
User Manual

for

AvailSys

Prepared by:

Lekhana Mitta - IIT2019204

Dhanush Vasa - IIT2019208

Pravallika Kodi - IIT2019234

Kandagatla Meghana Santhoshi- IIB2019030

Indian Institute of Information Technology, Allahabad

Table of contents

Table of contents	2
1 Prerequisites	4
1.1 Install python	4
1.2 Downloaded with errors(optional)	4
1.3 Install pip manually(optional)	5
1.4 Install django	5
1.5 Setting up virtual environment for running django server	5
2 Running Django server	7
2.1 MakeMigrations and Migrate (Creates the Database or to update the database - Must do always)	7
2.2 Create admin login (if admin login was previously not created)	7
2.3 Execution of server	8
3 Instructions for navigating the application:	9
4 User features	11
4.1 Login - http://127.0.0.1:8000/login/	11
4.2 Signup - http://127.0.0.1:8000/register/	11
4.3 Dashboard - http://127.0.0.1:8000/DashBoard/	12
4.4 Manage Sections- http://127.0.0.1:8000/Manage_Sections/	12
4.5 Create Sections - http://127.0.0.1:8000/create_section/	13
4.6 Add Student - http://127.0.0.1:8000/Create_Student/	13
4.7 View Students - http://127.0.0.1:8000/students/	14
4.8 Search Students - http://127.0.0.1:8000/students/?section=&emails=&roll_number=	14
4.9 Delete Students- http://127.0.0.1:8000/delete_student/24/	15
4.10 Update Student- http://127.0.0.1:8000/update_student_id/24/	15
4.11 View Sections - http://127.0.0.1:8000/view_section/	16
4.12 Delete Section - http://127.0.0.1:8000/delete_section/8/	16
4.13 Manage Labs - http://127.0.0.1:8000/Manage_labs_page/	17
4.14 Schedule a slot - http://127.0.0.1:8000/room_list/	17
4.15 View labs - http://127.0.0.1:8000/booking_list/	18
4.16 Manage Resources - http://127.0.0.1:8000/Manage_resources_page/	18
4.17 Schedule Resources - http://127.0.0.1:8000/resource_book/	19
4.18 Cancel Slot - http://127.0.0.1:8000/delete_booked_lab/22/	19
4.19 Cancel Resources - http://127.0.0.1:8000/delete_resource/11/	20
4.20 Mail Notifications	20
5 Admin features	21
5.1 Login - http://127.0.0.1:8000/admin/login/?next=/admin/auth/user/	21
5.2 Bookings - http://127.0.0.1:8000/admin/app/booking/	21

5.3 Resource_bookings http://127.0.0.1:8000/admin/app/resource_booking/	22
5.4 Resources - http://127.0.0.1:8000/admin/app/resource/	22
Here we can find all the resources which are available to allot for the labs in their respective category	22
5.6 Sections - http://127.0.0.1:8000/admin/app/section/	23
5.7 Student_links - http://127.0.0.1:8000/admin/app/student_link/	24
5.8 Groups - http://127.0.0.1:8000/admin/auth/group/add/	24
5.9 Users - http://127.0.0.1:8000/admin/auth/user/	25
6 Student Features	25
6.1 Receive Email	25

1 Prerequisites

1.1 Install python

- For windows - install the latest version of python from
<https://www.python.org/downloads/windows/>
- For Linux/Unix - install the latest version of python from
<https://www.python.org/downloads/source/>
- For Mac OS X - install the latest version of python from
<https://www.python.org/downloads/mac-osx/>
- Other platforms - install the latest version of python from
<https://www.python.org/download/other/>
- Installing from any of these websites would automatically download pip packages, but in case they are not downloaded, it can be done manually.

1.2 Downloaded with errors(optional)

- If python / pip packages downloaded with errors follow this section, else skip to installing django.
- Make sure you have Python and that the expected version is available from your command line. You can check this by running:

Windows	Linux/Unix/MAC OS X
<code>py --version</code>	<code>python3 --version</code>

You should get some output like Python 3.6.3. If you do not have Python, please install the latest 3.x version from <https://www.python.org/downloads/>

- Additionally, you'll need to make sure you have pip available. You can check this by running:

Windows	Linux/Unix/MAC OS X
<code>py -m pip --version</code>	<code>python3 -m pip --version</code>

- If you installed Python from source, with an installer from <https://www.python.org/downloads/> you should already have pip. If you're on Linux and installed using your OS package manager, you may have to install pip separately, see <https://packaging.python.org/guides/installing-using-linux-tools/>

If for some reason, pip hasn't installed properly, refer installing pip manually or else skip the section

- While pip alone is sufficient to install from pre-built binary archives, up to date copies of the setuptools and wheel projects are useful to ensure you can also install from source archives:

```
python -m pip install --upgrade pip setuptools wheel
```

1.3 Install pip manually(**optional**)

- Using python download will automatically download pip. If that is the case please skip this section and start installing django.
- To manually install pip, securely download get-pip.py by following this link: <https://bootstrap.pypa.io/get-pip.py>
Or use the command : `curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py`
- get-pip.py options
 - no-setuptools If set, do not attempt to install setuptools
 - no-wheel If set, do not attempt to install wheel
- For detailed instructions visit : <https://pip.pypa.io/en/stable/installing/>

1.4 Install django

- To install django enter the following command in the command line

```
python -m pip install Django
conda install -c anaconda django
pip install django-allauth
pip install django-filter
pip install django-environ
pip install pillow
pip install bcrypt
pip install django[argon2]
pip install django-email
```

After all these steps are completed, python, pip and django must have been installed.

1.5 Setting up virtual environment for running django server

- Type the following command in command line

Windows	Linux/Unix/MAC OS X
<code>py -m venv djangovenv</code>	<code>python3 -m venv djangovenv</code>

Actual syntax - `py/python3 -m venv <desired venv name>`

- Following commands should produce a similar result on your PC

```
cd djangovenv
```

```
dir
```

```
05/18/2021 11:31 PM <DIR> .
05/18/2021 11:31 PM <DIR> ..
05/18/2021 11:31 PM <DIR> Include
05/18/2021 11:31 PM <DIR> Lib
05/18/2021 11:31 PM 133 pyvenv.cfg
05/18/2021 11:31 PM <DIR> Scripts
1 File(s) 133 bytes
5 Dir(s) 250,147,717,120 bytes free
```

```
cd Scripts
```

```
05/18/2021 11:31 PM <DIR> .
05/18/2021 11:31 PM <DIR> ..
05/18/2021 11:31 PM 1,992 activate
05/18/2021 11:31 PM 984 activate.bat
05/18/2021 11:31 PM 19,328 Activate.ps1
05/18/2021 11:31 PM 368 deactivate.bat
05/18/2021 11:31 PM 106,371 pip.exe
05/18/2021 11:31 PM 106,371 pip3.9.exe
05/18/2021 11:31 PM 106,371 pip3.exe
05/18/2021 11:31 PM 539,312 python.exe
05/18/2021 11:31 PM 537,776 pythonw.exe
9 File(s) 1,418,873 bytes
2 Dir(s) 250,142,232,576 bytes free
```

- Now run activate.bat file by using the following command:

```
activate.bat
```

```
C:\Users\Admin.[REDACTED]\djangovenv\Scripts>activate.bat

(djangovenv) C:\Users\Admin.[REDACTED]\djangovenv\Scripts>.
```

If you get a result similar to this then the virtual environment is activated and all the commands from now on can be run on the virtual environment.

