Ex no :11

MINI PROJECT

SMART ATTENDANCE SYSTEM

1. PROBLEM STATEMENT

Develop a software that replaces the conventional attendance system which is time consuming, ambiguous and requires human effort. The software developed should be accurate and efficient in use and maintenance unlike the pen and paper method.

2. REQUIREMENTS SPECIFICATION

2.1 FUNCTIONAL REQUIREMENTS

- Adding new students- we could easily add their details in the database provided a key user .
- Report students-they are now able to check on their current attendance situation through the system.
- Report faculties - Teachers update the attendance in case of any on-duty(OD),exception or medical reason.

2.2 NON – FUNCTIONAL REQUIREMENTS

2.2.1 Performance Requirements :

- The important aspects of the system is time constrain.
- In real time it should be performed with minimum requirements.
- The accountability is a vital feature and this could only be assured if the system is working in full capability.
- So uninterrupted internet connectivity is needed.

2.2.2 Security Requirements

- The security system features from having a login for all the users to access the software.
- The login details will be used in the system also.
- So the chances of the software getting intruded are very less.

2.2.3 Software Quality Attributes

- Adaptability & Reusability : The source code of the product is going to be open as this is going to be open source software. It will be free for further modifications and improvements.

- Availability & Usability : The system is usable when there is internet facility.
- Flexibility & Maintainability : The system is flexible to changes in the student entries and faculty changes and henceforth is easy to maintain .

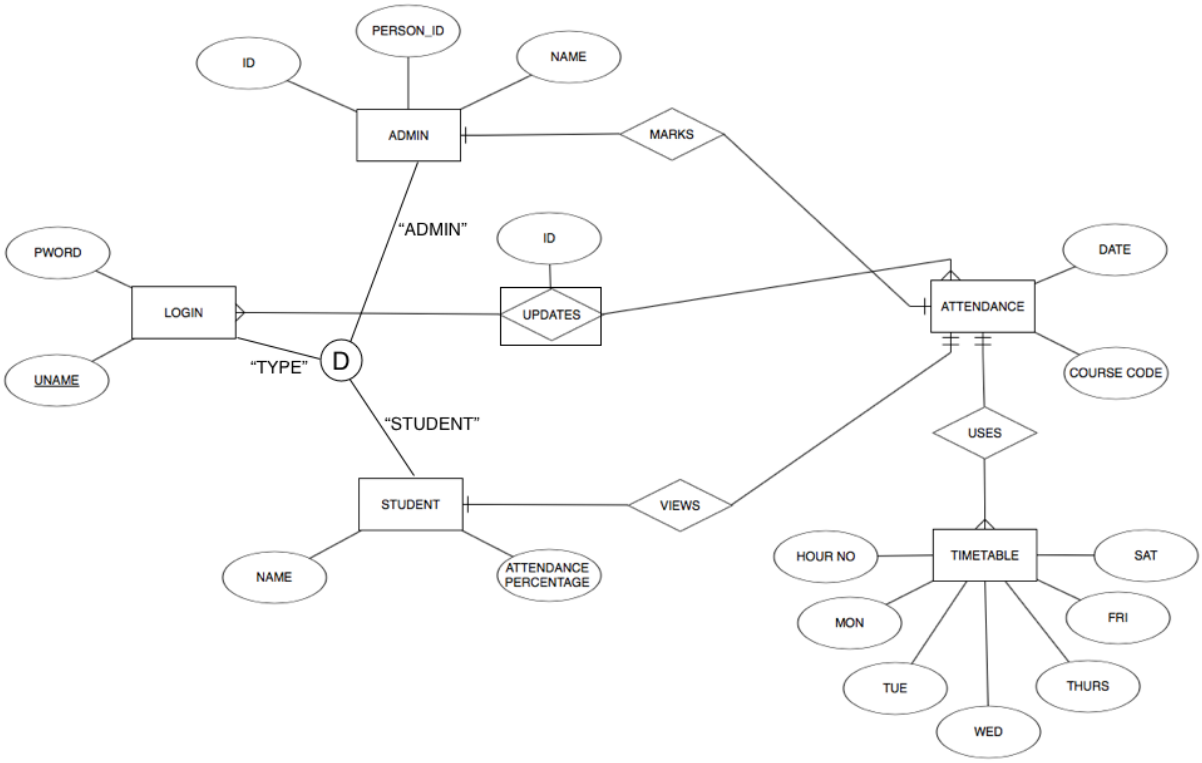
2.3 SOFTWARE REQUIREMENTS :

