto:Akshay_Kolli@student.uml.edu)

Naga Sai Tejaswi Gandu  
ID: 01987559

Department of Computer Science  
Richard A. Miner School of Computer &  
Information Sciences  
University of Massachusetts Lowell  
[NagaSaiTejaswi\\_Gandu@student.uml.edu](mailto:NagaSaiTejaswi_Gandu@student.uml.edu)

## PROBLEM ABSTRACT:

When students want to take a course at UML today, they have to first ask for permission to enroll in the form of a Permission number from the Professor. The student requests the professor via email, and if the professor accepts, He responds with a unique code that the student can use to join the class via UML SIS or UML NOW. This process is tedious, as there can be a long back and forth via email. The number of permission numbers the professor has to manage is not a manageable number either, and trying to keep track of the use of permission numbers can result in confusion on both the student's and the teacher's side.

## ABOUT THE PROJECT:

The goal of the project is to simplify the management and transfer of permission numbers for courses. It aims to do this by creating an intuitive interface and a robust backend that takes care of handling the permission numbers and requests for the permission numbers.

A permission number distribution system is a website where professors and students can better communicate and manage permission numbers and keep track of requests for permission numbers. It is a helpful tool for professors to manage permission numbers and incoming requests, especially when teaching a large class. It simplifies the professor's task of issuing permission numbers.

On the Student's login page, the students can create a profile, view the professors' classes, and request permission numbers for said classes. They can also check on the status of their request for permission numbers.

The Professor can add classes and save permission numbers for the classes at in bulk. The professor has a dashboard where he or she can quickly approve or deny requests for permission numbers for each of his classes.

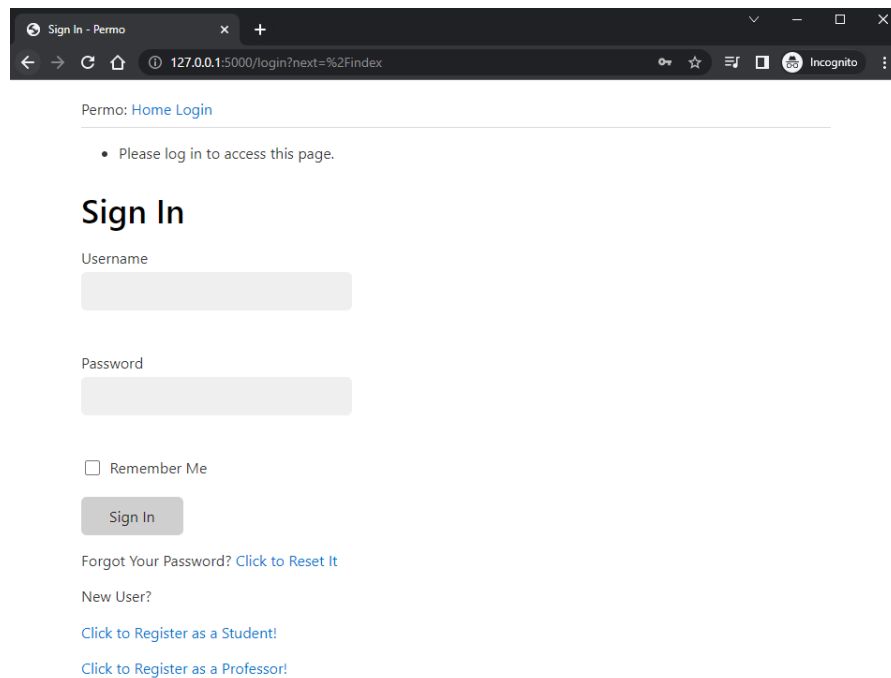
Each request can be individually tracked by the professor, and he can rescind the approval at any time.

The website logs the time of each permission number distributed to a student and keeps track of how many permission numbers the professor has left to distribute. Statistics regarding the distribution of permission numbers for each class are provided and the inventory of permission numbers is tracked for each class.