If you want to exit the virtual environment, then press **ctrl+c**

This completes the installation process 

2 Running Django server

In the command line change the current directory to the virtual environment we have created and activate the virtual environment. (Refer section 1.5 in the document)

2.1 MakeMigrations and Migrate (Creates the Database or to update the database - Must do always)

```
python manage.py makemigrations
```

```
(MyDjangoEnv) PS C:\Users\vasad\documents\lab\lab_system_05-04> python manage.py makemigrations
No changes detected
(MyDjangoEnv) PS C:\Users\vasad\documents\lab\lab_system_05-04>
```

```
python manage.py migrate
```

```
(MyDjangoEnv) PS C:\Users\vasad\documents\lab\lab_system_05-04> python manage.py migrate
Operations to perform:
  Apply all migrations: account, admin, app, auth, contenttypes, sessions, sites, socialaccount
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying account.0001_initial... OK
  Applying account.0002_email_max_length... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying app.0001_initial... OK
  Applying app.0002_booking_section... OK
  Applying app.0003_remove_room_capacity... OK
  Applying app.0004_resource_resource_booking... OK
  Applying app.0005_auto_20210324_1929... OK
  Applying app.0006_auto_20210407_2204... OK
  Applying app.0007_auto_20210407_2235... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying sessions.0001_initial... OK
  Applying sites.0001_initial... OK
  Applying sites.0002_alter_domain_unique... OK
  Applying socialaccount.0001_initial... OK
    Applying socialaccount.0002_token_max_lengths... OK
    Applying socialaccount.0003_extra_data_default_dict... OK
(MyDjangoEnv) PS C:\Users\vasad\documents\lab\lab_system_05-04>
```

2.2 Create admin login (if admin login was previously not created)

Here Before we run the server we will have to create a superuser:

This is done by the command:

```
python manage.py createsuperuser
```

```
(MyDjangoEnv) PS C:\Users\vasad\onedrive\desktop\lab\lab_system_05-04> python manage.py createsuperuser
Username (leave blank to use 'vasad'): Dhanush
Email address: vasadhanush99@gmail.com
Password:
Password (again):
Superuser created successfully.
```

2.3 Execution of server

After activation, move to the project folder and type the following command to start the server : (The command is common to all OSs.)