- Microsoft cognitive services for face API
- opencv with numpy, scipy implemented in python
- Tkinter module for user interface
- SQLite for database

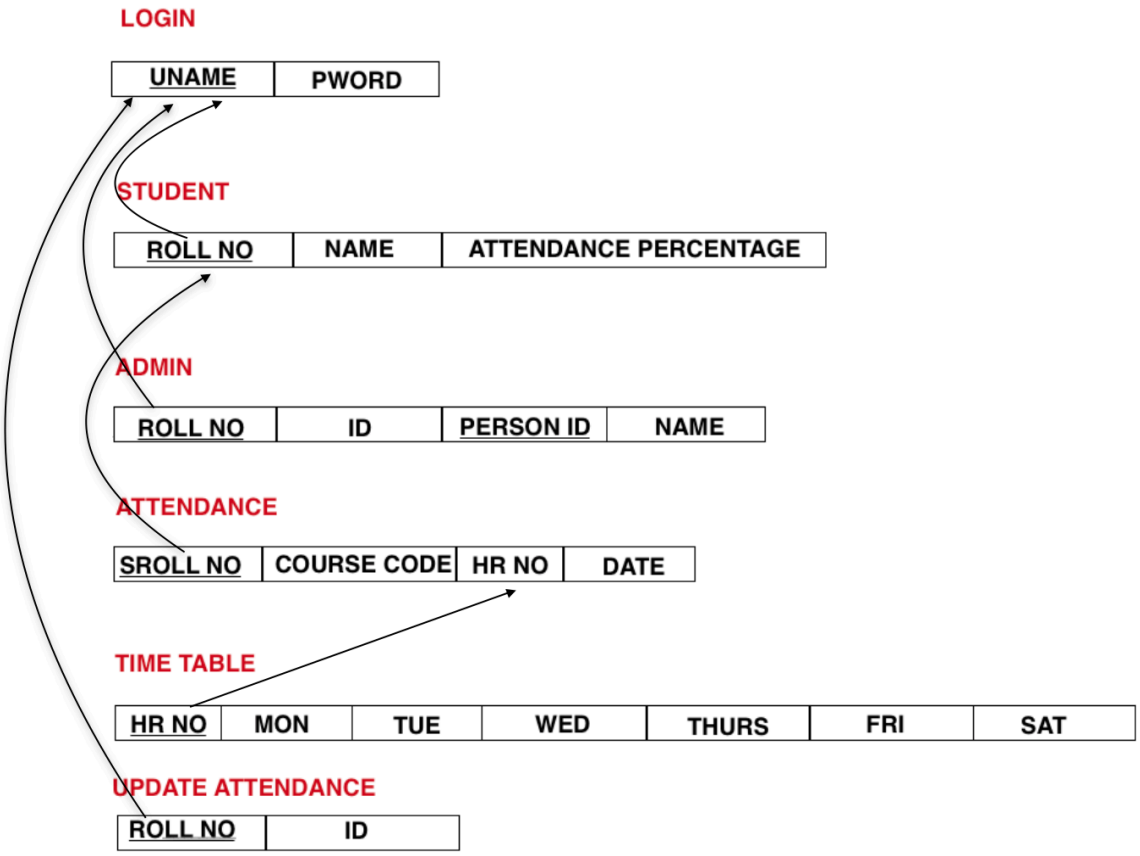
2.4 HARDWARE REQUIREMENTS:

- Laptop with camera.