We employed the use of flask and SQLite to create a simple yet scalable solution. Flask is a great backend system for small to mid-range deployments and provides great scalability and development tools. Compared to other backends and databases, the Flask and SQLite approach is fast and lightweight.

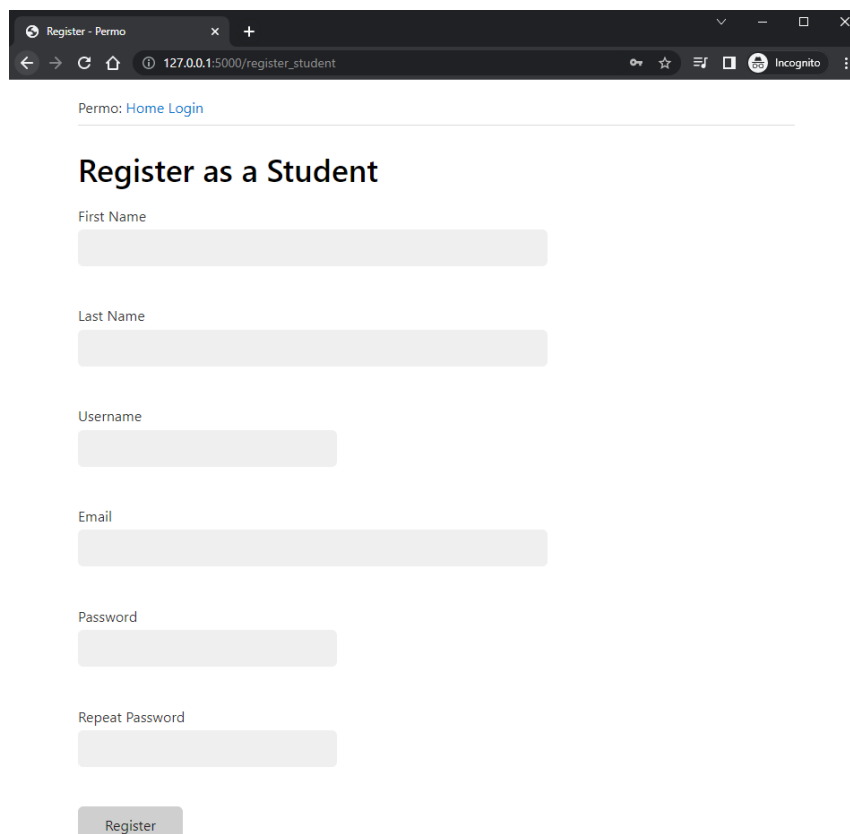
## ACCOMPLISHMENTS:

(i) Created a register page that collects information about the users, validates the information provided and registers the users.



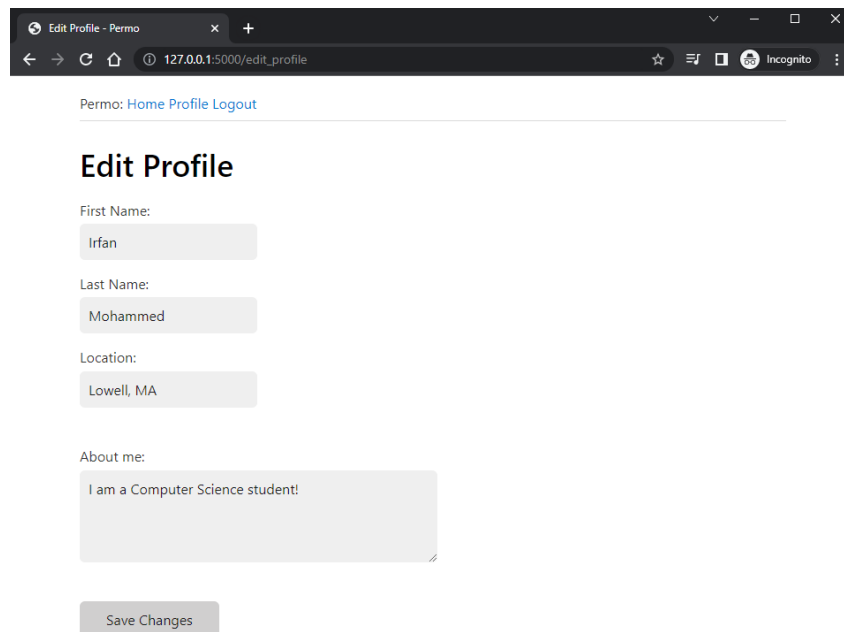
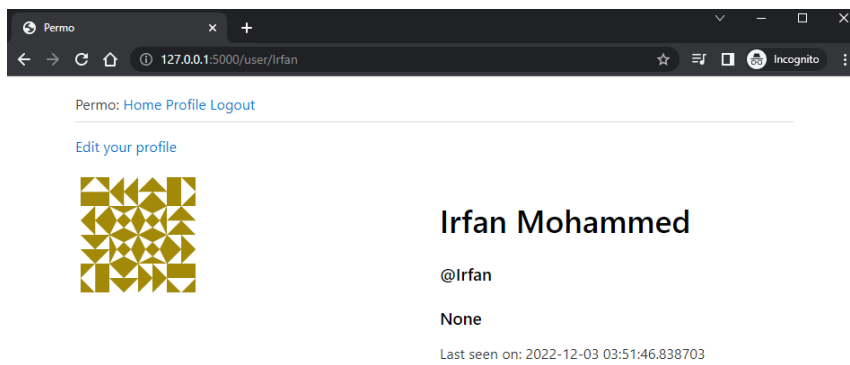
The screenshot shows a web browser window with the title "Sign In - Permo". The address bar displays "127.0.0.1:5000/login?next=%2Findex". The page content includes a link "Permo: [Home Login](#)" and a message "Please log in to access this page." Below this is a "Sign In" heading, followed by input fields for "Username" and "Password". There is a checkbox for "Remember Me" and a "Sign In" button. At the bottom, there are links for "Forgot Your Password? [Click to Reset It](#)", "New User?", "[Click to Register as a Student!](#)", and "[Click to Register as a Professor!](#)".

(ii) For a New User, we have two separate login pages for Student and Professor where it asks for First Name, Last Name, Username, Email ID (Accepts only @student.uml.edu for Students OR @uml.edu for Professors), and accepts a password.

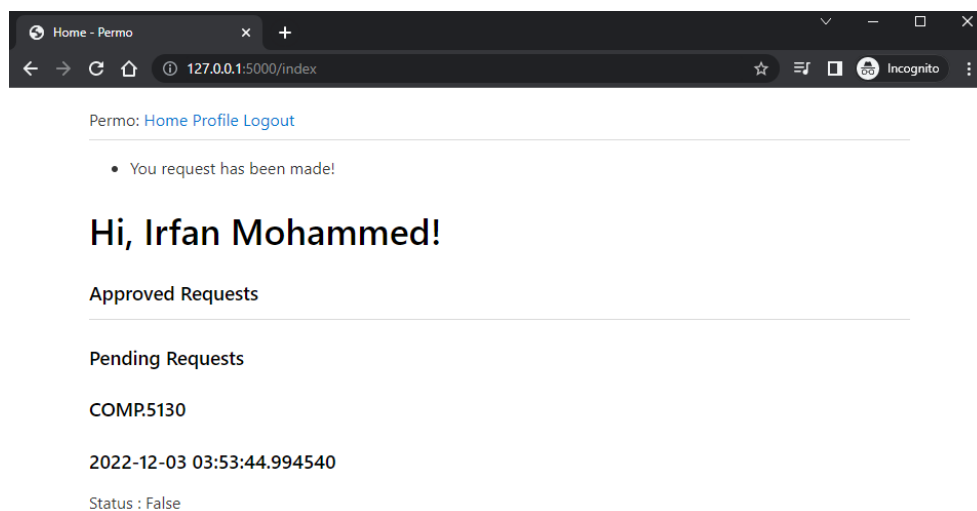


The screenshot shows a web browser window with the title "Register - Permo". The address bar displays "127.0.0.1:5000/register\_student". The page content includes a link "Permo: [Home Login](#)" and a heading "Register as a Student". Below the heading are input fields for "First Name", "Last Name", "Username", "Email", "Password", and "Repeat Password". At the bottom is a "Register" button.

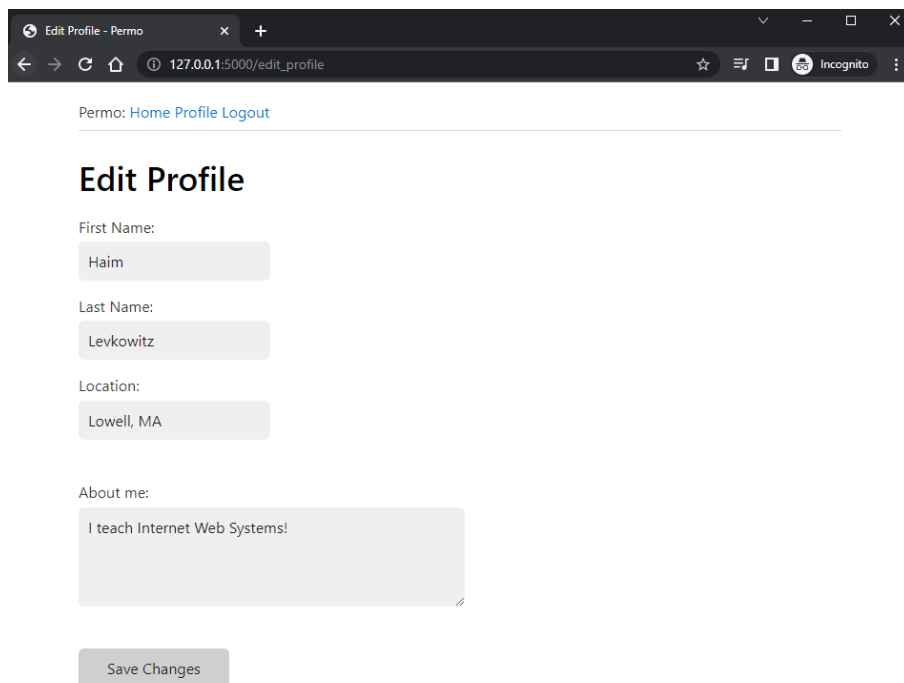
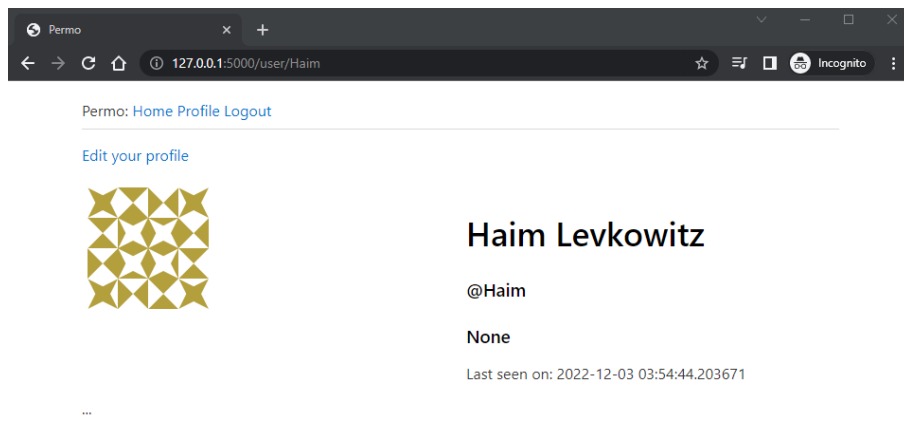
(iii) In Student's User Page, the student can customize the profile by adding their location and a short 'About Me'



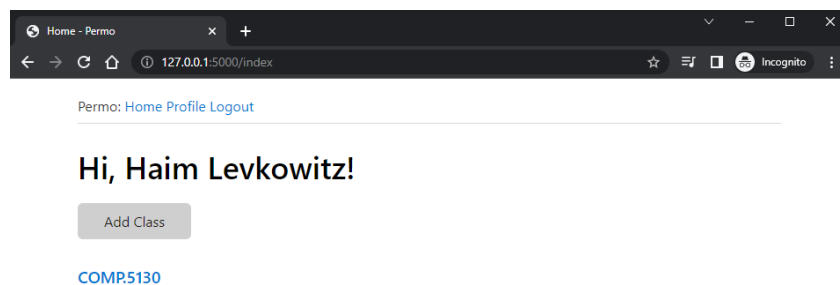
(iv) In Student's Home Page, he/she can see the status of their requests for permission numbers.



(v) In Professor's User Page, the professor can edit the profile by adding their location and a short 'About Me' and a student viewing a professor's user page can see the classes they offer and make a request for a permission number.



(vi) In Professor's Home page, he/she can add classes, and see the incoming requests for each of his listed classes.



(vii) Professor can add a permission number and when the professor either approves/declines a request, a permission number will be given to the student and it is sent to the student. The timestamp of the request will also be displayed.

On this screen, the professor can add any quantity of permission numbers as long as they are separated by commas. These permission numbers will be added and distributed to the students as requests are approved. If a request is declined, the request's status is immediately updated on the student's home page.

Permo: [Home Profile Logout](#)

- Request has been approved!

Permission numbers available: 4 / 4

Save Numbers

### Pending

Ranga Rohit Nallamolu Approve

@Rohit Decline

2022-12-03 03:58:14.369588

Status : False

### Approved

Irfan Mohammed Decline

@Rohit


2022-12-03 03:53:44.994540

Status : True

number: 12345

(viii) A student can check the Professor's profile with the URL: <http://127.0.0.1:5000/user/<Prrofessor Username>> and Request a Permission Number.

Permo: [Home Profile Logout](#)



## Haim Levkowitz

@Haim

Lowell, MA

I teach Internet Web Systems!

Last seen on: 2022-12-03 04:00:50.140408

...

### Courses offered

COMP.5130
<button>Request a Permission Number</button>

### **FEATURES IN PROGRESS:**

- (i) Password reset by sending a link to the user's email needs is currently being worked upon.
- (ii) When a student/professor is registering, the website does not verify if the email exists or not. This needs to be fixed by adding the ability to validate the emails.
- (iii) The ability for students to upload documents like Resumes & Transcripts will help the professor check if the student is eligible to enroll in the course is still under progress.
- (iv) The ability for the website to see if a student has accessed the permission number given to him. i.e., a notification should pop up if the student has used the permission number or not. Also, an expiry date can be set up for the permission number after which the permission number is rescinded if unused and added back to the inventory for that class.

### **UNABLE TO IMPLEMENT:**

- (i) Although the components are ready, we are having problems with requests and responses from different origins and wanted to implement the front end with REACT, but due to the tight deadline, we were unable to work on the idea of offering a high-quality front end.
- (ii) Apart from that, we decided to settle down and deliver the project while utilizing the Bootstrap front end, as that would allow us to implement everything else in the allotted time.

### **FEATURES TO ADD IN THE FUTURE:**

- (i) Generation of Permission Numbers so that the Professor doesn't have to add them.
- (ii) Integration of the flask project with the REACT as a front end with a proper directory structure and with proper planning of the project with real-time database integration.