python manage.py runserver

This starts the server and your command line will look something like this:

```
(MyDjangoEnv) PS C:\Users\vasad\onedrive\desktop\lab\lab_system_05-04> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
May 19, 2021 - 21:24:53
Django version 3.0.5, using settings 'Lab_System.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Copy the url and paste it into your web browser.

 Now you have the website opened in your system

This is just to ensure that the server keeps track of bookings made on the website.

3 Instructions for navigating the application:

1. Run the django server (refer section 2 in the document)
2. Login with the same credentials used to create superuser (from section 2)
3. After logging in successfully you will be faced with the following list of options:
 1. Bookings
 2. Resource_bookings
 3. Resources
 4. Rooms
 5. Sections
 6. Student_links
 7. Groups
 8. Users
4. Now you can see each of the options but you will notice that they would all be empty.
5. First you will have to go into Rooms and create all the different rooms that are available in your campus or school and select each of their categories with room number.
6. Secondly you have to go into Resources where you now have to add each and every resource into the system based on its category.
7. Now the system is ready for faculty to register and login to book rooms. Now Logout.
8. Register as a teacher and make sure to give a valid email address.
9. Sign in with the registered login and password
10. After logging in you will come to the menu page where you will have 3 options.
 1. About
 2. Dashboard
 3. logout
11. The About page has details of the project and the logout logs you out. Firstly click on Dashboard
12. When you come to the Dashboard page again you will have 3 options.
 1. Manage Sections
 2. Manage Labs
 3. Manage resources
13. First you have to enter manage sections when you have to create a new section first and then add as many student as you want to the same section (optional you can make multiple sections and add multiple sections)
14. Return to Dashboard
15. Now to book a lab go into the manage labs where you can book a lab by following the instructions and requests on the screen.

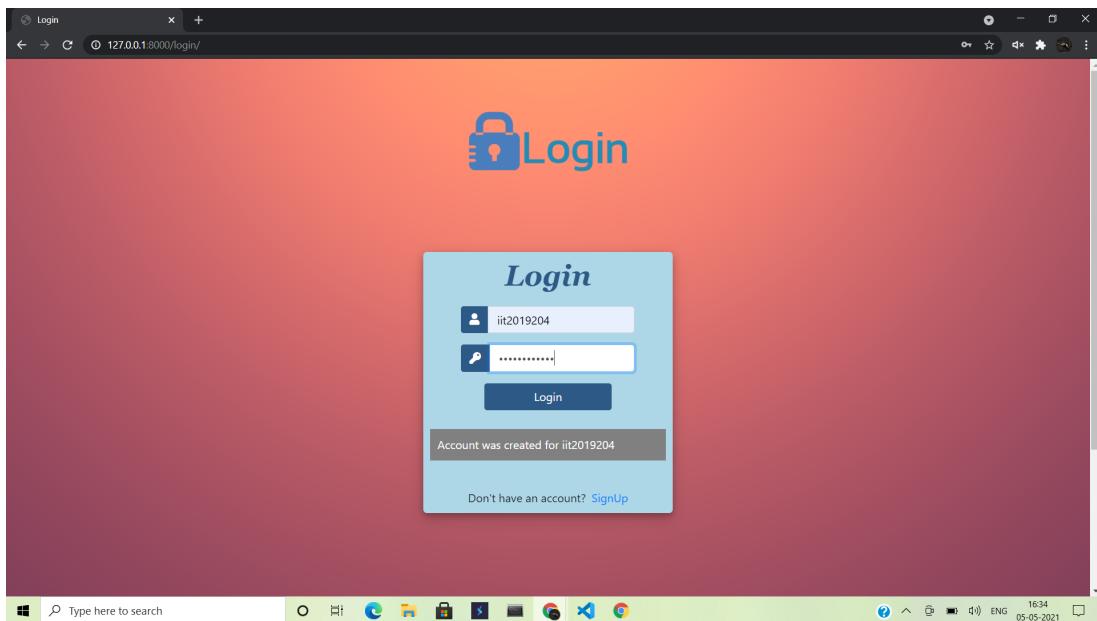
16. After a successful booking return back to the DashBoard.
17. Now go into manage resources where you can book the resource by pressing the book resource button.
18. Here after selecting the resource you have to press save not return or else resource won't be requested
19. After all the resources have been requested then you can press return.
20. Logout
21. Now login in again into the admin page again.
22. Now you would be able to see all the options have some sort of information regarding the system.
23. In the admin page you have to go into resource_bookings where you have the control to accept or decline the request or to leave it pending.
24. Return back to home page for admin where you can also view all the users in Users options
25. Logout
26. For all booking of labs or creating sections or adding students or booking resources an email is always sent to notify the required recipients.
27. This is the complete of avalisys.

For detailed information about every feature refer section 4 and 5

4 User features

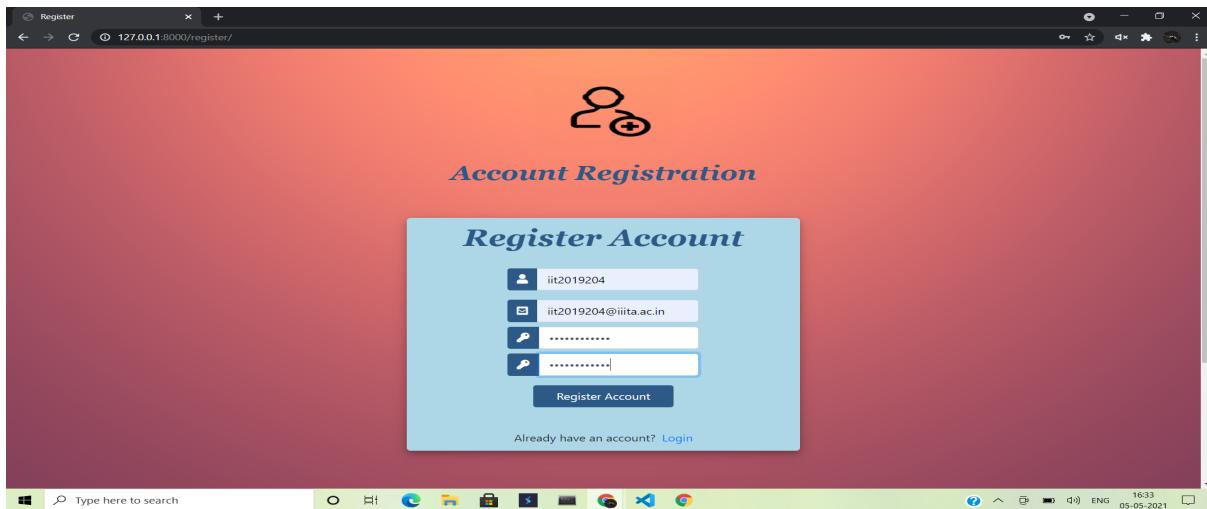
4.1 Login - <http://127.0.0.1:8000/login/>

Enter the Email ID and Password and click on the “SignIn” button.



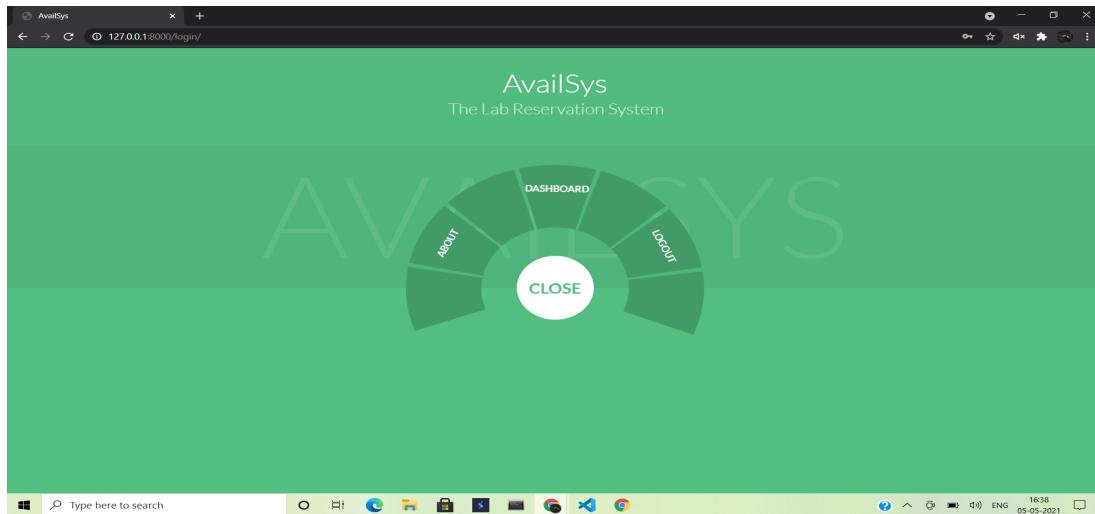
