**A**

**Project Report On**

**FRAM**

**Prepared by**

14CE110 Vignesh Patel

14CE117 Urvish Rana

14CE120 Kushal Reshamdalal

**Under the guidance of**

Prof. Nikita Bhatt

Prof. Gaurang Panchal

**Submitted to**

Charotar University of Science & Technology

Degree of Bachelor of Technology

in Computer Engineering

CE324 -Software Group Project

Of 6th Semester of B.Tech

**Submitted at**



**U. & P. U PATEL DEPARTMENT OF COMPUTER ENGINEERING**

**Faculty of Technology & Engineering, CHARUSAT**

**Chandubhai S. Patel Institute of Technology**

**At: Changa, Dist: Anand – 388421**

**Jan-June 2017**



**CERTIFICATE**

This is to certify that the report entitled “FRAM” is a bonafied work carried out by **Mr. Vignesh Patel (14CE110)**, **Mr. Urvish Rana (14CE117), Mr. Kushal Reshamdalal (14CE120)** under the guidance and supervision of **Prof. Nikita Bhatt and Prof. Gaurang Panchal** for the subject **Software Group Project (CE324)** of 6th Semester of Bachelor of Technology in **Computer Engineering** at Faculty of Technology & Engineering (C.S.P.I.T.) – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate themselves, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

|  |  |
| --- | --- |
| Under the supervision of,  **Prof. Nikita Bhatt**  **Prof. Gaurang Panchal**  Assistant Professor  U. & P. U Patel Dept. of Computer Engg.  C.S.P.I.T., CHARUSAT-Changa. |  |
| Dr. (Prof.) Amit Ganatra  Dean,  Faculty of Technology & Engineering  Head, U. & P. U Patel Department of Computer Engineering  C.S.P.I.T., CHARUSAT- Changa, Gujarat. | |

**Chandubhai S Patel Institute of Technology (C.S.P.I.T.)**

**Faculty of Technology & Engineering, CHARUSAT**

**Acknowledgement**

We are highly intended to the following personalities who have helped us throughout our project and without their support this project would never have been completed. We are extremely thankful to them. We are thankful to our Department HOD **Dr. Amit Ganatra,** who has provided us resources from the college and we are also very thankful to our project guide **Asst. Prof. Nikita Bhatt and Gaurang Panchal** who has provided her time and never ending support. For their support and help we heartily thank them. This project would be incomplete without them.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table of Contents**  **Abstract……………………………………………………………………………..**  **Acknowledgement………………………………………………………………….**   1. **Introduction…………………………………………………………………….**    1. Project Summary……………………………………………………….........    2. Purpose…………………………………………………………….………...    3. Scope……………………………………………………………….………..    4. Objectives…………………………….……………………………………...    5. Technology and Literature Review…….……………………...…………… 2. **Project Management**    1. Project Planning   2.1.1 Project Development Approach and Justification……………  2.1.2 Project Effort and Time, Cost Estimation………………………   * + 1. Roles and Responsibilities……………....……….....……….....   1. Project Scheduling (Gantt Chart/PERT/Network Chart)……….………  1. **System Requirements Study**   3.1 User Characteristics …….…………………………………………….  3.2 Hardware and Software .………………………………………………  3.3 Assumptions and Dependencies.………..…………………………….   1. **System Analysis**   4.1 Study of Current System .………..…………………………………..  4.2 Problem and Weaknesses of Current System ……………………….  4.3 Requirements of New System  4.3.1 Functional Requirements……………………..……………..  4.3.2 Non Functional Requirements……………………………….  4.4 Feasibility Study   * + 1. Does the system contribute to the overall objectives of the organization? ………………………………………….……..     2. Can the system be implemented using the current technology and within the given cost and schedule constraints? ………...     3. Can the system be integrated with other system which are already in place? ……………………………………………..   1. Activity/Process In New System……………………………………..   2. Features Of New System……………………………………………..   3. Class Diagram………………………………………………………..   4. System Activity………………………………………………………   5. Sequence Diagram…………………………………………………...  1. **System Design**    1. System Application Design       1. Method Pseudo code………………………………………….    2. Database Design/Data Structure Design       1. Table and Relationship……………………………………….       2. Logical Description Of Data………………………………….    3. Input/Output and Interface Design       1. State Transition/UML Diagram………………………………       2. Samples Of Forms, Reports and Interface…………………… 2. **Implementation Planning**    1. Implementation Environment …………………………………………..    2. Program/Modules Specification…………………………………………    3. Security Features………………………………………………………..    4. Coding Standards……………………………………………………….. 3. **Testing**    1. Testing Plan ……………………………………………………………...    2. Testing Strategy…………………………………………………………..    3. Testing Methods………………………………………………………….    4. Test Cases…………………………………………..……………………. 4. **Limitation and Future Enhancement**    1. Limitations……………………………………………………………….    2. Future Enhancements…………………………………………………….   **9 Conclusion and Discussion**   * 1. Problem Encountered and Possible Solutions………………………………..   9.2 Summary of Project work………………………………………………......... | 3  4  7  7  7  8  8    3  5  5  5    7  7  7  8  8  8  9    10  10  10  10  11  11  12  13  15  16  19  20  21  26  26  26  27  28  28  28  28  32  32  33  33 |  |  |  |

