

# SMART ATTENDANCE

DBMS PROJECT 2015

*Gurvinder Singh (U12C0062)*

*Aadesh Bagmar (U12C0092)*

*Deepak Singh (U12C0063)*

# TABLE OF CONTENTS

## Contents

Abstract	1
Relational Schemas	2
Process Flow and Entity-Relational Model	3
Screenshots	4

# ABSTRACT

## THE IDEA IN BRIEF

Attendance Monitoring is a monotonous though important exercise performed by students and teachers every day. Moreover, reliability over a manual system is always in serious doubt! In this project we present a simple system which uses face recognition to mark a student's presence.

Further, we can query over the database to check if a student's attendance is over a particular limit or not. Thus, it automates and simplifies the process of attendance management.

## TECHNOLOGIES USED

1. **OpenCV C++** libraries to implement Face Recognition.
2. **Qt C++** for the GUI and backend processing.
3. **Sqlite** for managing the Database.

## HIGHLIGHTS AND NOVELTY

It includes open text fields for the administrator to simplify writing the queries. It also includes a "Write your own Query" box, so that a user can customize queries to generate output as per the needs.

It also, considers the fact if a student leaves a class within 40 minutes, his attendance is not marked. It notes the time when a student enters the class and leaves it and marks his attendance in the classes scheduled in that time slot.

Thus, it relieves students and teachers of marking attendance and helps prevent proxies. An automatic system makes it easier to find critical data like number of classes attended by a student of a particular subject by a particular professor, etc.

## LOOKING AHEAD

The project is in Alpha version currently. Once it passes Beta testing, it can be implemented at various locations around our own campus, maybe starting with our Computer Engineering Department!

---

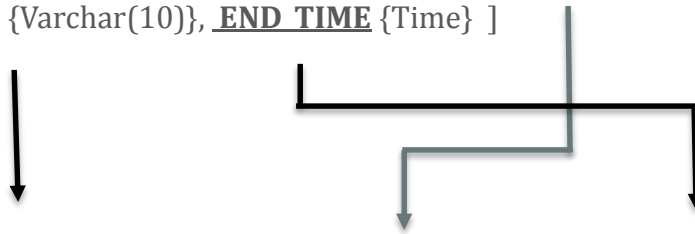
# RELATIONAL SCHEMA

## TABLES

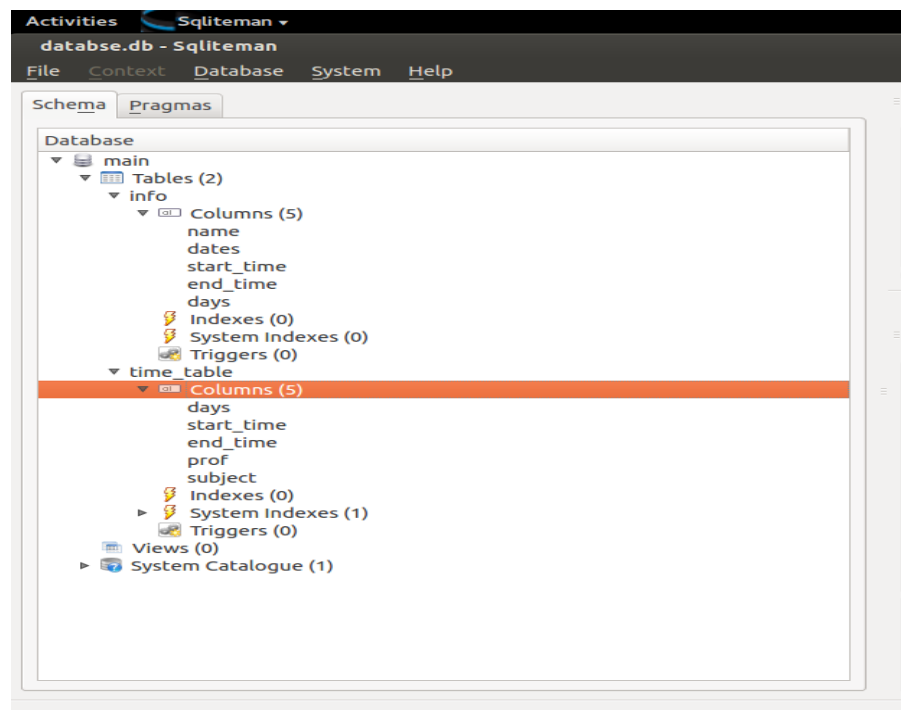
**INFO:** Used to store daily information of students.