3. ENTITY RELATIONSHIP MODEL :



4 .TRANSFORMATION TO RELATIONAL MODEL :



ER MODEL COMPONENT	TRANSFORMATION RULE
STUDENT ENTITY (UNMAE TO ROLL NO)	DISJOINT RULE
ADMIN ENTITY (UNAME TO ROLL NO)	DISJOINT RULE
STUDENT : ATTENDANCE (ROLL NO TO SROLLNO)	1:M RELATIONSHIP RULE
UPDATE ATTENDANCE (UNAME TO ROLL NO)	M:N RELATIONSHIP ENTITY TYPES TRANSFORMATION RULE
ATTENDANCE TO TIMETABLE (HOUR NO)	MANY TO ONE REALTIONSHIP RULE
STRONG ENTITY TYPES	TRANSFORMED TO TABLES AND THEIR ATTRIBUTES AS COLUMNS

5. NORMALIZATION

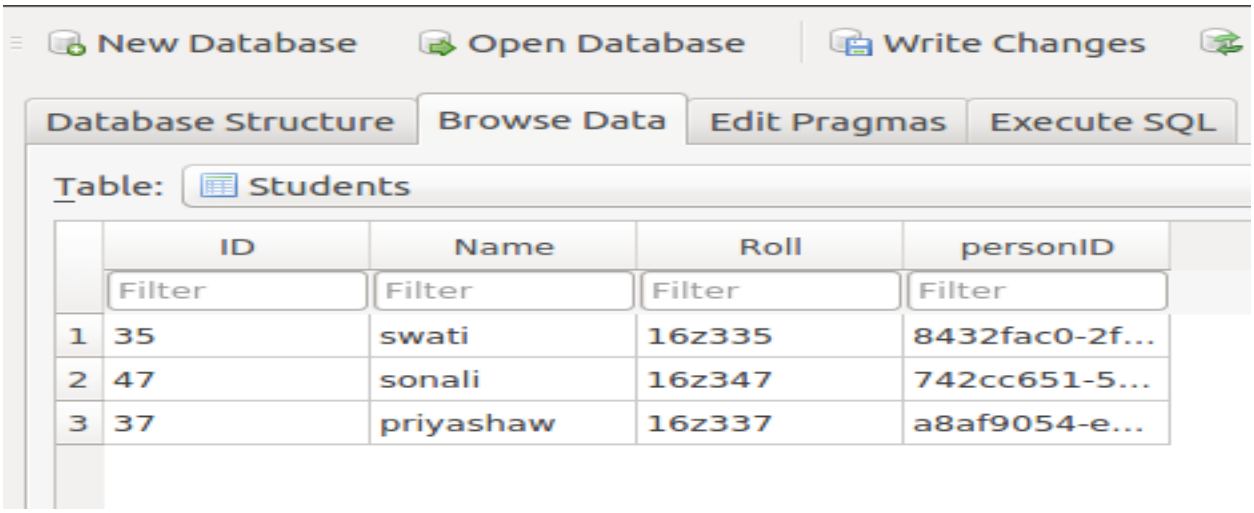
- ROLL NO -> ID
- ROLL NO, NAME -> PERSON ID, ID

There are no repeating groups, therefore, table is in 1NF form.

By removing Functional Dependencies, We Obtain 2NF form.

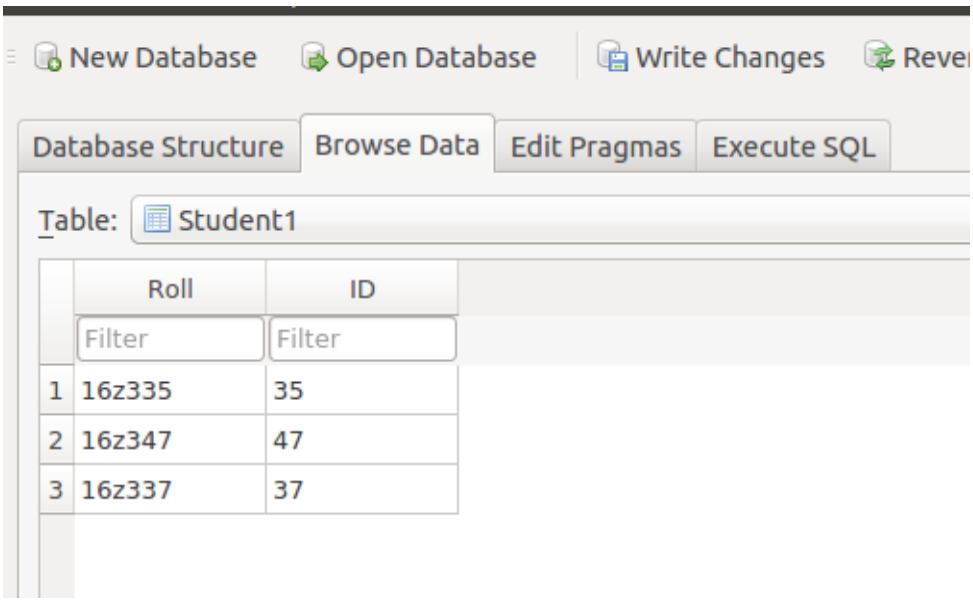
- ROLL NO -> ID
- ROLL NO, NAME -> PERSON ID