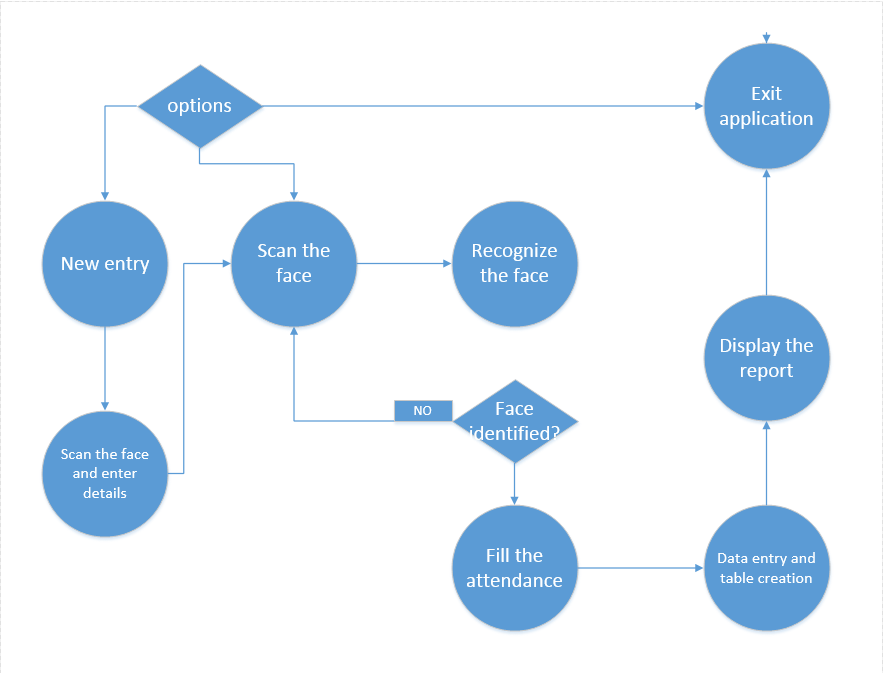
1. **INTRODUCTION**
   1. **PROJECT SUMMARY**

* Project is based on filling attendance by face recognition.
* Each and every student can fill their attendance.
* Built for making attendance process easier.
  1. **PURPOSE**
* This web app detects the face of student, recognizes who is the student and then fills the attendance of respective student in the records.
* The main purpose of this web app is to fill the attendance easier just by showing our face.
  1. **SCOPE**
* This web app is somewhat similar compared to other face recognize apps available but adding few more ideas. Adding more ideas, we can record the attendance and also we can generate report based on the attendance.
* There are two type of user 1. Faculty 2. Students
* Faculties people can add remove students and fill their attendance and generate report via daily, weekly, monthly and semester vise.
* Student can only see attendance on daily basis. They able to see report if particular faculty gives them permission. Otherwise they can only see before one week of internals.

* 1. **OBJECTIVE**
* The main objective of making this application is to remove the hectic process to fill the attendance one by one.
* Hence saving the time and workload.
* The system is automatic hence it reduces the paperwork and workload of taking attendance of respective students.
* Hence now onwards students will not be able to fill proxy of others.
  1. **TECHNOLOGY AND LITERATURE REVIEW**
* This project is made in OpenCV, Netbeans and the language is Python, Java, MySQL. [1]
* This application will be run in Chrome compactible at any version.
* OpenCV is released under a BSD license and hence it’s free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform.[1]
* A Java servlet is a Java program that extends the capabilities of a server. Although servlets can respond to any types of requests, they most commonly implement applications hosted on Web servers. Such Web servlets are the Java counterpart to other dynamic Web content technologies such as PHP and ASP.NET.[3]

1. **OVERALL DESCRIPTION**
   1. **PRODUCT PERSPECTIVE**

* Following diagram shows perspective of product**.**



**Fig 2.1 perspective of product**

* 1. **PRODUCT FUNCTIONS**
* The web app will firstly detect our face by enabling the camera by itself when that site will be opened. Then after detecting the face, the app will automatically recognizes the face by comparing the data with the database available. Then app will fill the attendance of that student in the attendance record and then teachers can generate attendance report based on that database.

**2.3 USER CHARCTERISTICS**

* Generally this type of web app can be accessed by both teachers as well as student.
* This web application can detect our face, recognize it and fill the attendance in database.

**2.4 CONSATRAINTS**

* This web app is not the only web app which detects and recognize the face. And this app may not be that efficient as the other apps available today.
* The web app is implemented in several languages. Java, CSS, HTML, MySQL, Python. Also, to make this app work; XAMPP, OpenCV, Notepad is used. Hence increasing complexity to make this project work and understand.

**2.5. ASSUMPTIONS AND DEPENDENCIES**

* The web app will be fully functional and error free, running without any crash.
* The web app remains stable and compatible with Windows 7 and greater.

1. **SYSTEM STUDY**
   1. **REQUIREMENT**
      1. **Functional Requirement**

* **SCAN THE FACE:**

This application should scan the face when we place our face towards camera.

**Input**: Our face.

**Output:** The application scans the face provided as an input.

* **IDENTIFY THE FACE:**

When the face is scanned, the application should recognize who’s face it is.