**TIME\_TABLE:** A table to record the current Time Table for a particular division.

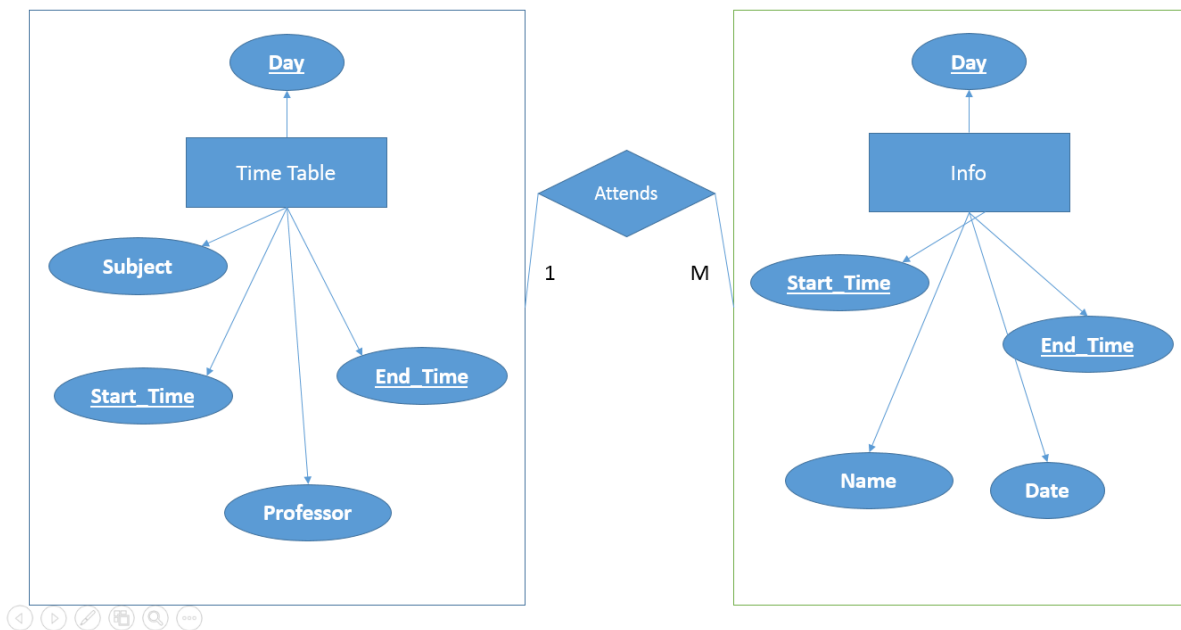