BEFORE NORMALIZATION:

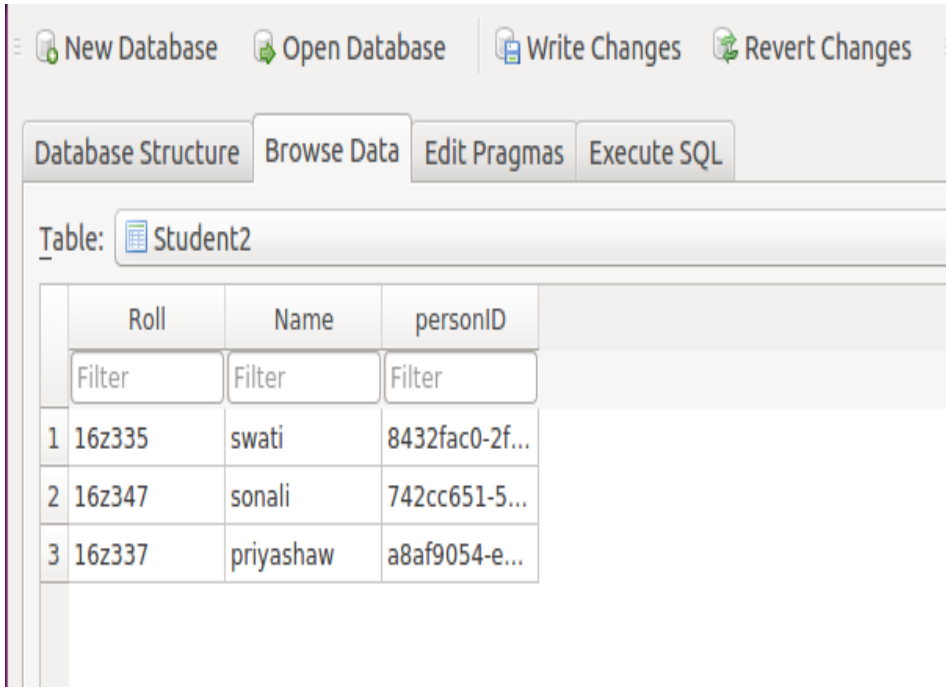


	ID	Name	Roll	personID
	Filter	Filter	Filter	Filter
1	35	swati	16z335	8432fac0-2f...
2	47	sonali	16z347	742cc651-5...
3	37	priyashaw	16z337	a8af9054-e...

AFTER NORMALIZATION:



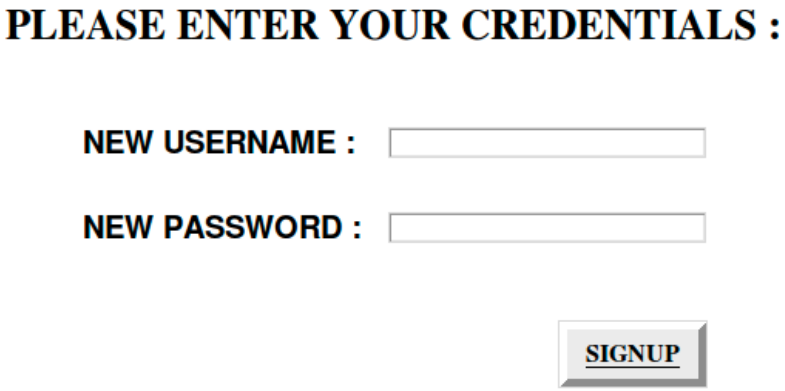
	Roll	ID
	Filter	Filter
1	16z335	35
2	16z347	47
3	16z337	37