**Input:** Our face.

**Output:** Identify who’s face it is.

* + 1. **Non Functional Requirement**

**LOADING:**

Face should be identified fast.

**DETAIL:**

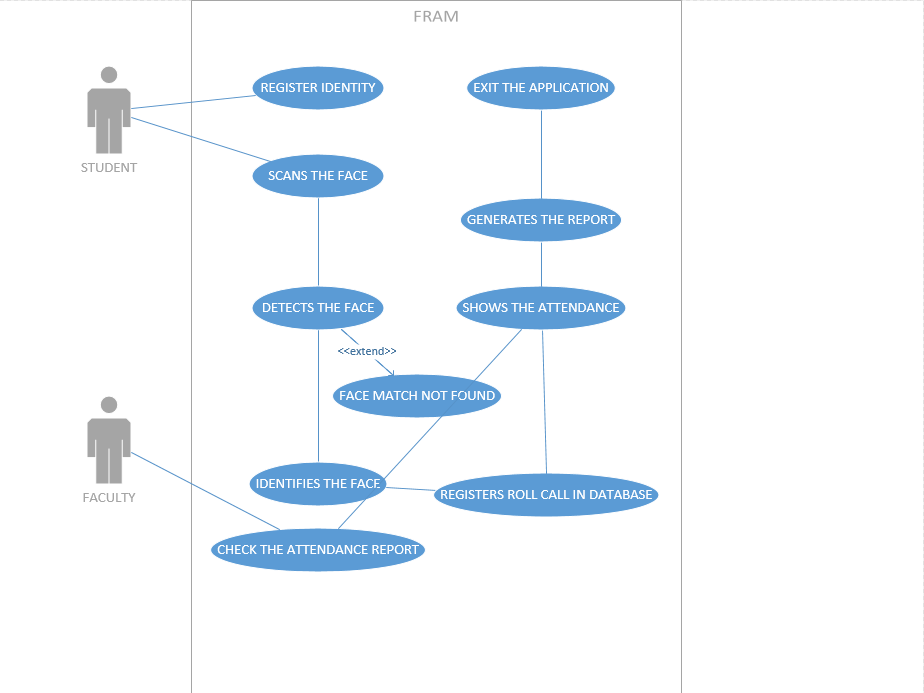
Application should provide more details of the student.

**UI:**

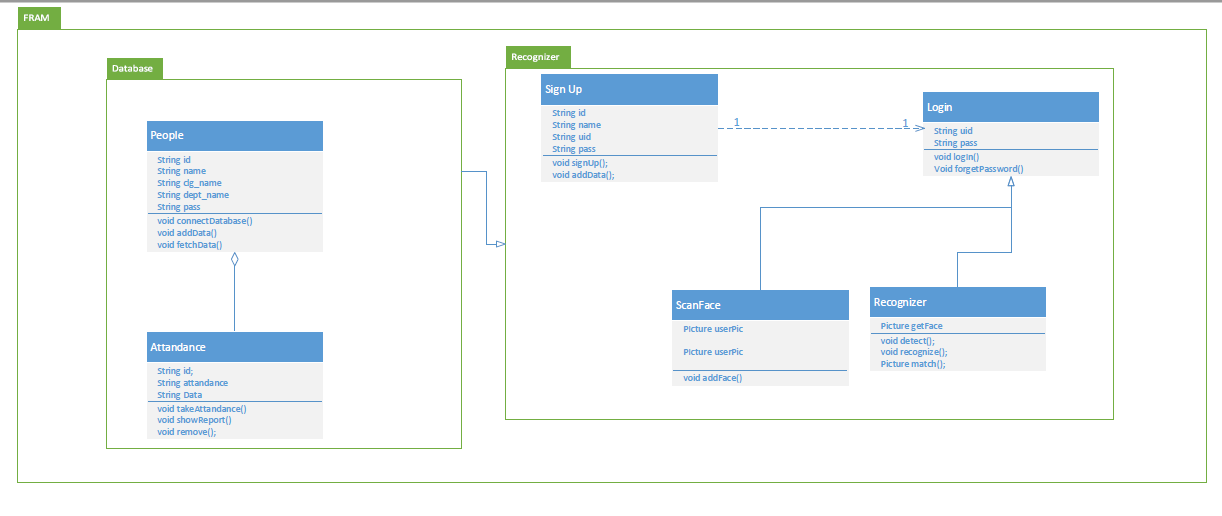
It should be easily understandable, in other words, it should be handy.