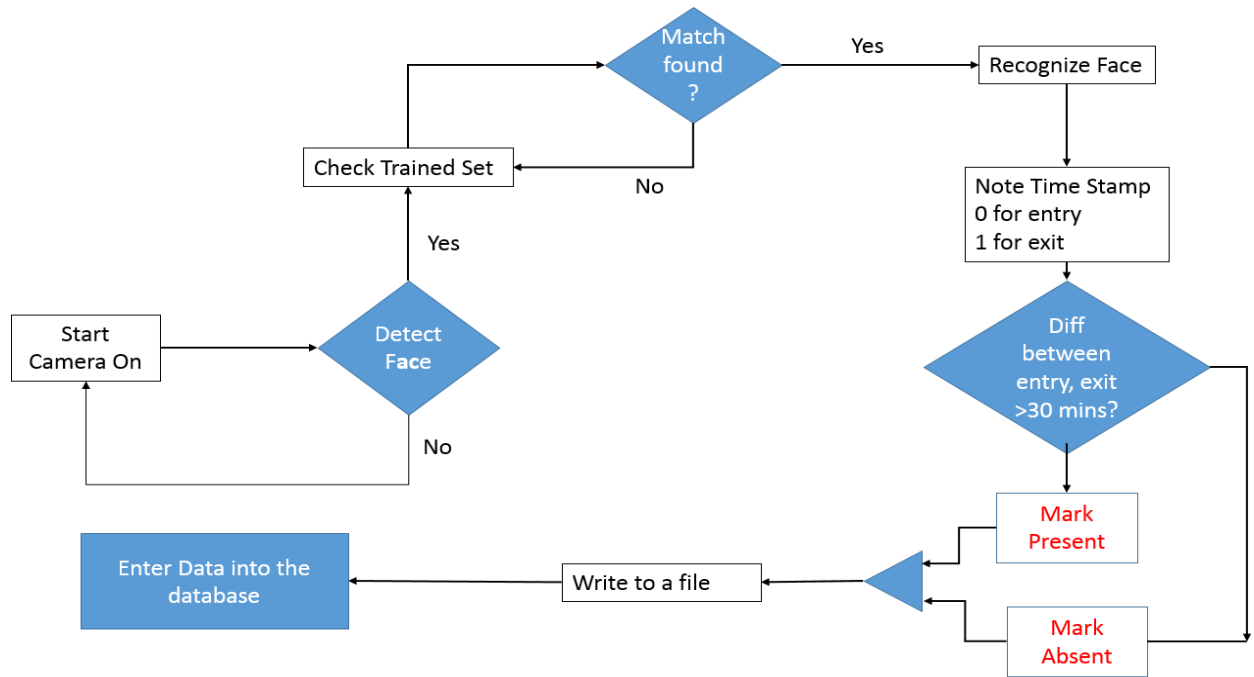
INFO [ NAME {Varchar (10)}, DATE {Date} , START TIME {Time} ,  
DAYS {Varchar(10)}, END TIME {Time} ]