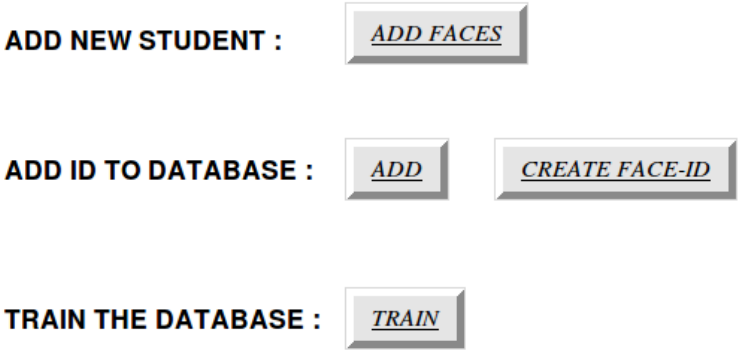
6. UI MOCKUPS AS PER WORK FLOW

LOGIN PAGE:

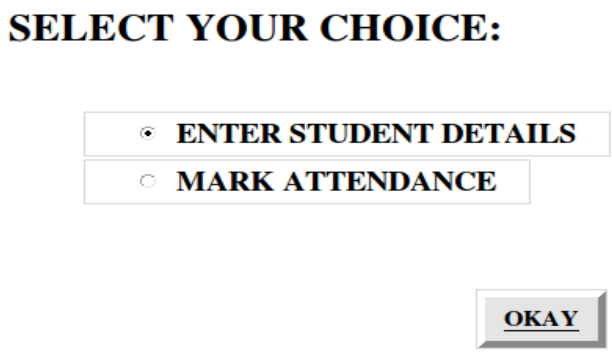
SIGN-UP PAGE :



ADMIN PAGE 1 :



ADMIN OPTIONS PAGE :



TEACHER'S PAGE :

ROLL NUMBER : 16z335

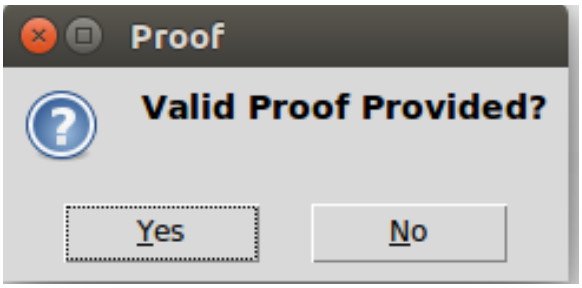
DAY : 23

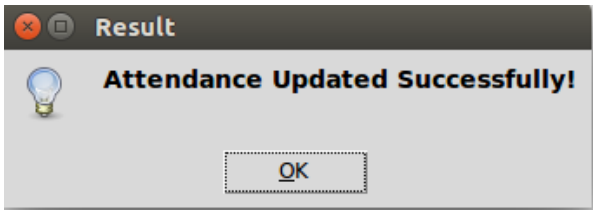
MONTH : april

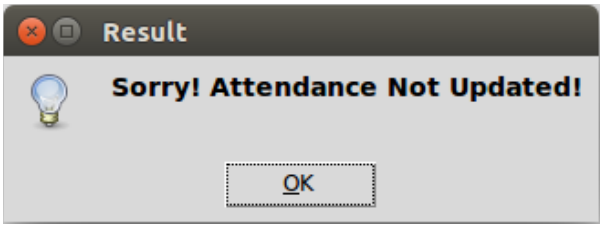
YEAR : 2018

UPDATE

TEACHER'S PAGE :







7. PLATFORMS FOR IMPLEMENTATION