**3.2 FEASIBLITY STUDY**

* **Does the system contribute to the overall objectives of the organization?**
  + YES
* **Can the system be implemented using the current technology and within the given cost and schedule constraints?**
  + YES
* **Can the system be integrated with other system which are already in place?** 
  + NO

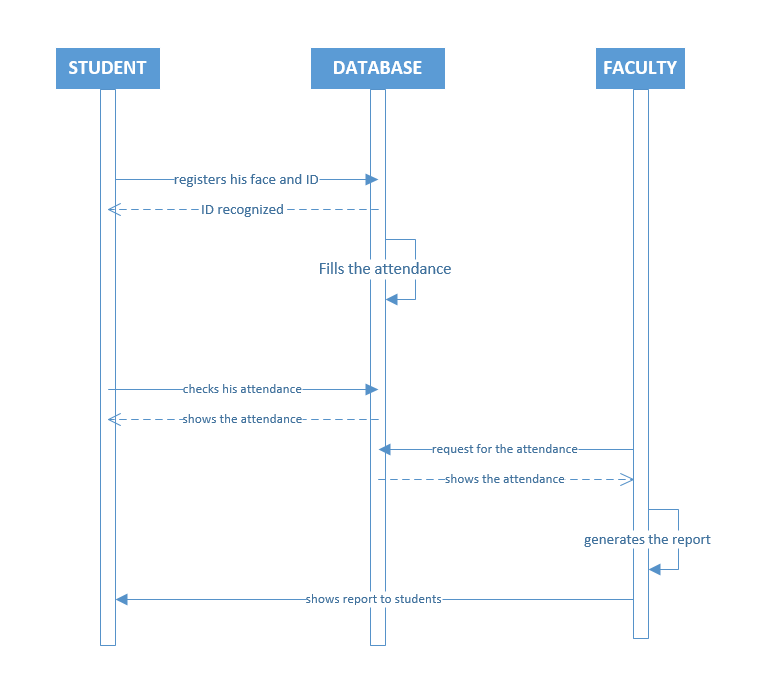
1. **SYSTEM ACTIVITY(Use Case)** 

**3.2 CLASS DIAGRAM**

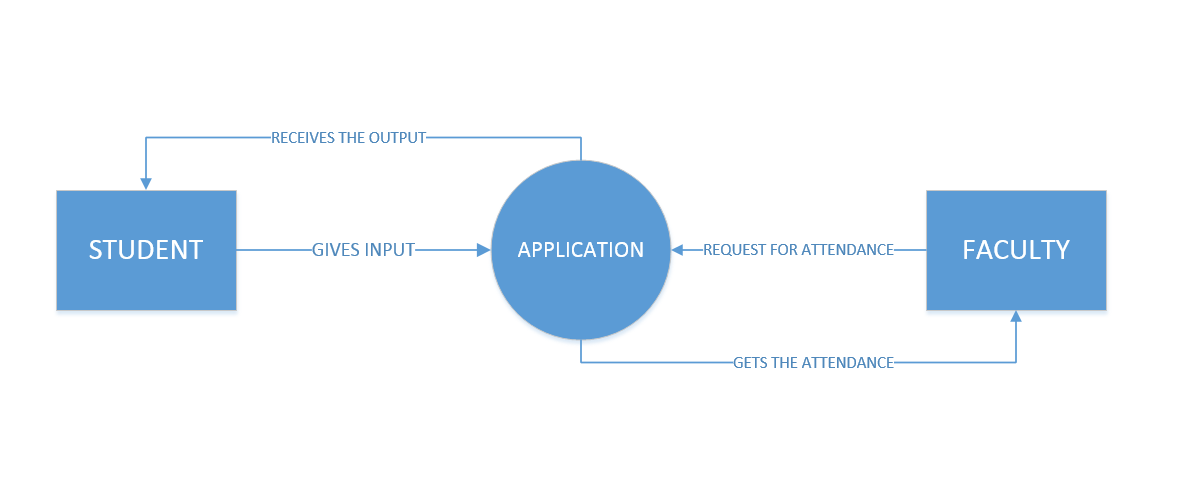


**FIG 3.2 CLASS DIAGRAM**

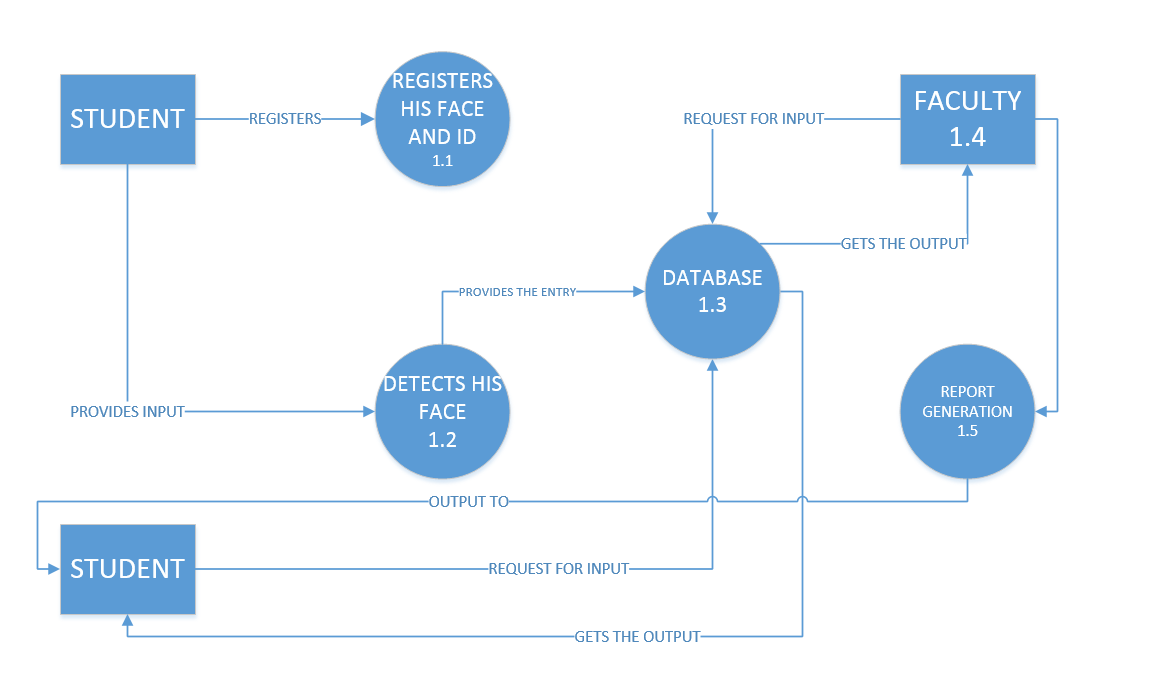
* 1. **SEQUENCE DIAGRAM**



* 1. **sequence diagram**
  2. **CONTEXT DIAGRAM**



* 1. **Context diagram**

**3.5 DFD DIAGRAM**

**3.5 DFD diagram**

1. **PROJECT ESTIMATION**
   1. **ESTIMATION METHOD USED[2]**

* We used cocomo method for our project.
* Formula :FP=count total\*[0.65+ (0.01\* Fi)]
* Function point count for FRAM.

Table 4.1 General System characteristics

|  |  |
| --- | --- |
| Adjustment factor | value |
| 1. Does the system require reliable backup and recovery? | 4 |
| 2. Are data communications required? | 2 |
| 3. Are there distributed processing functions? | 3 |
| 4. Is performance critical? | 3 |
| 5. Will the system run in an existing, heavily utilized operational environment? | 2 |
| 6. Does the system require on-line data entry? | 0 |
| 7. Does the on-line data entry require the input transaction to be built over multiple screens or operations? | 0 |