TIME\_TABLE [ DAYS {Varchar(10)}, START TIME{Time}, END TIME {Time},  
PROFESSOR {Varchar(10)}, SUBJECT {Varchar(10)} ]

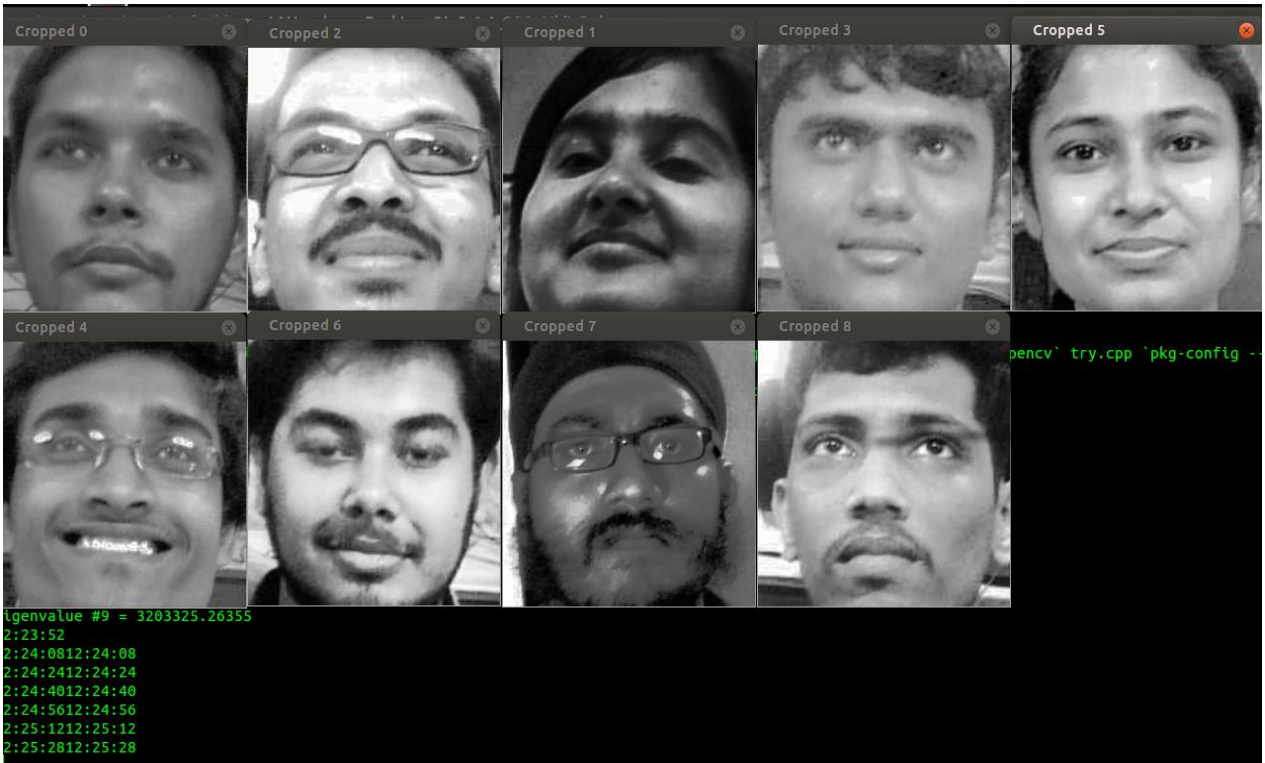


# PROCESS FLOW AND ER MODEL



# SCREEN SHOTS

Training in Progress



# SCREEN SHOTS

Displaying all Queries without any filter

The screenshot shows the 'MainWindow' application. At the top, there is a button labeled 'Update Queries'. Below it, the text 'Count is 15' is displayed next to a 'Go' button. Underneath, there is a 'Run your Query' button followed by a text input field. Below these are several empty input fields for filtering. The main part of the window is a table with 8 columns: 'name', 'dates', 'start\_time', 'end\_time', 'days', 'prof', and 'subject'. The table contains 7 rows of data, numbered 7 to 13 in the first column. A scrollbar is visible on the right side of the table.

	name	dates	start_time	end_time	days	prof	subject
7	Gurvinder	2015-04-29	08:30:00	09:25:00	Wednesday	TA5	SS
8	Gurvinder	2015-04-29	09:25:00	10:20:00	Wednesday	-	Elective
9	Gurvinder	2015-04-29	10:30:00	11:25:00	Wednesday	RGM	DBMS
10	Gurvinder	2015-04-29	11:25:00	12:20:00	Wednesday	TA8	UP
11	Aadesh	2015-04-29	02:00:00	03:50:00	Wednesday	UP Lab	Lab
12	Aadesh	2015-04-29	08:30:00	09:25:00	Wednesday	TA5	SS
13	Aadesh	2015-04-29	09:25:00	10:20:00	Wednesday	-	Elective

