**Chapter 4 :- System Design**

**4.1 Basic Module**

Project contains two webapp's developed using flask and python3. Database used : MySQL community edition. For face reocognition I used python3 "face\_recogntion" by ageitgey.(<https://github.com/ageitgey/face_recognition>), Built using dlib's state-of-the-art face recognition built with deep learning. The model has an accuracy of 99.38% on the Labeled Faces in the Wild benchmark.

For spoof detection I used tensorflow inception model by retraining it's last layer so that it can detect mobile phones in an image. To generate and manage excel I used xlrx and xlrd and pandas.

For sending email's I used flask-mail.

admin site dependency : flask, mysqlclient, sklearn, numpy, scipy, pillow, dlib, face\_recognition

teachers site dependency : flask\_bootstrap, pytz, xlsxwriter, pandas, flask\_mail, tensorflow,xlrd

**Admin Site:-**

**Home Page Module:-** Web application starts with home page module. Navigation bar contain three things Admin login , Teacher registration and contact page. It is static page.

**Admin Login Module:-** Admin is the super user. To access the Student Registration page Admin need to login first with Register Credentials.

**Teacher Registatrion module:-** Admin can also register teachers using this site.

**Student Registration and Model training Module:-** The whole concept is at the time of admission to college or school admin should register the students details such as his name email address and also create training data of each student by entering his roll id and taking snaps of his or her frontal face and then webapp will automatically create model for that particular roll id and save it on server, The model which is created for each student is about 8kb in size.

**Teachers Site:-**

**Teacher Login Page Module:-** Now using teacher's site (It will be used when teacher will actually enter the class), teacher has to login first to access the Attendace module.

**Attendance Module:-**Then after clicking on attendance tab there will be no back button as teacher will pass on the phone to student. Student's will then have to just click a snap enter class and roll id and press enter to mark their attandance.

After that there is also a problem of spoof attack in face recognition i.e. someone will show someone's image of face through their mobile phone and trick our webapp and they will mark the attendace of their friends who were not present.

But I solved this issue using tensorflow, by training inception to detect mobile phones in an image then I used that model in the webapp as soon as student's click a snap it will first check if the face is spoof or original. To retrain inception's last layer I used 200 images of mobile phones and I feed them to tensorflow to retrain the last layer of inception.

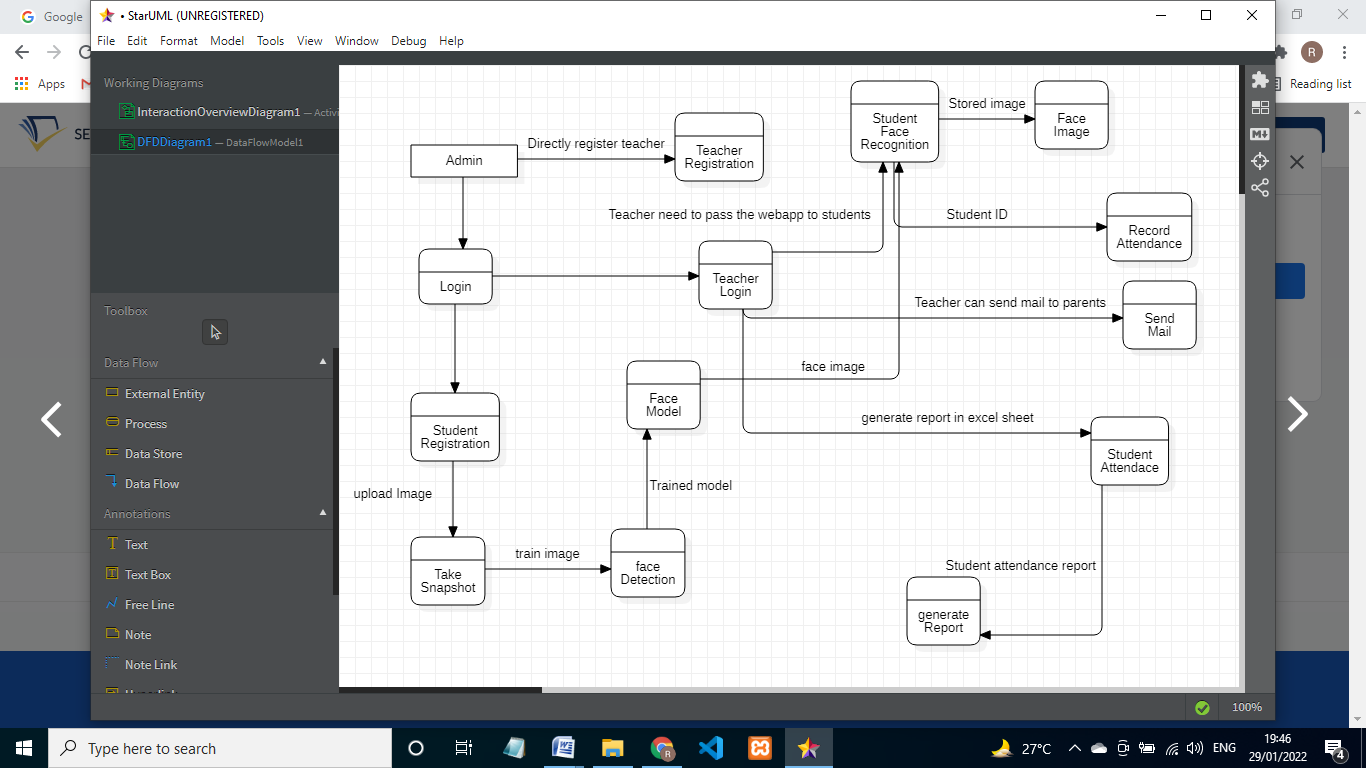
**View Report module:-** Now after student's part is done. Teacher can then login again and then go to report's tab to see attendance. Here there are several option's I have given.

If teacher want to see today's attendace, just select date and time to see the attendance. And there is also an option to download the attendace sheet in excel form and then again reupload it after making any changes if sometime required by the teacher. And the teacher can also see total attendance for his or her lecture. so that they can analyze how many lectures each student from particular class had attended so far.

**Send Mail module:-** One additional feature is that teacher can send email for the attendance marked to all the parents as well as students by selecting class and clicking on send mail button.

**4.2: Data Design:-**

**4.2.1 Schema Design:-**



**4.2.2 data Integrity and constraints.**

**Admin**

|  |  |  |  |
| --- | --- | --- | --- |
| Fields | Datatypes | Null | Value |
| username | Varchar | No | 30 |
| Password | varchar | No | 20 |

**Teacher Registration:-**

|  |  |  |  |
| --- | --- | --- | --- |
| Fields | Datatypes | Null | Value |
| Username/name | Varchar | No | 50 |
| Password | varchar | No | 50 |
| email | varchar | No | 50 |

**Student registration**

|  |  |  |  |
| --- | --- | --- | --- |
| Fields | Datatypes | Null | Value |
| Username/name | Varchar | No | 30 |
| Password | varchar | No | 50 |
| Student email | varchar | No | 50 |
| Parent email | varchar | No | 50 |
| Roll id | varchar | No | 10 |