| 8. Are the master file updated on-line? | 0 |
| 9. Are the inputs, outputs, files, or inquiries complex? | 0 |
| 10. Is the internal processing complex? | 3 |
| 11. In the code designed to be reusable? | 1 |
| 12. Are conversion and installation included in the design? | 4 |
| 13. Is the system designed for multiple installations in different organizations? | 3 |
| 14. Is the application designed to facilitate change and ease of use by the user? | 5 |
| Total | 30 |

|  |  |  |
| --- | --- | --- |
| Type of components | Complexity of components | Value |
| External inputs | 2\*4 | 12 |
| External outputs | 1\*5 | 20 |
| External inquiries | 1\*4 | 4 |
| External logical files | 1\*10 | 10 |
| External interface files | 1\*7 | 7 |
| Total | 34 |

Table 4.2 Unadjusted function point

* Formula
* FP = count total\*[0.65+ (0.01\*ΣFi)]
* = 34\*[0.65+ (0.01\*30)]
* = 32.3
* Function point count for FPS Game= 58.28
* Lines of code=32.3\*30=969=0.969 KLOC
* Estimation of effort using Cocomo Model:

a=2.4; b=1.05

Effort =a\*(KLOC) b

=2.4\*(0.969)1.05

=2.24 person-month

* Estimation of time using Cocomo Model:

Type of Project = Organic. Then, c=2.5; d=0.38

Time = (Effort) d \* c

= (2.24) 0.38 \* 2.5

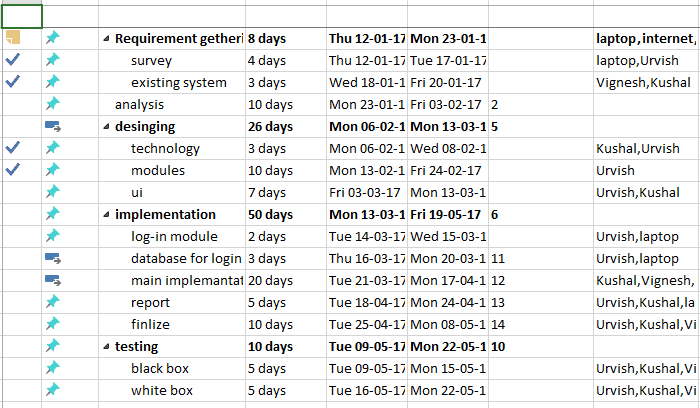
= 2.13 months

* Estimation of Cost using Cocomo Model:

Cost = 2.13 \* 5000

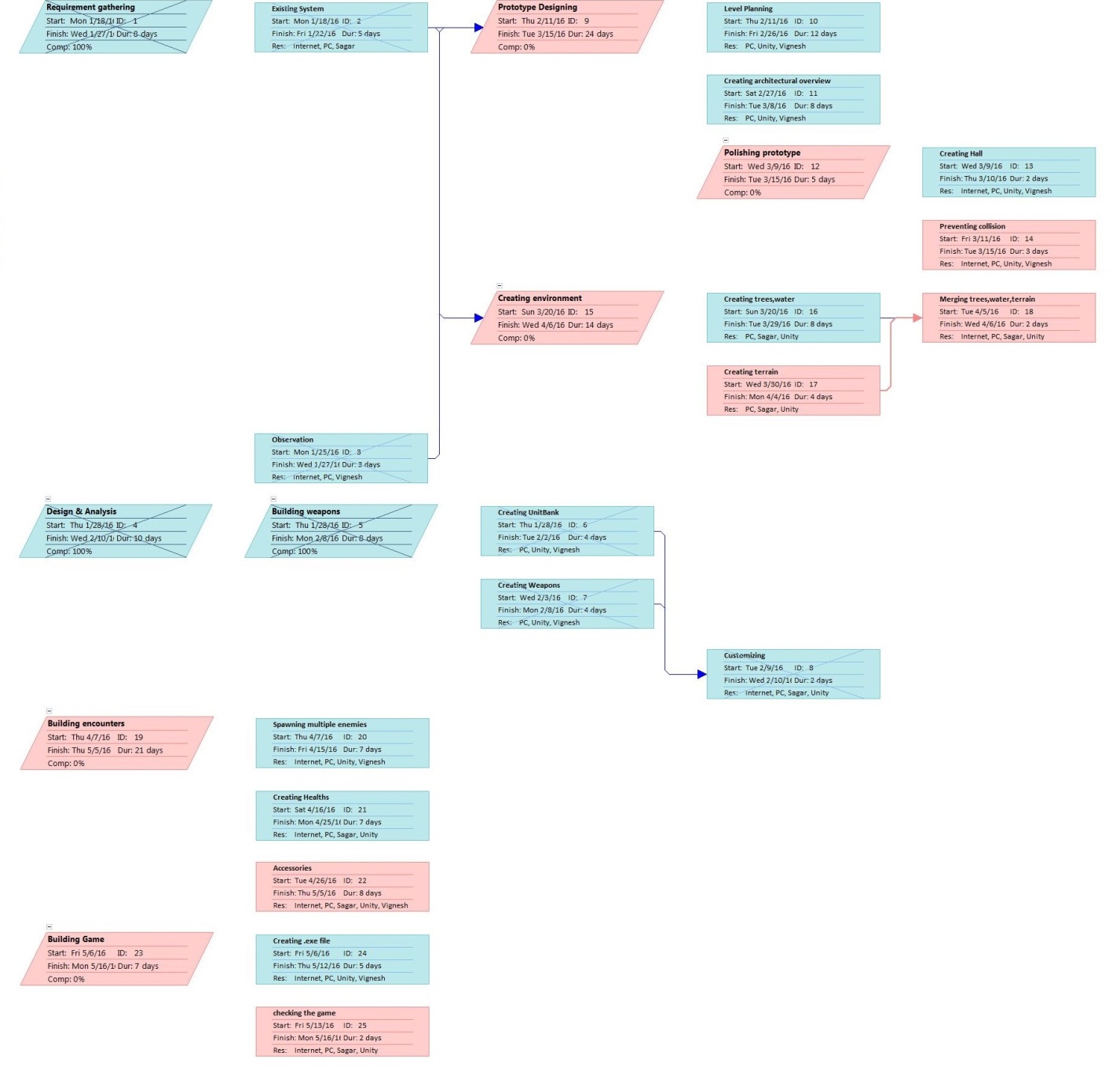
= Rs. 10,611

1. **SCHEDULE IN WHICH**
   1. **BREAKDOWN STRUCTRE**



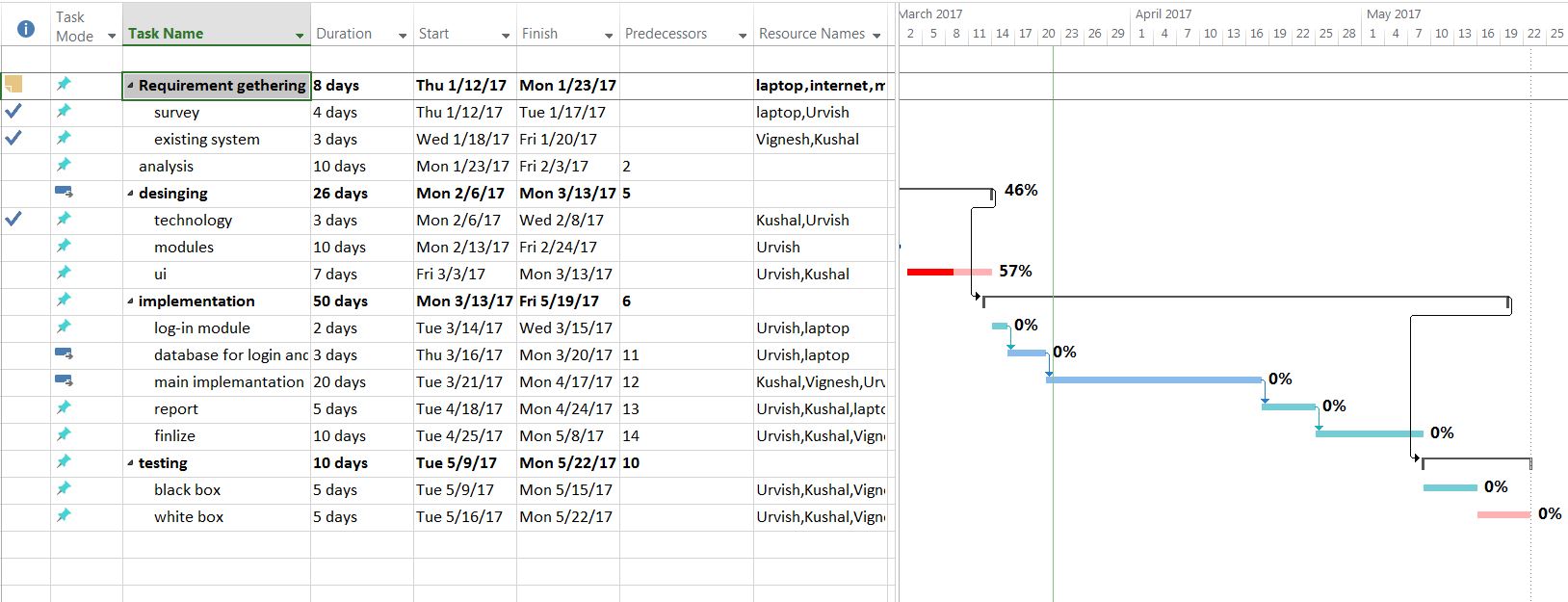
**FIG 5.1 Breakdown structure**

* 1. **TASK NETWORK**



**FIG 5.2 Task Network**

* 1. **GANTT CHART**



**FIG 5.3 GANTT CHART**

1. **PROJECT RESOURCE** 
   1. **PEOPLE**

|  |  |  |
| --- | --- | --- |
| **PEOPLE** | **TASK** | **ROLE** |
| Mr. Vignesh Patel | designing, backend, database connectivity, documentation, frontend. | Developer |
| Mr. Urvish Rana | designing, backend, database connectivity, testing, frontend. | Developer |
| Mr. Kushal Reshamdalal | database connectivity, testing, frontend. | Developer |

* 1. **HARDWARE AND SOFTWARE**
* Minimum requirements

|  |  |
| --- | --- |
| Os | Windows 7.0 and above |
| Ram | 512mb |
| Storage | 1gb |

1. **SYSTEM DESIGN**

**7.1. SYSTEM APPLICATION DESIGN**

* + 1. **Method Pseudo code**

import cv2

import numpy as np

import sqlite3

faceDetect = cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml');

cam = cv2.VideoCapture(0);

rec = cv2.createLBPHFaceRecognizer();

rec.load("recognizer\\training.yml")

atm1=0

def getProfile(id):

conn = sqlite3.connect("FaceBase.db")

cmd = "SELECT \* FROM People WHERE ID="+str(id)

cursor = conn.execute(cmd)

profile = None

for row in cursor:

profile=row

conn.close()

return profile

procedure= """

asdfgh

fdsgh

adgfshnj

dgfsdjf

"""

def setAt(ids):

conn = sqlite3.connect("FaceBase.db")

cursor = conn.cursor()

for id in ids:

cmd1= "SELECT att FROM People WHERE id="+str(id)

cursor1 = conn.execute(cmd1)

print cmd1

for row in cursor1:

atm1=row[0]

print atm1

for id in ids:

atm1=atm1+1

cmd="UPDATE People SET att="+str(atm1)+" WHERE id="+str(id)

cursor.execute(cmd)

print("hello got ",id,cmd);

conn.commit()

conn.close()

id=0

flag=set()

font = cv2.cv.InitFont(cv2.cv.CV\_FONT\_HERSHEY\_COMPLEX\_SMALL,2,1,0,2)

while(True):

ret,img = cam.read();

gray = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

faces = faceDetect.detectMultiScale(gray,1.3,5);

for(x,y,w,h) in faces:

cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)

id,conf = rec.predict(gray[y:y+h,x:x+w])

profile = getProfile(id)

if(profile!=None):

cv2.cv.PutText(cv2.cv.fromarray(img),str(profile[1]),(x,y+h+30),font,255)

cv2.cv.PutText(cv2.cv.fromarray(img),str(profile[2]),(x,y+h+60),font,255)

cv2.cv.PutText(cv2.cv.fromarray(img),str(profile[3]),(x,y+h+90),font,255)

cv2.cv.PutText(cv2.cv.fromarray(img),str(profile[4]),(x,y+h+120),font,255)

print("id",id)

flag.add(id)

cv2.imshow("Face",img);

if(cv2.waitKey(1) == ord('q')):

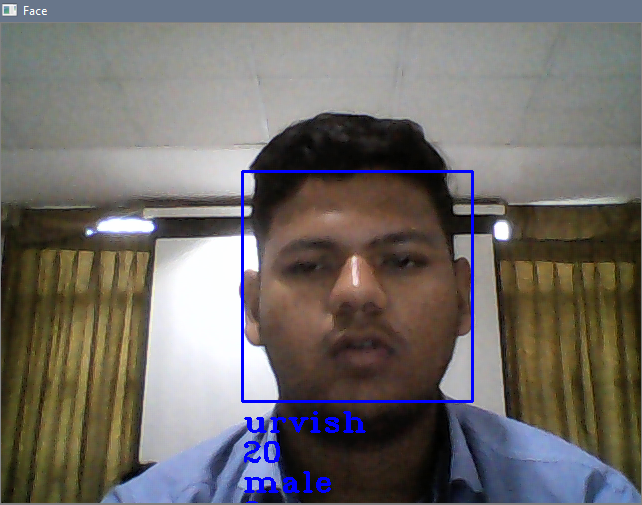
break;

setAt(flag)

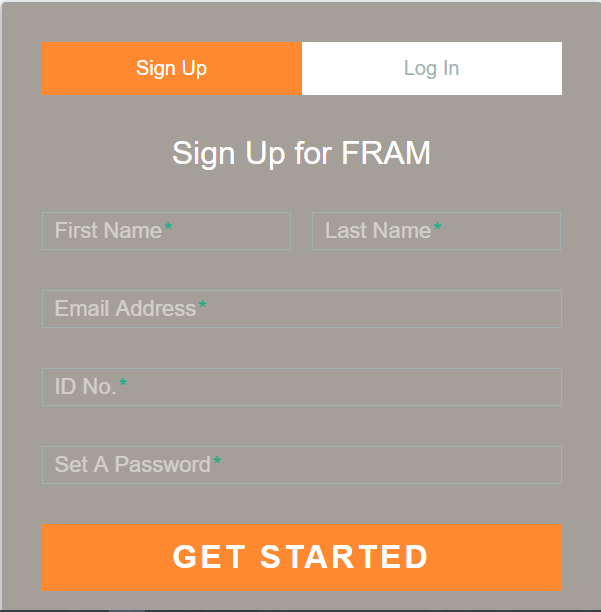
cam.release()

cv2.destroyAllWindows()

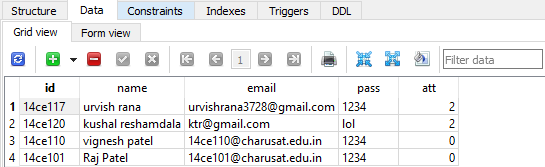
* + 1. **Interface**

****

**FIG 7.1 START UP**



**FIG 7.2 LOGIN PAGE**



**FIG 7.3 DATATBASE**

**8. TESTING**

**8.1 TESTING PLAN**

* We are doing two types of testing here black box and white box.

**8.2 TEST CASE DESIGN**

* **Register function test suit:**

Test Suites No: 1

Test Suite Detail: User wants to register his/her face and ID.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Function Name** | **Test Case (condition)** | **Expected Results** | **Actual Result** | **Pass/Fail** |
| 1 | registernew | User clicks the ‘register’ button | New entry should be done | New entry actually gets done. | Pass |

* **Student scans his/her face.**

Test Suites No: 1

Test Suite Detail: Student scans his/her face to get his/her attendance filled.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Function Name** | **Test Case (condition)** | **Expected Results** | **Actual Result** | **Pass/Fail** |
| 1 | entry | Student presents his/her face | Attendance should be filled | Attendance gets filled | Pass |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Function Name** | **Test Case (condition)** | **Expected Results** | **Actual Result** | **Pass/Fail** |
| 2 | entry | Student presents his/her face without registering | Attendance should not be filled | Attendance doesn’t get filled | Pass |

* **Student wants to see the attendance.**

Test Suites No: 1

Test Suite Detail: Student wants to see his/her attendance of so far.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Function Name** | **Test Case (condition)** | **Expected Results** | **Actual Result** | **Pass/Fail** |
| 1 | checkattendance | Student tries to see his/her attendance | Attendance upto this far should be shown | Attendance upto this far is shown | Pass |

* **Faculty wants to see the attendance.**

Test Suites No: 1

Test Suite Detail: Faculty wants to get the attendance of all students along with ID and their other data this far.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Function Name** | **Test Case (condition)** | **Expected Results** | **Actual Result** | **Pass/Fail** |
| 1 | attendanceid | Faculty wants to get the attendance of all students along with their other data | Attendance along with other data should be shown. | Attendance along with other data is shown. | Pass |

Test Suites No: 2

Test Suite Detail: Faculty generates the report of each and every student.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Function Name** | **Test Case (condition)** | **Expected Results** | **Actual Result** | **Pass/Fail** |
| 1 | generatereport | Faculty wants to generate the report | Report should be generated | Report gets generated | Pass |

1. **LIMITATION AND FUTURE ENHANCEMENTS**

**9.1 LIMITATIONS**

* There might be a possibility that sometimes the face isn’t get scanned.
* Also, sometimes some unusual change in face like specs, beard shave might not detect the face.

**9.2 FUTURE ENHANCEMENTS**

* We will try to improvise the face detection better.
* Application will catch and recognize more number of faces once.

**10. CONCLUSION AND DISCUSSION**

**10.1 Self-Analysis of Project Viabilities**

* From this project we studied about face detection and recognition

**10.2 Problem Encountered and Possible Solutions**

* OpenCV was not able to recognize the face of student earlier.
  + **Solution:** specific changes to recognizer set has been done.
* Database connectivity was not done profoundly.
  + **Solution:** specific modules are now added and appropriate modules are replaced.

**10.3 Summary of Project work**

* This project is about web app development. This app is developed for student purpose. This app will help to fill the attendance easier just by showing our face.

**BIBLIOGRAPHY**

**WEBSITE TOPIC**

**[1].www.opencv.org image processing library**

**[2].www.getcocomoonline.com cocomo model**

**[3]. www.en.wikipedia.com java servlet**