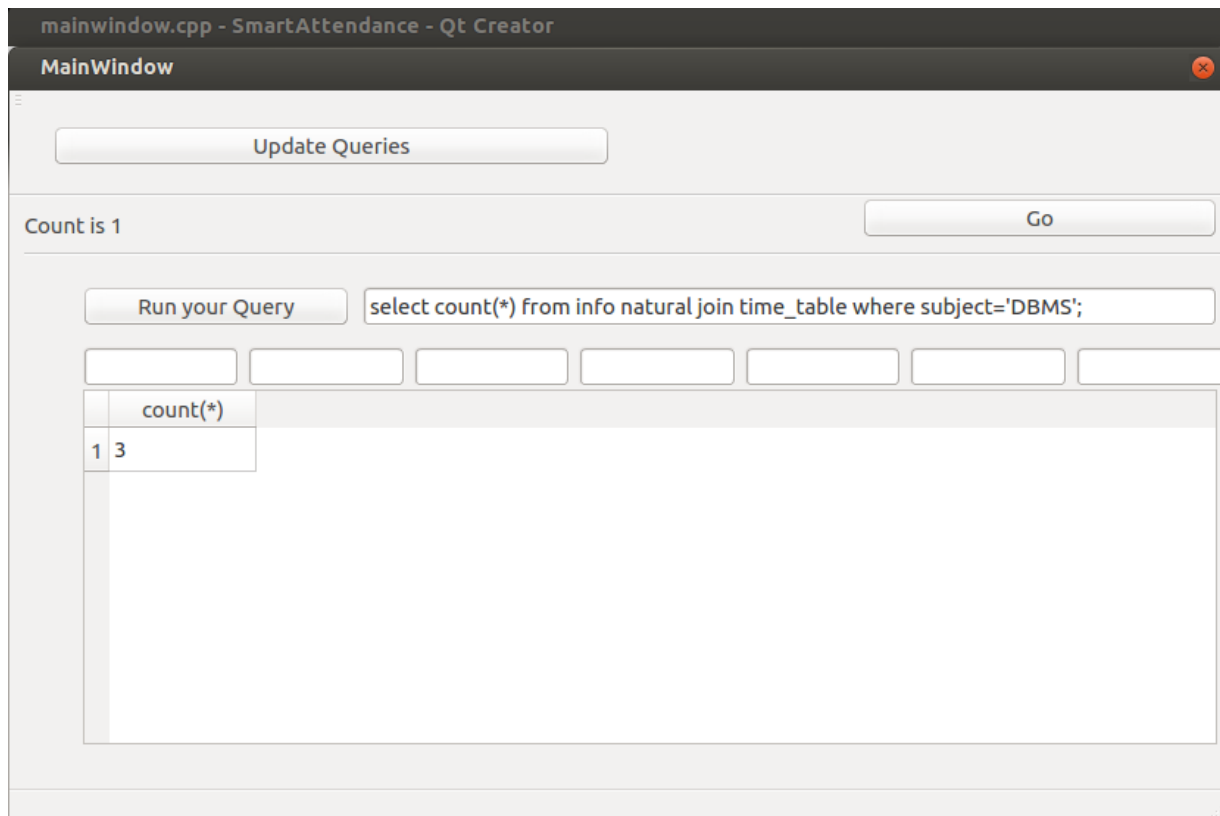
Queries with Filter (Automatic)

The screenshot shows the 'MainWindow' application with the title bar 'mainwindow.cpp - SmartAttendance - Qt Creator'. The 'Update Queries' button is at the top. Below it, the text 'Count is 3' is displayed next to a 'Go' button. Underneath, there is a 'Run your Query' button followed by a text input field. Below these are several empty input fields for filtering, with the last one containing the text 'DBMS'. The main part of the window is a table with 8 columns: 'name', 'dates', 'start\_time', 'end\_time', 'days', 'prof', and 'subject'. The table contains 3 rows of data, numbered 1 to 3 in the first column. A scrollbar is visible on the right side of the table.

	name	dates	start_time	end_time	days	prof	subject
1	Supreksha	2015-04-28	11:25:00	12:20:00	Tuesday	RGM	DBMS
2	Gurvinder	2015-04-29	10:30:00	11:25:00	Wednesday	RGM	DBMS
3	Aadesh	2015-04-29	10:30:00	11:25:00	Wednesday	RGM	DBMS

# SCREEN SHOTS

## Queries with Filter (Manual)



## A Project By:

