****



**ICTAK LEARNER TRACKER**

**Submitted By**

AMANA HUSSAIN

ANUSHA ANIL

NEETHU S NAIR

PREMY PRASAD

VAISHAK C KRISHNAN

CERTIFIED SPECIALIST IN FULL STACK DEVELOPMENT (MEAN)

Kerala Knowledge Economy Mission (KKEM)

**ICT ACADEMY OF KERALA**

**JULY 2022**

**ICTAK LEARNER TRACKER**

**ABSTRACT**

The project is entitled as ICTAK Learner Tracker. This project developed under the platform of Angular as front end, NODEJS as back end and MongoDB as database. The main aim of this project is leaner tracker, and it is done by Trainer head and Placement officer. And the process is controlled by Admin. The admin should have the overall control over the website with CRUD operation on Training head and Placement officer. Admin can also view the details added by Trainer head and Placement officer.

**INTRODUCTION**

ICTAK Learner Trackers a simple yet powerful option that allows you to quickly plan, structure, and organise training, especially to create a long-term training program. In essence, the learning track is a series of courses that creates a linear flow of what your users have to learn and should enable the Training Head to view the status of each student enrolled with ICTAK.

**PROJECT OBJECTIVE**

The main objective of this system is to design or develop a Learner tracking portal for ICTAK for maintaining course details, Learner information, Training head and Placement officer. Users can use this system from anywhere to find out about the details of the learner. This platform must enrol the trainers and enable them to create, evaluate and analysis the feedback of students.

Project consists of an Admin login along with Trainer and Placement officer login. Admin have the overall control over the website with CRUD operations on Training Head & Placement Officer. Training Head and Placement Officer should be provided with unique login credentials which are added to the database statically by the admin/developer. Training Head should add the learners into this Dashboard by any of the below methods. Placement Officer should be able to to update the status of each student’s placement status.

**TEAM MEMBERS**

AMANA HUSSAIN

ANUSHA ANIL

NEETHU S NAIR

PREMY PRASAD

VAISHAK C KRISHNAN

**MODULE DESCRIPTION**

This project contains three modules

1.Admin Module

2.Trainer Module

3.Placement officer Module

**Admin Module**

* Login
* Admin panel
* Add new students
* Add new users (Trainer, Placement officer)
* Add batches, programme, courses
* Update batches, programme, course, trainer, placement officer
* View batches, programme, course, trainer, placement officer
* Also have privilege to add students using csv file uploading.

**Trainer Module**

* Login
* Trainer landing page
* Trainer has privilege to add, view, edit, delete students. Also have privilege to add students using csv file uploading.

**Placement Officer**

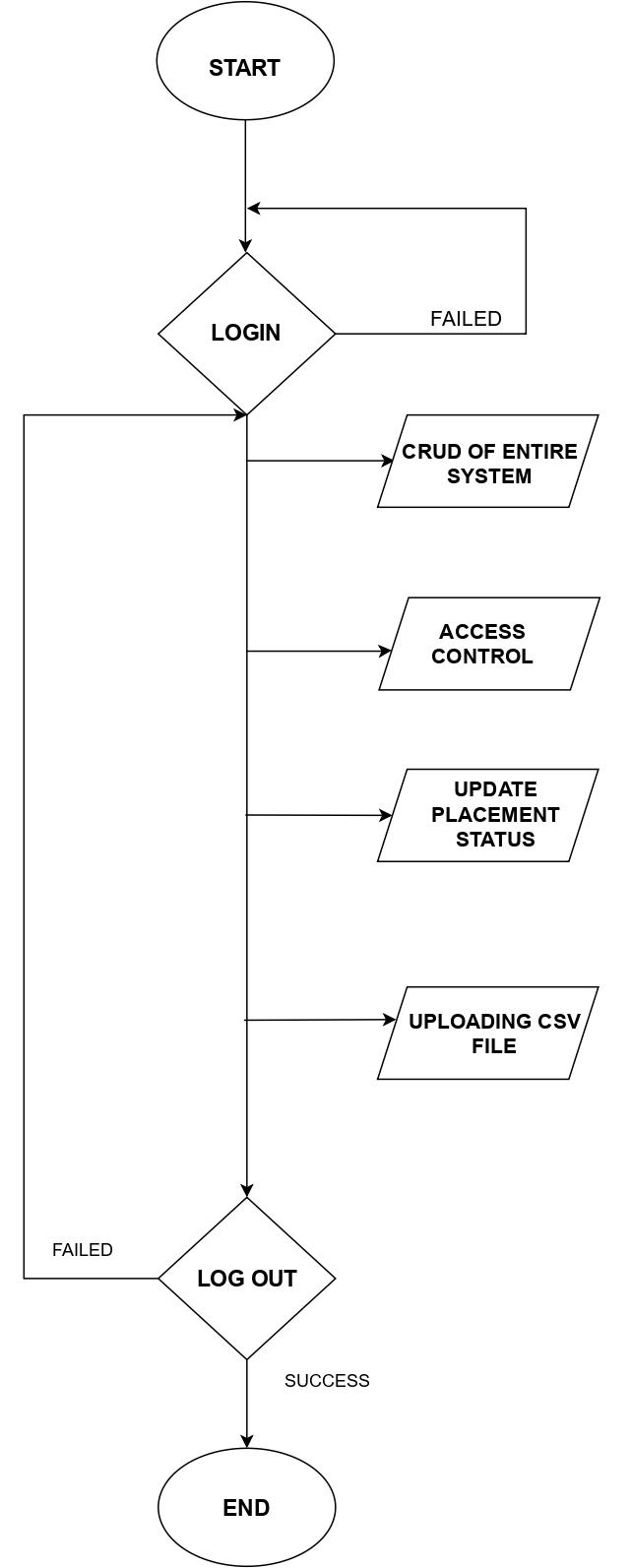
* Login
* Placement officer privilege over view students and has privilege to change placement status into placed, job seeking interested.

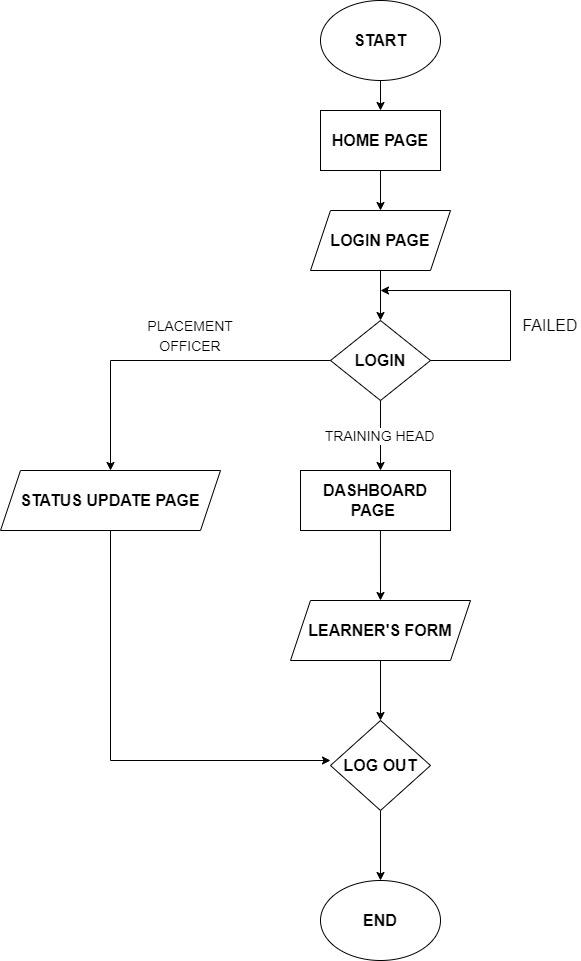
**Logout**

* Logout into login page.