4.2 Signup - <http://127.0.0.1:8000/register/>

This is the initial registration page so that the Faculty can create a personal account.



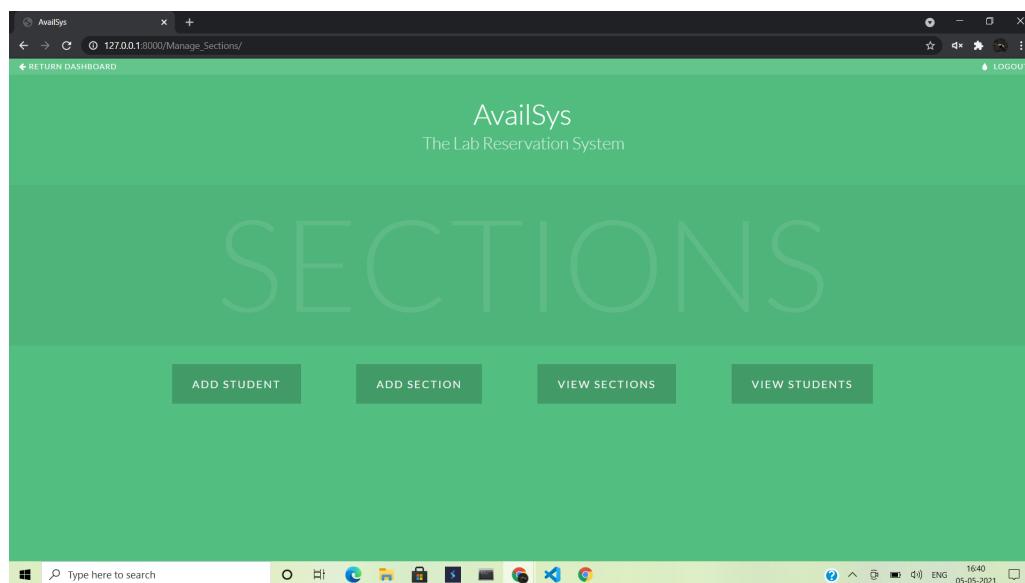
4.3 Dashboard - <http://127.0.0.1:8000/DashBoard/>

This is the page where Faculty is redirected after he\she logs into the system. Faculty can navigate from here to all his sections of the features.



4.4 Manage Sections- http://127.0.0.1:8000/Manage_Sections/

From the previous dashboard page if Faculty Selects manage sections page , then we get navigated to this page.



4.5 Create Sections - http://127.0.0.1:8000/create_section/

From the previous page , if we select the create section button . We get navigated to this page inorder to create a section.

The screenshot shows a web browser window titled "AvailSys". The URL in the address bar is "127.0.0.1:8000/create_section/". The page content is titled "AvailSys The Lab Reservation System" and features a large, semi-transparent "ADD SECTION" text overlay. Below this, there is a form with a single input field labeled "Section:" containing the value "sem-IV-C". At the bottom of the form is a "Submit" button.

4.6 Add Student - http://127.0.0.1:8000/Create_Student/

Faculty can add students into the section selected , he can add students later after the creation of the section.He includes roll number and email ID to add a student.

The screenshot shows a web browser window titled "AvailSys". The URL in the address bar is "127.0.0.1:8000/Create_Student/". The page content is titled "The Lab Reservation System" and features a large, semi-transparent "ADD STUDENT" text overlay. Below this, there are three form fields: "Section:" with the value "sem-IV-C", "Roll Number:" with the value "IIT2019208", and "Email:" with the value "iit2019208@iita.ac.in". At the bottom of the form is a "Submit" button.

4.7 View Students - <http://127.0.0.1:8000/students/>

In this page , Faculty can see all his students of all sections , with the feature included to search, delete,update.

The screenshot shows a web browser window titled 'AvalSys'. The URL in the address bar is '127.0.0.1:8000/students/'. The main content area has a green header with the text 'VIEW STUDENT'. Below the header is a search form with fields for 'Section:', 'Email:', and 'Roll Number:', followed by a 'SEARCH' button. A table below the search form displays student information:

Section ID	Student ID	Email ID	Update	Remove
sem-IV-C iit2019204	IIT2019208	iit2019208@iita.ac.in	UPDATE	DELETE

4.8 Search Students -

http://127.0.0.1:8000/students/?section=&emails=&roll_number=

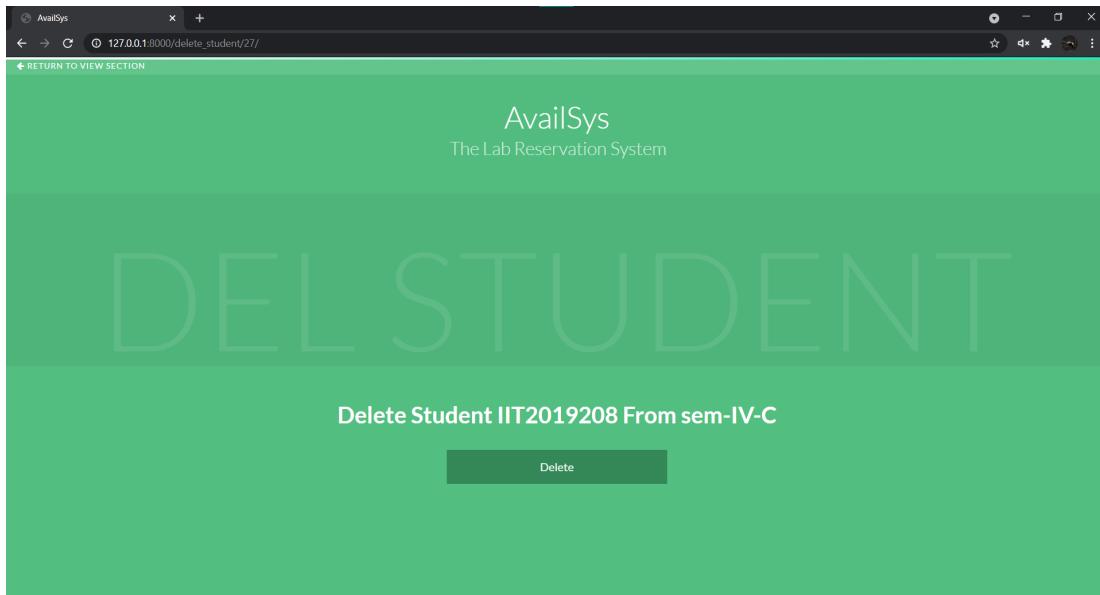
Faculty can search students whenever necessary to get all the details of a particular student or to delete or edit directly instead of scrolling and making things done.From the Buttons , we can navigate and later features are explained after this.

The screenshot shows a web browser window titled 'AvalSys'. The URL in the address bar is '127.0.0.1:8000/students/?section=&emails=iib2019030&roll_number='. The main content area has a green header with the text 'VIEW STUDENT'. Below the header is a search form with fields for 'Section:', 'Email:', and 'Roll Number:', followed by a 'SEARCH' button. A table below the search form displays student information:

Section ID	Student ID	Email ID	Update	Remove
sem-IV-C iit2019204	IIB2019030	iib2019030@iita.ac.in	UPDATE	DELETE

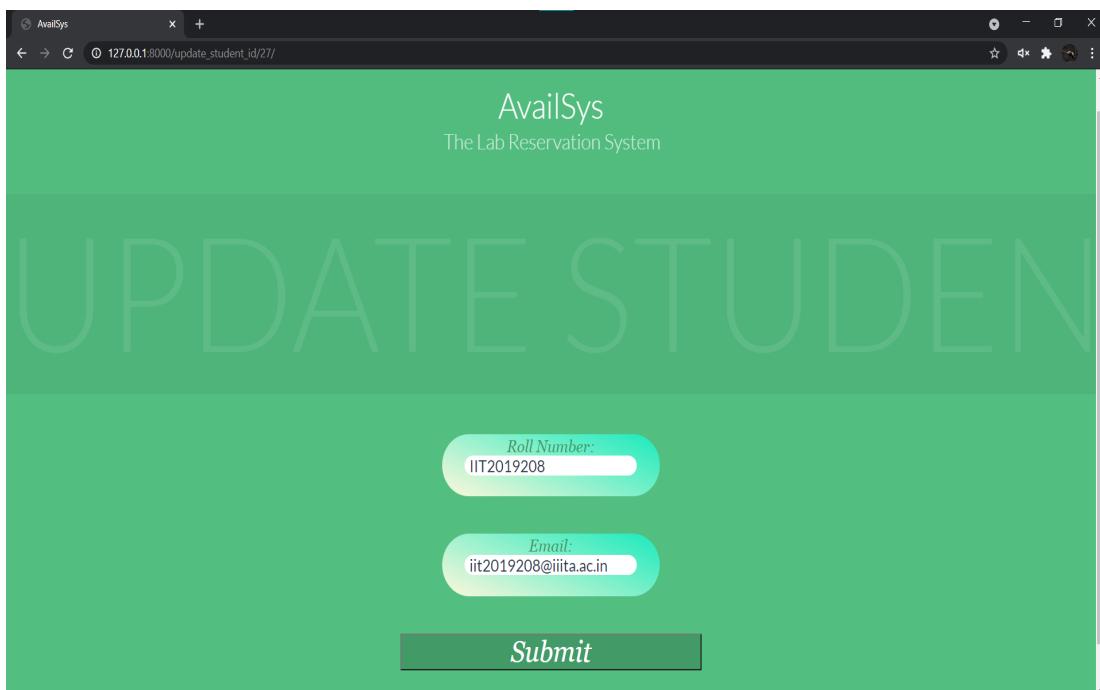
4.9 Delete Students-http://127.0.0.1:8000/delete_student/24/

To delete a student from the section , by selecting delete the deletion happens.



4.10 Update Student-http://127.0.0.1:8000/update_student_id/24/

Updation of students details whenever necessary.



4.11 View Sections - http://127.0.0.1:8000/view_section/

From the Dashboard of manage sections is selected , we reach this page.

The screenshot shows a web browser window titled 'AvailSys'. The URL in the address bar is '127.0.0.1:8000/view_section/'. The page has a green header with the text 'AvailSys' and 'The Lab Reservation System'. Below the header, the word 'VIEW SECTION' is displayed in large, semi-transparent letters. A table row is shown with the following data:

Section Id	Faculty	Remove
sem-IV-C	iit2019204	DELETE

4.12 Delete Section - http://127.0.0.1:8000/delete_section/8/

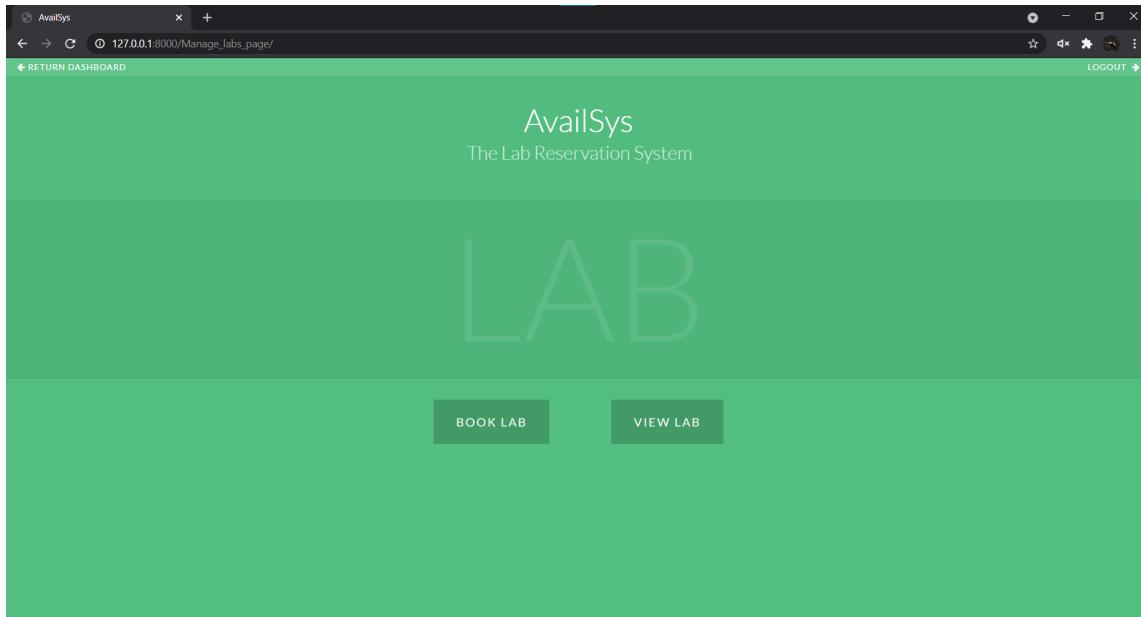
From the previous page , if we select delete we get redirected to this page.

The screenshot shows a web browser window titled 'AvailSys'. The URL in the address bar is '127.0.0.1:8000/delete_section/21/'. On the left, a modal dialog box titled 'Delete Section' asks 'Are you sure you want to delete your Section?'. It has 'Cancel' and 'Submit' buttons. On the right, the main content area displays the message 'Delete Section: sem-IV-C iit2019204' and a 'Delete' button.

Once the submit is selected , the section is deleted.

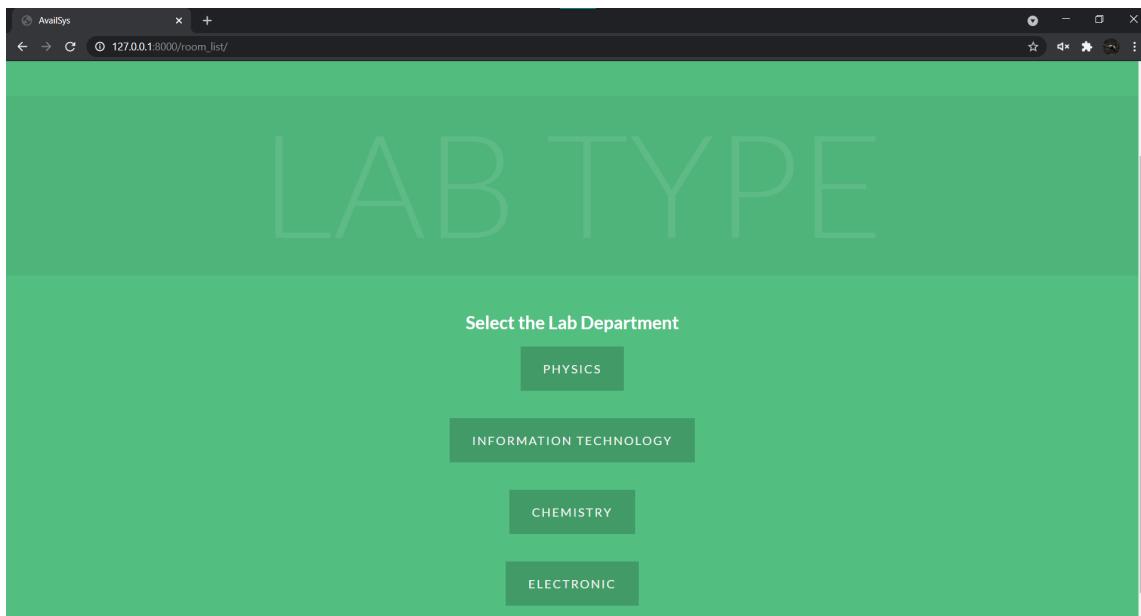
4.13 Manage Labs - http://127.0.0.1:8000/Manage_labs_page/

From the faculty dashboard , if faculty selects Manage labs we get redirected to this page.



4.14 Schedule a slot - http://127.0.0.1:8000/room_list/

From the previous page , if we select the book lab we get redirected to this page.From here we can select the type of lab we want (Department of the lab).



4.15 View labs - http://127.0.0.1:8000/booking_list/

From the Manage Labs dashboard , View labs button is selected.We get redirected to this page again.

The screenshot shows a web browser window for 'AvailSys' at the URL '127.0.0.1:8000/booking_list/'. The title bar says 'AvailSys' and the address bar shows the same URL. The page has a green header with the text 'AvailSys' and 'The Lab Reservation System'. Below the header, a large 'VIEW LABS' text is displayed. A table row is shown with the following data:

Lab Department	Room Number	From	To	Section	Action
IT	200	May 6, 2021, 10 a.m.	May 6, 2021, noon	sem-IV-C	CANCEL BOOKING

4.16 Manage Resources -

http://127.0.0.1:8000/Manage_resources_page/

From the Faculty Dashboard , if we select the Manage Resources button we get redirected to this page. To be able to access the features of this category.

The screenshot shows a web browser window for 'AvailSys' at the URL '127.0.0.1:8000/Manage_resources_page/'. The title bar says 'AvailSys' and the address bar shows the same URL. The page has a green header with the text 'AvailSys' and 'The Lab Reservation System'. Below the header, a large 'RESOURCES' text is displayed. At the bottom, there are two buttons: 'BOOK RESOURCE' and 'VIEW RESOURCES'.

4.17 Schedule Resources - http://127.0.0.1:8000/resource_book/

From the Manage Resources Dashboard , if we select the Schedule resources button . We get redirected here.We select the lab booked , type and number of resources.

The screenshot shows a web browser window titled 'AvailSys' with the URL '127.0.0.1:8000/resource_book/'. The main title is 'BOOK RESOURCE'. Below it, there is a section labeled 'Slot :'. A message says 'iit2019204 has booked 200. IT from 2021-05-06 10:00:00+00:00 to 2021-05-06 12:00:00+00:00~'. There are two dropdown menus: 'Resource : GPU' and 'Quantity : 3'. Below these is a button 'RETURN TO DASHBOARD (WARNING: REQUEST ABOVE WON'T BE ACCEPTED)'. At the bottom is a large green button labeled 'SAVE'.

4.18 Cancel Slot - http://127.0.0.1:8000/delete_booked_lab/22/

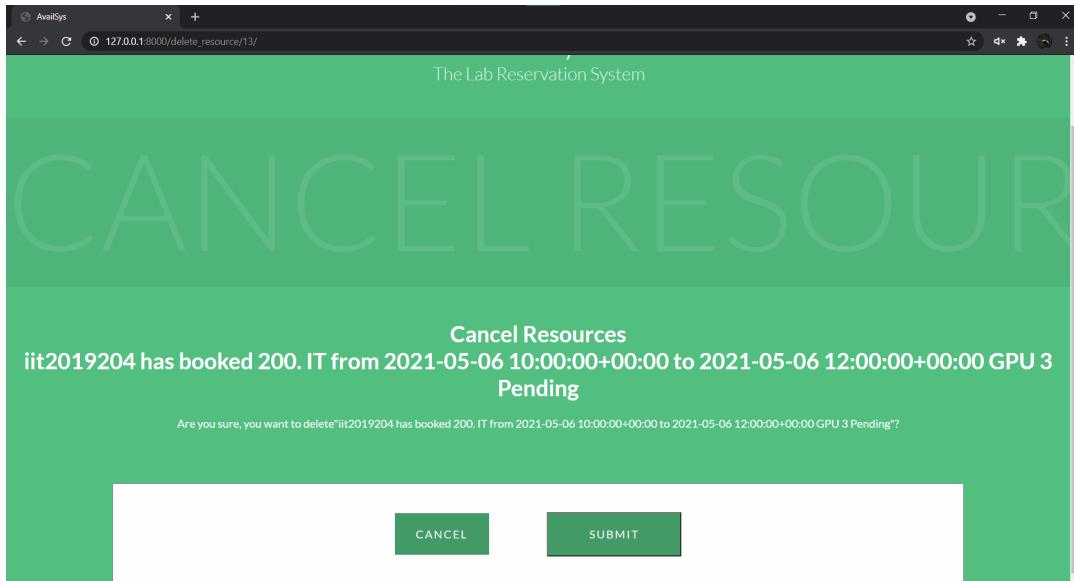
From the View labs page in the Manage labs dashboard , if selected cancel slot . We get redirected here.

The screenshot shows a web browser window titled 'AvailSys' with the URL '127.0.0.1:8000/delete_booked_lab/41/'. The main title is 'DELETE SLOT'. Below it, there is a section labeled 'Delete Slot'. A message asks 'Are you sure you want to delete this Scheduled Slot?'. It lists the following details:
Room Category : IT
Room Number : 200
Section ID : sem-IV-C
Check In : May 6, 2021, 10 a.m.
Check Out : May 6, 2021, noon

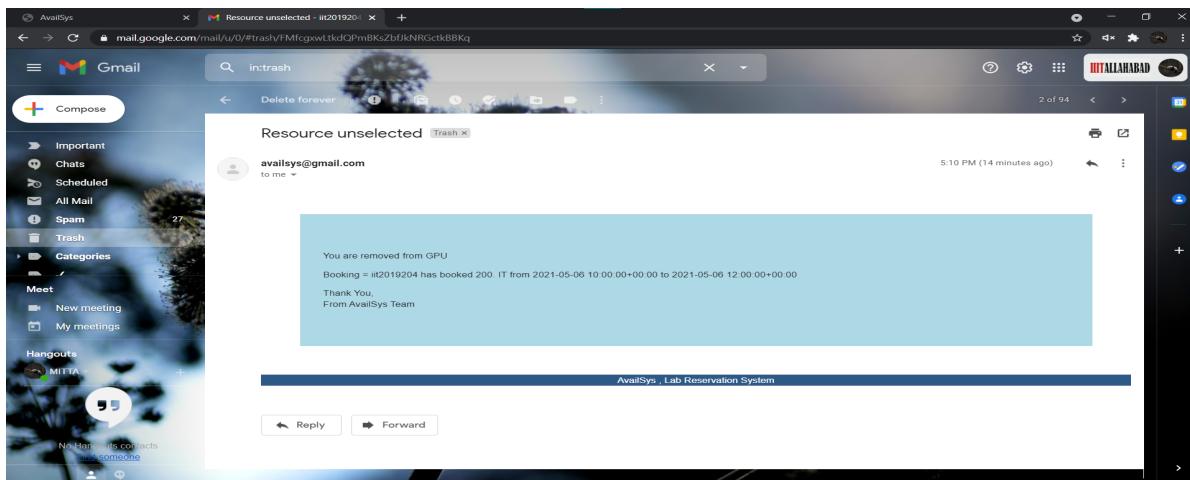
At the bottom are two buttons: 'RETURN' and 'CANCEL THE BOOKING'.

4.19 Cancel Resources - http://127.0.0.1:8000/delete_resource/11/

From the View resources page in the Manage resources dashboard , if selected cancel slot . We get redirected here.



4.20 Mail Notifications

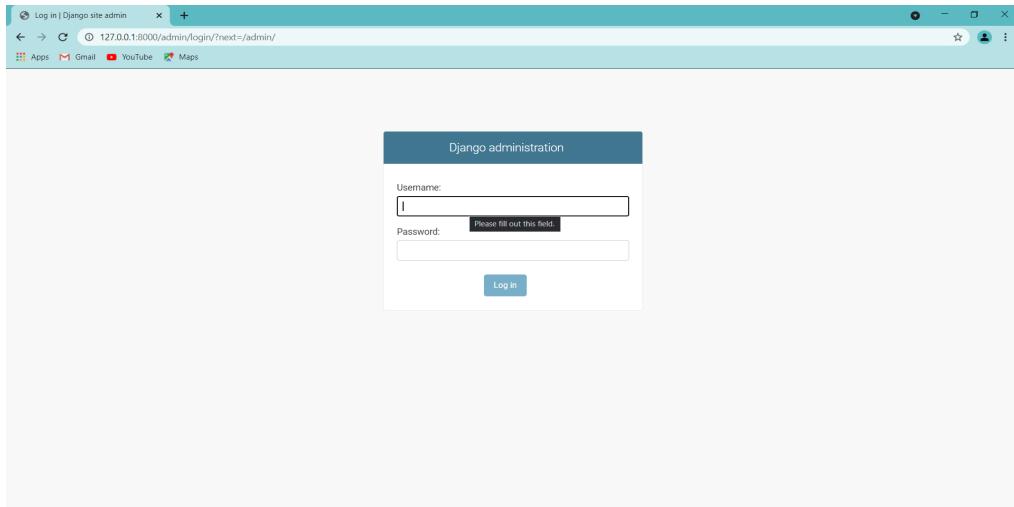


All types of notifications are in the similar way , when the lab is scheduled , resources booked , or cancelled we get similar notifications.

5 Admin features

5.1 Login - <http://127.0.0.1:8000/admin/login/?next=/admin/auth/user/>

Admin has to use their credentials to Login to the system and manage the working of the system.



5.2 Bookings - <http://127.0.0.1:8000/admin/app/booking/>

Admin after logging in they can check the labs booked,faculty and time slot booked.

A screenshot of the Django admin interface showing the "Bookings" list. The title bar says "Select booking to change | Django". The address bar shows "127.0.0.1:8000/admin/app/booking/". The top right says "WELCOME, ADMIN | VIEW SITE / CHANGE PASSWORD / LOG OUT". The left sidebar shows "ACCOUNTS", "APP" (with "Bookings" highlighted), "AUTHENTICATION AND AUTHORIZATION", "SITES", and "SOCIAL ACCOUNTS". The main content shows a table titled "Select booking to change" with 10 selected items. Each item is a checkbox followed by a booking detail: "Vasa has booked 200. IT from 2021-05-29 21:00:00+00:00 to 2021-05-04 22:00:00+00:00", "Iekhana has booked 200. IT from 2021-04-30 16:58:00+00:00 to 2021-04-30 17:58:00+00:00", "admin has booked 100. PHY from 2021-04-27 16:41:00+00:00 to 2021-04-27 17:41:00+00:00", "iit2019208 has booked 100. PHY from 2021-05-01 03:08:00+00:00 to 2021-05-01 03:08:00+00:00", "iit2019208 has booked 101. PHY from 2021-04-30 02:55:00+00:00 to 2021-04-30 06:55:00+00:00", "iit2019208 has booked 100. PHY from 2021-04-30 02:55:00+00:00 to 2021-04-30 06:55:00+00:00", "meghana has booked 300. CHEM from 2021-04-05 00:04:00+00:00 to 2021-04-05 02:04:00+00:00", "meghana has booked 101. PHY from 2021-04-05 01:01:00+00:00 to 2021-04-05 02:01:00+00:00", "meghana has booked 100. PHY from 2021-04-05 01:01:00+00:00 to 2021-04-05 02:01:00+00:00", and "meghana has booked 200. IT from 2021-04-05 01:59:00+00:00 to 2021-04-05 02:58:00+00:00". A "ADD BOOKING" button is at the top right of the list table.

5.3 Resource_bookings

http://127.0.0.1:8000/admin/app/resource_booking/

When the lab is booked then the corresponding faculty have to book the resources for that booked slot