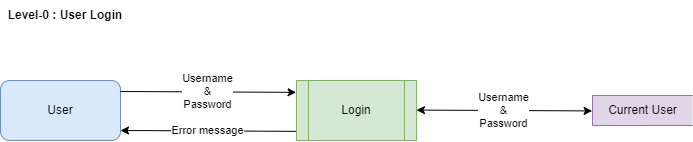
**FLOW CHARTS**

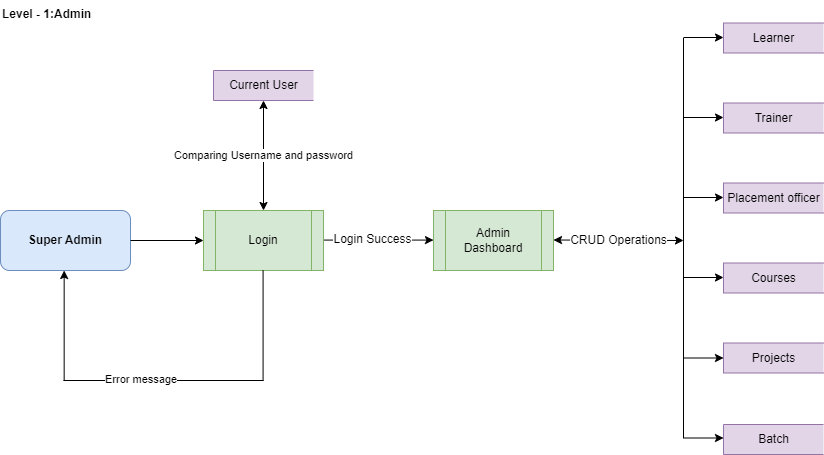
1. **Admin**

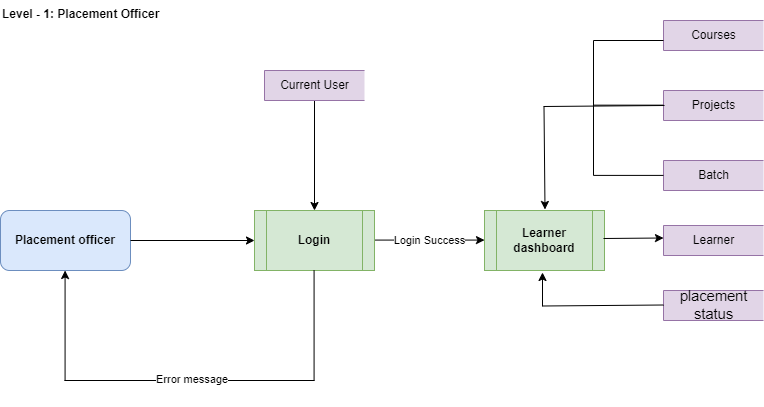
****

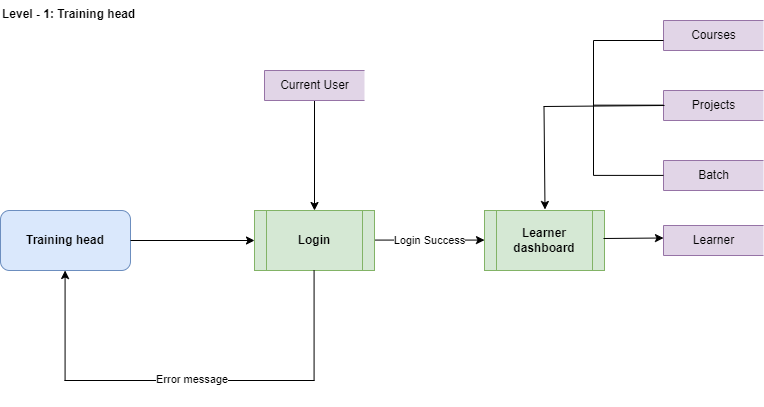
1. **Training Head and Placement officer**

**DATA FLOW DIAGRAMS**









**ICTAK Learner Tracker Schema Documentation**

**users**

|  |  |
| --- | --- |
| user\_name | username provided by admin   /admin/name of trainer/name of placement officer |
| password | password provided by admin for admin provided developer |
| user\_type | type of user (admin,trainer,placement officer) |
| user\_email | Email of user |
| office\_location | Location of ICTAK office |
| course | admin: Admin  trainer: Courses assigned by admin  placement officer: Placement |

**students – adding by form and uploading csv file**

|  |  |
| --- | --- |
| student\_name | name of the student |
| contact\_number | student contact number |
| email\_id | student email id |
| contact\_address | student contact address |
| course | course which is student is enrolled for |
| batch | enrolled batch |
| program | program name |
| training\_head | name of the training head |
| placement\_officer | name of placement officer |
| employment\_status | current employment status (placed/job seeking) |
| course\_status | course completed or not completed |

**for settings**

**courses**

|  |  |
| --- | --- |
| course\_name | name of the course |
| course\_description | description of course |

**programs**

|  |  |
| --- | --- |
| program\_name | name of the program |
| program\_description | description of program |

**batch**

|  |  |
| --- | --- |
| batch\_name | batch name |
| batch\_description | batch description |

# MODULE DESCRIPTION

**ADMIN**

* Landing page
* Add students
* Add students using csv
* Edit/delete students
* Add user
* Edit/delete user
* Add course
* Edit/delete course
* Add batch
* Edit/delete batch
* Add program
* Edit/delete program
* Login

**TRAINER**

* Add students
* Add students using csv
* Edit/delete students
* Add user
* Edit/delete user
* Login

**PLACEMENT OFFICER**

* Edit/delete placement status
* Login

**SYSTEM IMPLEMENTATION**

|  |  |  |
| --- | --- | --- |
| Project title: | ICTAK LEARNER TRACKER | Status |
| Description: | ICTAK Learner Trackers a simple yet powerful option that allows you to quickly plan, structure, and organise training, especially to create a long-term training program. In essence, the learning track is a series of courses that creates a linear flow of what your users have to learn and should enable the Training Head to view the status of each student enrolled with ICTAK. | **Frontend and backend completed 100%.** |
| FRONT- END DEVELOPMENT | |  |
| Modules | * LOGIN * DASH-BOARD * STUDENTS LIST * ADD, EDIT, DELETE STUDENTS * UPLOAD CSV FOR BATCH ADD * USER LIST * ADD, EDIT, DELETE STAFF * COURSE, BATCH, PROGRAM * ADD, DELETE FOR COURSE, BATCH, PROGRAM * SEARCHING AND PAGINATION |  |
| Back-End Development | | |
| mODULES &  mODELS | * AUTHENTICATION * AUTHORIZATION * POST STUDENTS * GET STUDENTS * PUT STUDENTS * DELETE STUDENTS * ADD USER * GET USER * PUT USER * DELETE USER * ADD BATCH COURSE PROGRAM * DELETE BATCH COURSE PROGRAM |  |
| Database | | |
|  | * USER COLLECTION WITH ROLE * STUDENTS COLLECTION * BATCH COLLECTION * COURSE COLLECTION * PROGRAM COLLECTION |  |

**Schema: JSON input sample**

Student Schema JSON Sample

{

  "\_id": "638871e75fdcaba42be66370",

  "student\_name": "test",

  "contact\_number": "987654321",

  "email\_id": "test@testmail.com",

  "contact\_address": "qwertyqwertyqwertyqwertyqwertyqwertyqwertyqwertyqwertyqwertyqwerty",

  "course": "FSD",

  "batch": "July 22",

  "program": "KKEM",

  "training\_head": "test tearining head",

  "placement\_officer": "test placement officer",

  "employment\_status": "not employed",

  "course\_status": "enrolled"

}

Course Schema JSON Sample

  {

      "course\_name":"test course 7",

      "course\_description":"lorem ipsum course"

     }

program Schema JSON Sample

  {

      "program\_name":"test program 7",

      "program\_description":"lorem ipsum program"

     }

batch Schema JSON Sample

{

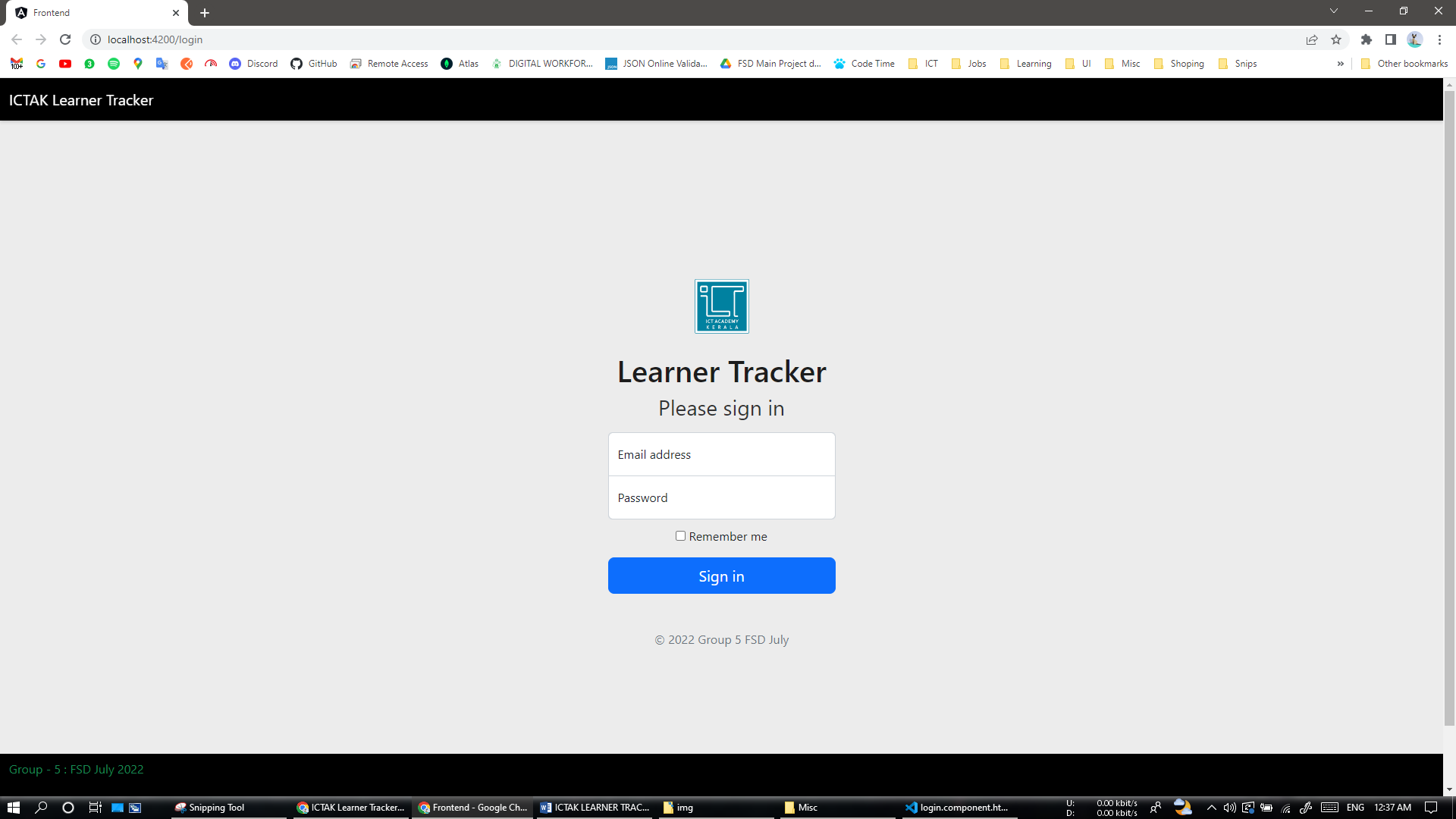
    "batch\_name":"test batch 7",

    "batch\_description":"lorem ipsum batch"

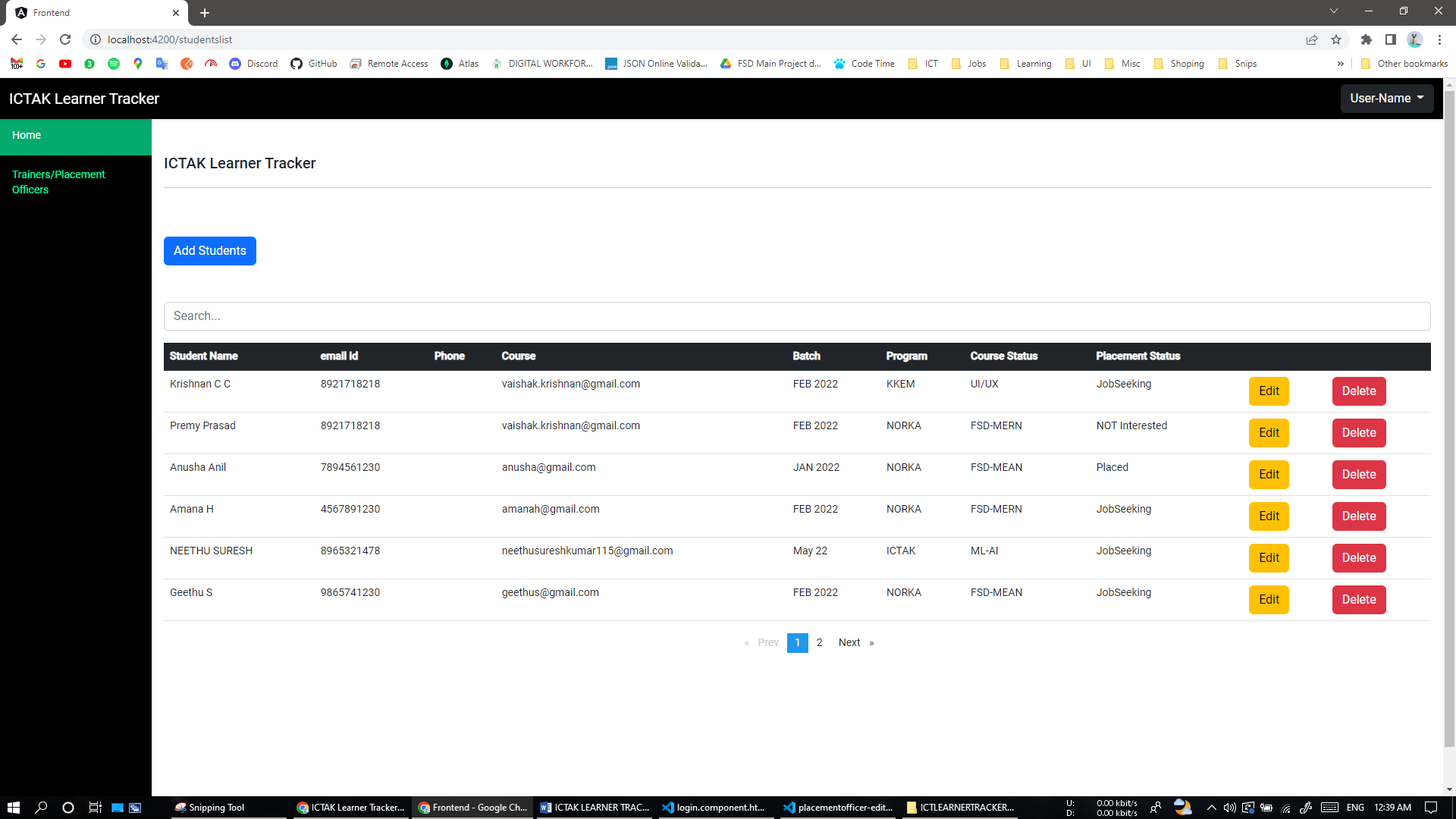
   }

**SCREENSHOTS**

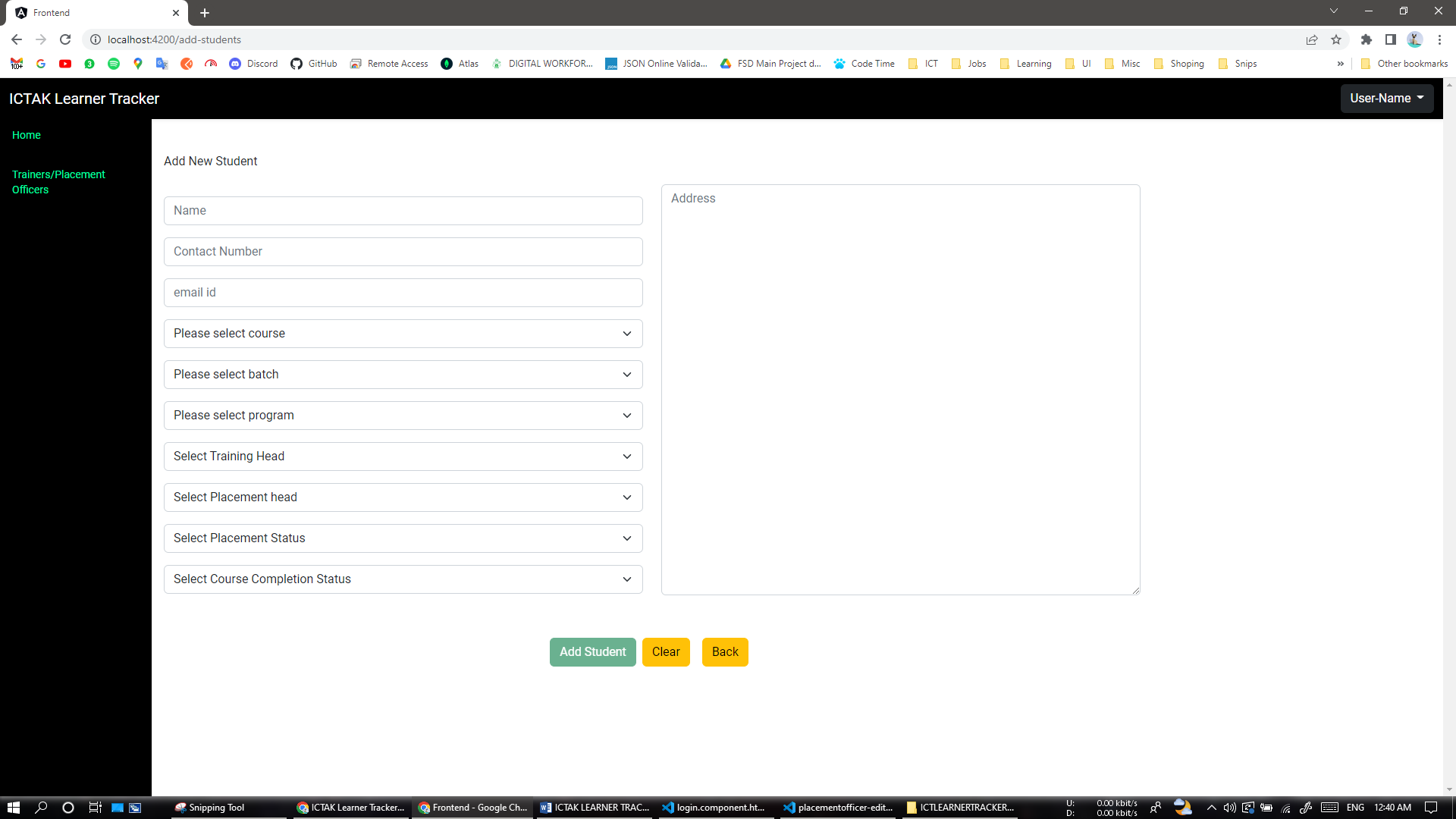
Login



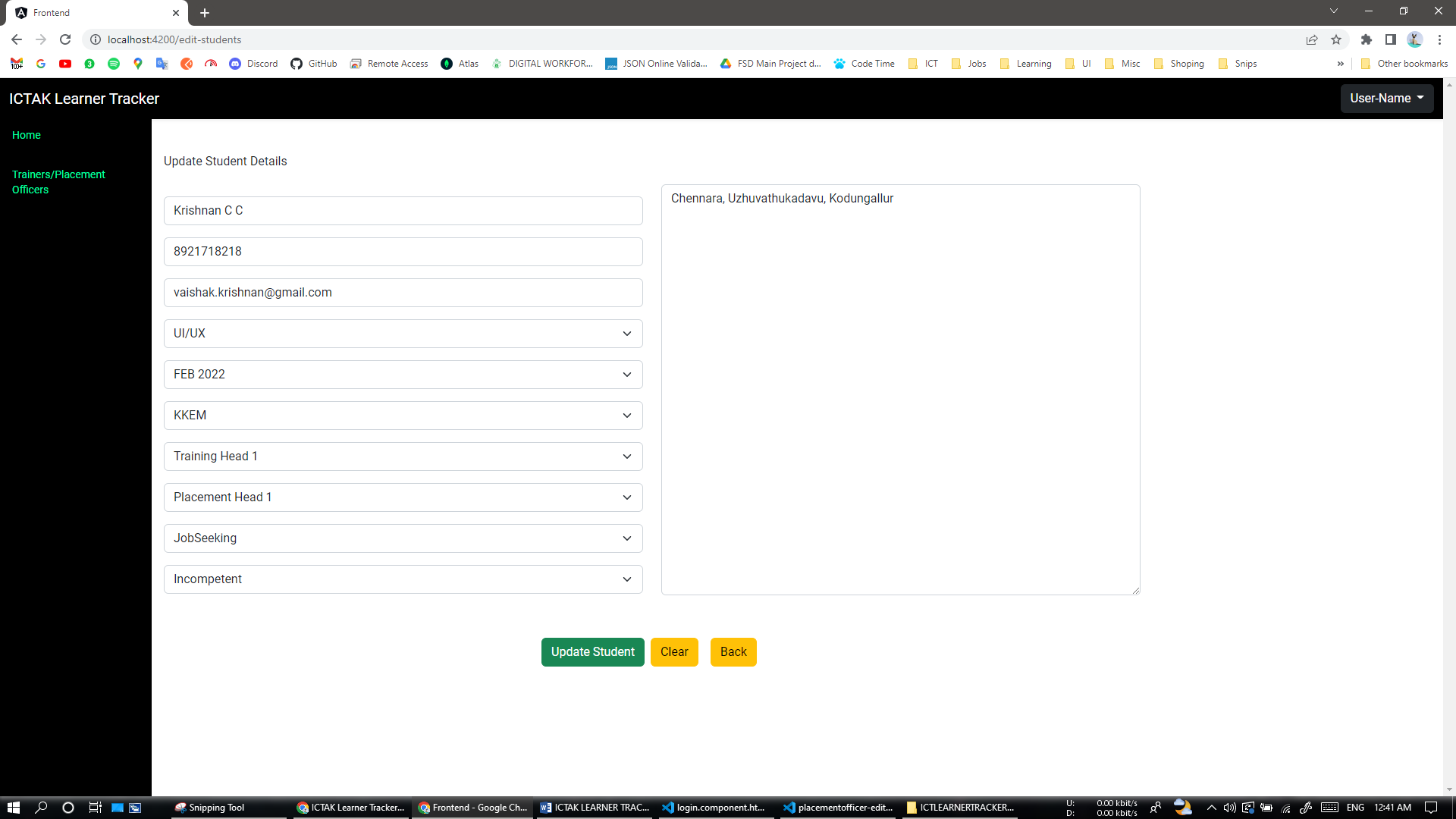
Admin: Landing page



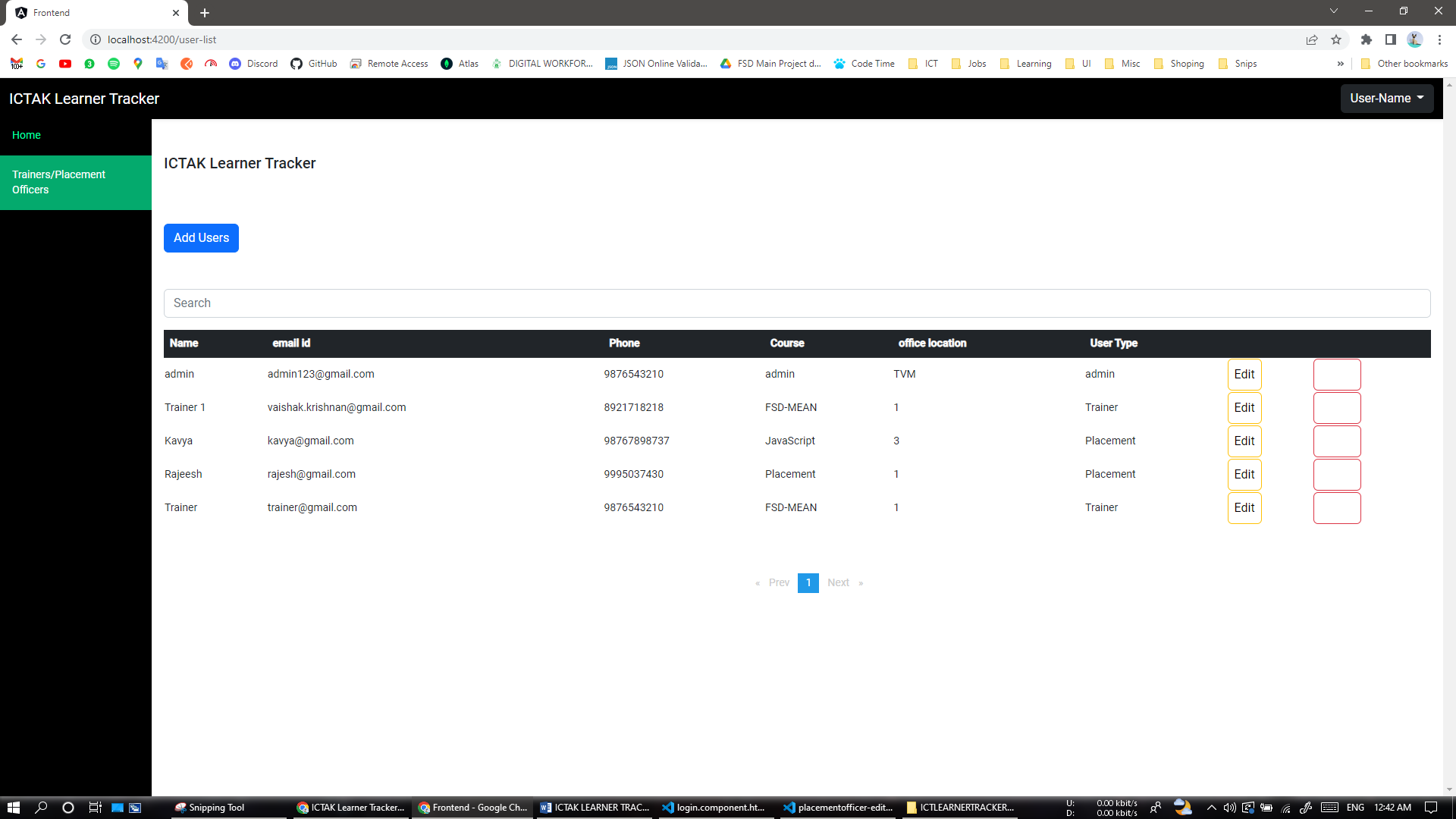
Admin: Add-Student



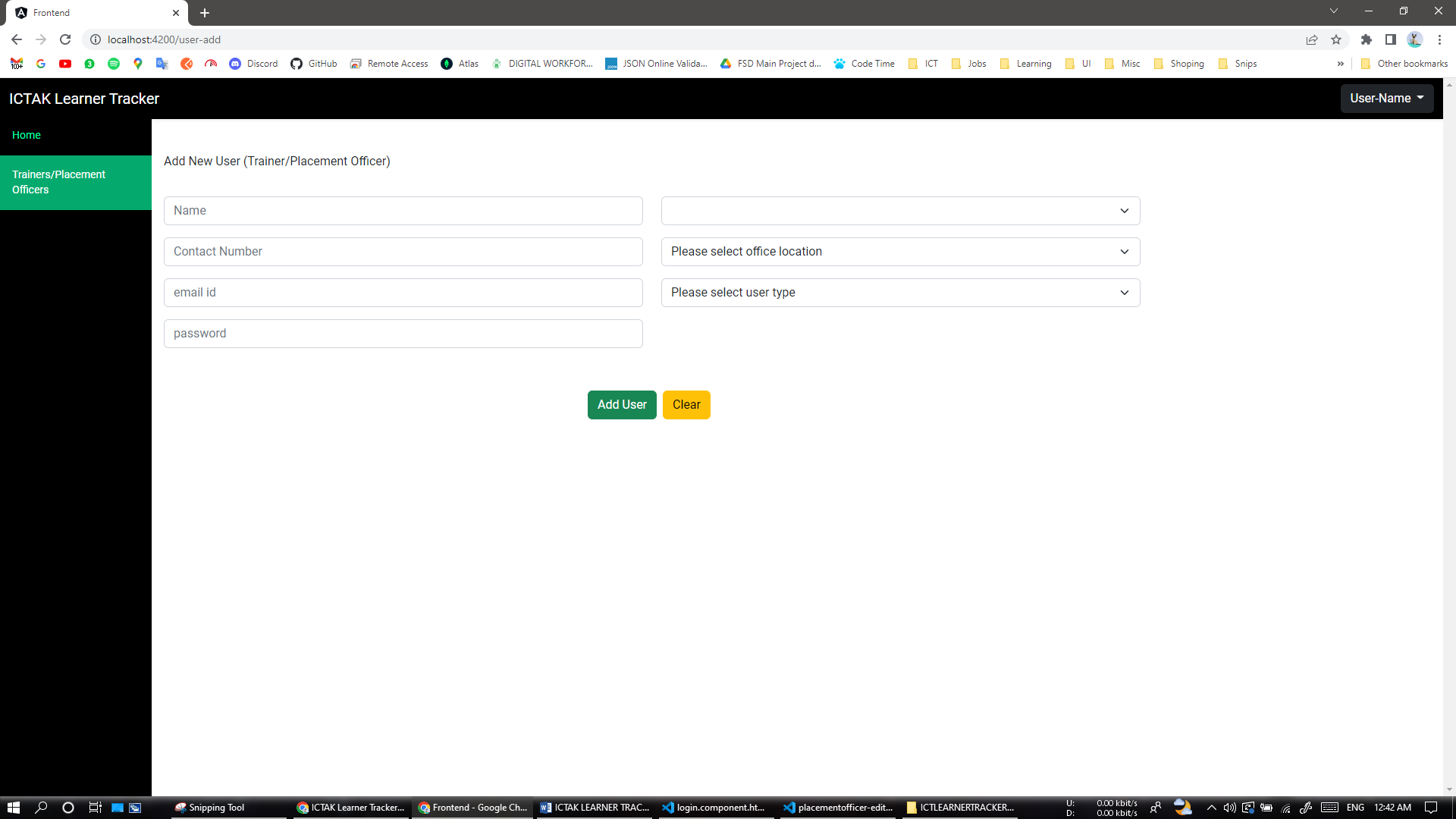
Admin: Edit-Student



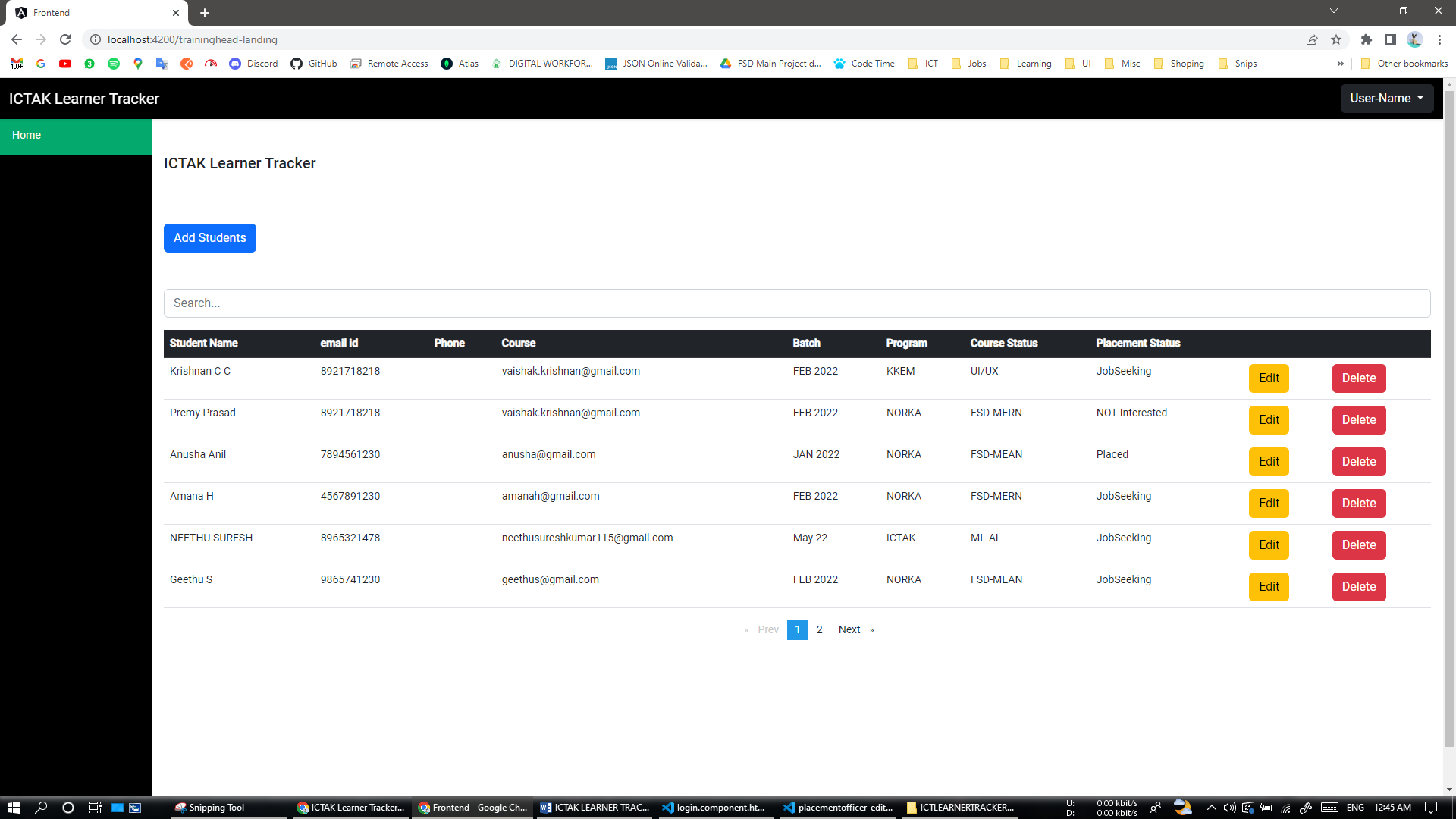
Admin: User-list (Training Head/Placement Officer)



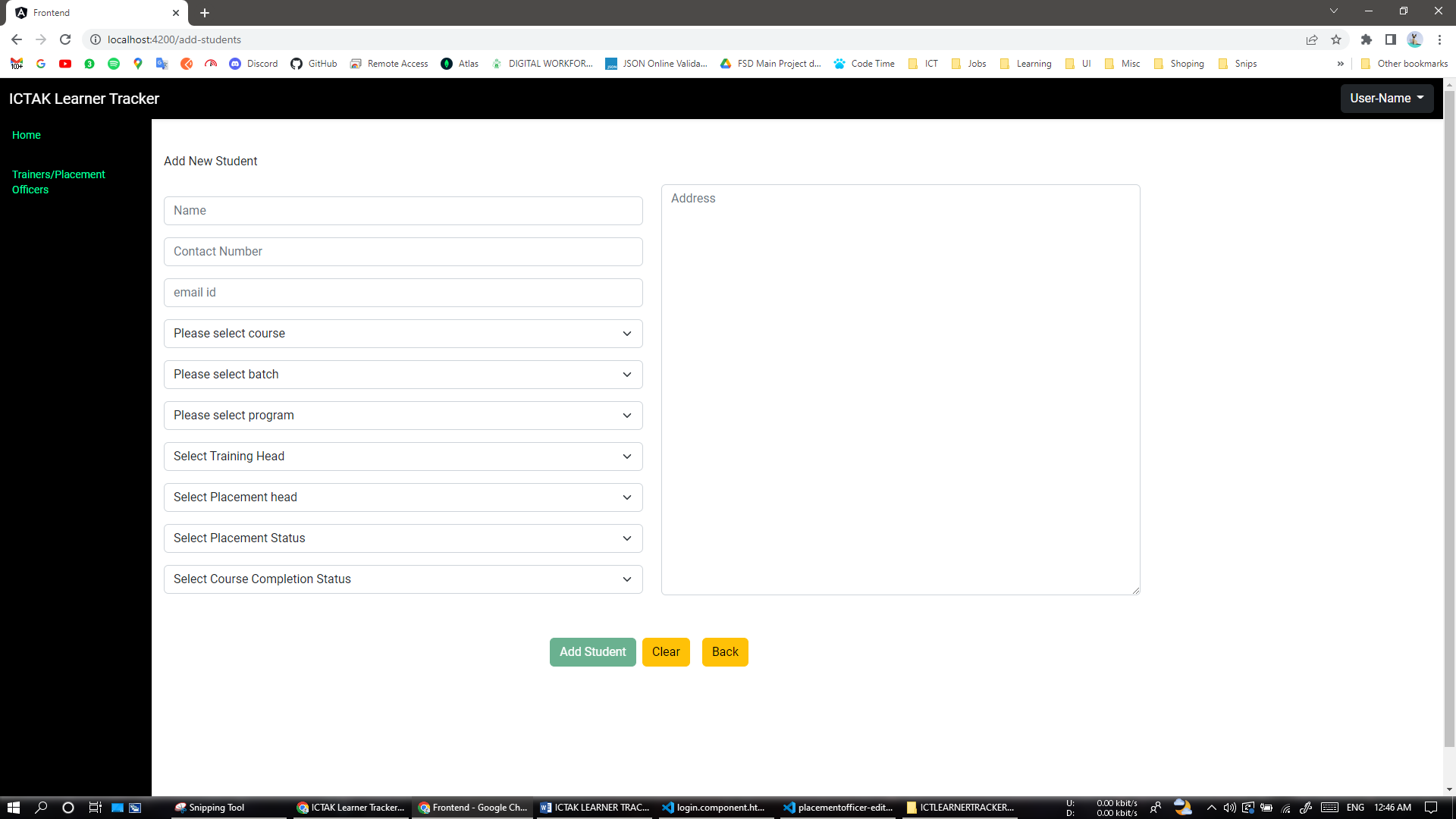
Admin: Add-User (Training Head/Placement Officer)



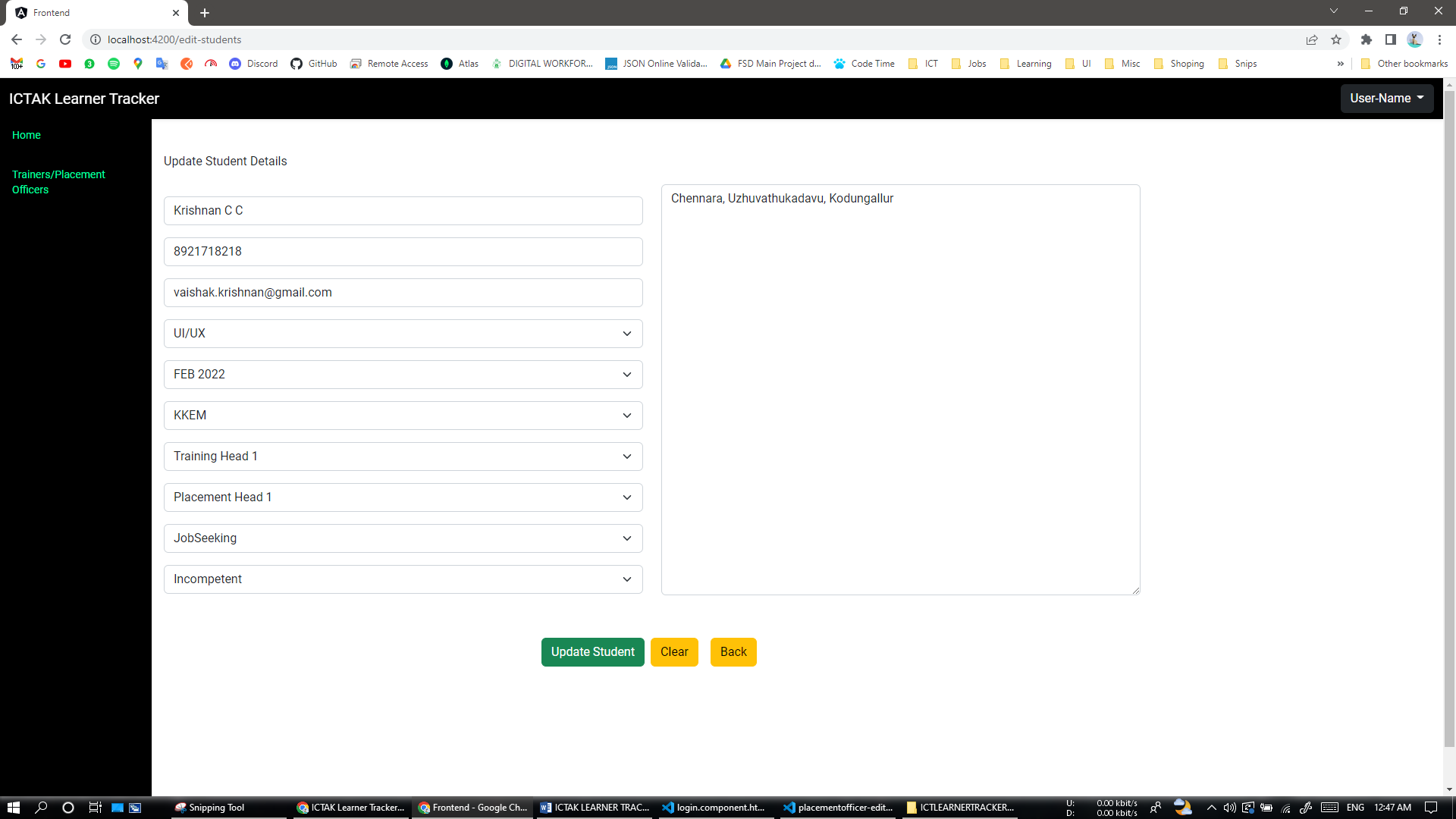
Training Head: Landing Page



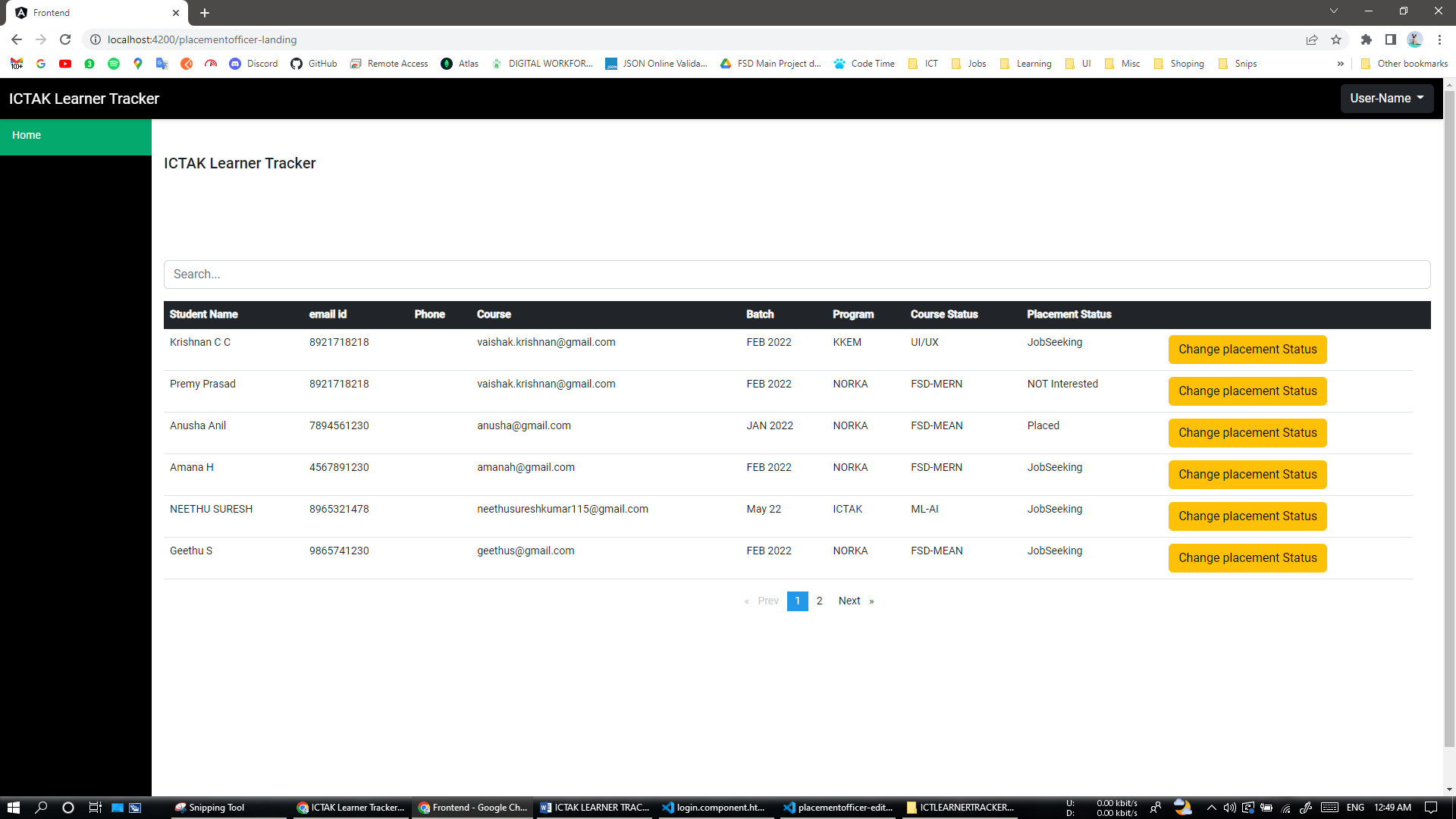
Training Head: Add-Student



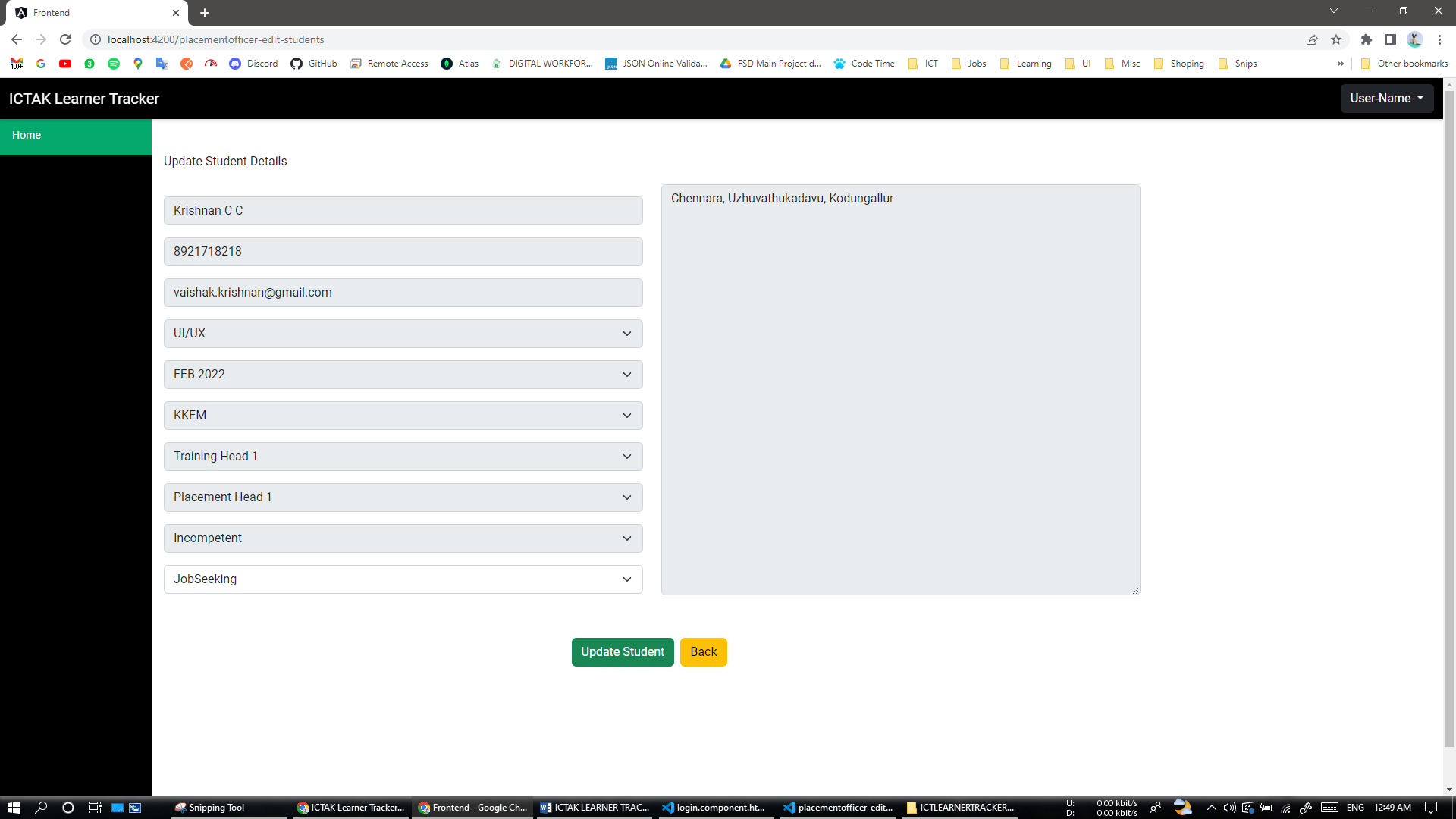
Training Head: Edit-Student



Placement Officer: Landing Page



Placement Officer: Change Placement status (student)



**LINKS**

**Hosted Link**

https://calm-pear-pike-gear.cyclic.app/login

**Git hub build deployment link**

https://github.com/vaishak-krishnan/ICTAK-Learner-tracker-Deployment

**Complete Project (backend and frontend)**

https://github.com/vaishak-krishnan/ICTK-Group-5

**CONCLUSION**

This ICTAK Learner Tracking application as per the Software Requirement Specification (SRS). The application is tested completely and it is properly debugged.

**END**