The screenshot shows the Django admin interface for the 'Resource_bookings' model. The left sidebar lists various models: ACCOUNTS (Email addresses), APP (Bookings, Resource_bookings, Resources, Rooms, Sections, Student_links), AUTHENTICATION AND AUTHORIZATION (Groups, Users), SITES (Sites), and SOCIAL ACCOUNTS (Social accounts). The 'Resource_bookings' model is highlighted in yellow. The main content area is titled 'Select resource_booking to change'. It shows a list of two entries:

- RESOURCE_BOOKING
- Vasa has booked 200. IT from 2021-05-29 21:00:00+00:00 to 2021-05-04 22:00:00+00:00 GPU 80 Pending
- meghana has booked 200. IT from 2021-04-05 01:59:00+00:00 to 2021-04-05 02:58:00+00:00 String 33 Pending

Below the list, it says '2 resource_bookings'.

5.4 Resources-<http://127.0.0.1:8000/admin/app/resource/>

Here we can find all the resources which are available to allot for the labs in their respective category

The screenshot shows the Django admin interface for the 'Resources' model. The left sidebar lists various models: ACCOUNTS (Email addresses), APP (Bookings, Resource_bookings, Resources, Rooms, Sections, Student_links), AUTHENTICATION AND AUTHORIZATION (Groups, Users), SITES (Sites), and SOCIAL ACCOUNTS (Social accounts, Social application tokens). The 'Resources' model is highlighted in yellow. The main content area is titled 'Select resource to change'. It shows a list of five entries:

- RESOURCE
- Register 4 Omens
- HCI 1mole
- GPU
- String

Below the list, it says '4 resources'.

5.5 Rooms - <http://127.0.0.1:8000/admin/app/room/>

Here it manages the database of lab rooms for that particular category of lab type.

The screenshot shows the Django admin interface for the 'Rooms' model. The left sidebar lists various models: ACCOUNTS (Email addresses), APP (Bookings, Resource_bookings, Resources, Rooms, Sections, Student_links), AUTHENTICATION AND AUTHORIZATION (Groups, Users), SITES (Sites), and SOCIAL ACCOUNTS (Social accounts, Social application tokens). The 'Rooms' model is selected and highlighted in yellow. The main content area is titled 'Select room to change' and shows a list of 8 rooms with checkboxes. The rooms listed are: ROOM, 401. EEE, 400. EEE, 301. CHEM, 300. CHEM, 201. IT, 200. IT, 101. PHY, and 100. PHY. A button labeled 'ADD ROOM +' is visible in the top right corner.

5.6 Sections - <http://127.0.0.1:8000/admin/app/section/>

All the sections created by admin/faculty are shown here, which basically shows the database of sections

The screenshot shows the Django admin interface for the 'Sections' model. The left sidebar lists various models: ACCOUNTS (Email addresses), APP (Bookings, Resource_bookings, Resources, Rooms, Sections, Student_links), AUTHENTICATION AND AUTHORIZATION (Groups, Users), SITES (Sites), and SOCIAL ACCOUNTS (Social accounts, Social application tokens). The 'Sections' model is selected and highlighted in yellow. The main content area is titled 'Select section to change' and shows a list of 7 sections with checkboxes. The sections listed are: SECTION, Sem - 8 - C Vasa, sem-IV-C lekhana, y iit2019208, Sem-IV-E IIT2019208, Sem-IV-GH IIT2019208, Sem-IV-F IIT2019208, and sem-IV-C admin. A button labeled 'ADD SECTION +' is visible in the top right corner.

5.7 Student_links - http://127.0.0.1:8000/admin/app/student_link/

Here it gives the list of students with their respective email address and the section in which they were being added to.

The screenshot shows the Django administration interface for the 'student_link' model. The left sidebar lists various models: ACCOUNTS (Email addresses), APP (Bookings, Resource_bookings, Resources, Rooms, Sections, Student_links), AUTHENTICATION AND AUTHORIZATION (Groups, Users), SITES (Sites), and SOCIAL ACCOUNTS (Social accounts, Social application tokens). The 'Student_links' model is highlighted. The main content area is titled 'Select student_link to change' and shows a list of 5 selected entries. Each entry is a checkbox followed by the email address and section name, such as 'lekanamitta@gmail.com. sem-IV-C'. A 'Go' button and a note '0 of 5 selected' are at the top of the list. A 'ADD STUDENT_LINK +' button is located in the top right corner.

5.8 Groups - <http://127.0.0.1:8000/admin/auth/group/add/>

We can make groups of students in order of simple identification.

The screenshot shows the Django administration interface for creating a new group. The left sidebar is identical to the previous screenshot, with 'Groups' highlighted. The main content area is titled 'Add group' and has a 'Name:' input field. Below it, a 'Permissions' section contains two panels: 'Available permissions' and 'Chosen permissions'. The 'Available permissions' panel lists numerous permission codes like 'account | email address | Can add email address' and 'admin | log entry | Can add log entry'. The 'Chosen permissions' panel is currently empty. At the bottom, there are buttons for 'Save and add another', 'Save and continue editing', and a large 'SAVE' button.

5.9 Users - <http://127.0.0.1:8000/admin/auth/user/>

All the users i.e., faculty along with admin are shown here.

The screenshot shows the Django Admin interface for managing users. The left sidebar has sections for ACCOUNTS, APP, AUTHENTICATION AND AUTHORIZATION, SITES, and SOCIAL ACCOUNTS. Under AUTHENTICATION AND AUTHORIZATION, 'Users' is selected. The main area displays a table titled 'Select user to change' with columns: USERNAME, EMAIL ADDRESS, FIRST NAME, LAST NAME, and STAFF STATUS. The table lists seven users: Dhanush, IIT2019208, Vasa, admin, iit2019208, lekhana, and meghana. The 'STAFF STATUS' column indicates that Dhanush, Vasa, admin, and meghana are staff members (green checkmarks), while IIT2019208, iit2019208, and lekhana are not (red X's). A 'FILTER' sidebar on the right allows filtering by staff status (All, Yes, No), superuser status (All, Yes, No), and active status (All, Yes, No).

Action:	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	Dhanush	IIT2019208@iita.ac.in			✓
<input type="checkbox"/>	IIT2019208	vasadhanush99@gmail.com			✗
<input type="checkbox"/>	Vasa	vasadhanush99@gmail.com			✗
<input type="checkbox"/>	admin	iit2019234@iita.ac.in			✓
<input type="checkbox"/>	iit2019208	iit2019208@iita.ac.in			✗
<input type="checkbox"/>	lekhana	iit2019204@iita.ac.in			✗
<input type="checkbox"/>	meghana	meghana@gmail.com			✗

6 Student Features

6.1 Receive Email

The screenshot shows an email inbox with a single message from 'availsys@gmail.com'. The message subject is 'Booking Confirmation' and it was sent 'to me, lekhnamitta' at '11:22 AM (27 minutes ago)'. The message body contains the following text:

Your booking is successful.
Faculty : lekhnaMitta
category : PHY
room no. : 100
section : sem-4
Start Time : April 6, 2021, noon
End Time : April 6, 2021, 1:30 p.m.
Thank You,
From AvailSys Team

At the bottom of the email, there is a footer bar with the text 'AvailSys , Lab Reservation System'.