

Project 04

Chicago crimes

Group G

Team Members:

Pushparaj P. Parab

Rujuta Mahindrakar

Urmil Maru

Table of Content

TABLE OF CONTENT	2
INTRODUCTION	3
USE CASES	4
ACTORS:	4
ACTOR DESCRIPTIONS	4
PLATFORM	5
ILLUSTRATION OF FUNCTIONALITY	5
OVERALL DESIGN:	6
REQUEST LIFE CYCLE:	6
SUMMARY DISCUSSION	7
CHALLENGING PARTS	7
EASY PARTS	7
DATABASE DESIGN	8
WE HAVE 2 MAJOR TABLES:	8
AND THE REMAINING SUPPORTING TABLES	9
DEMO VIDEO LINK	10

Introduction

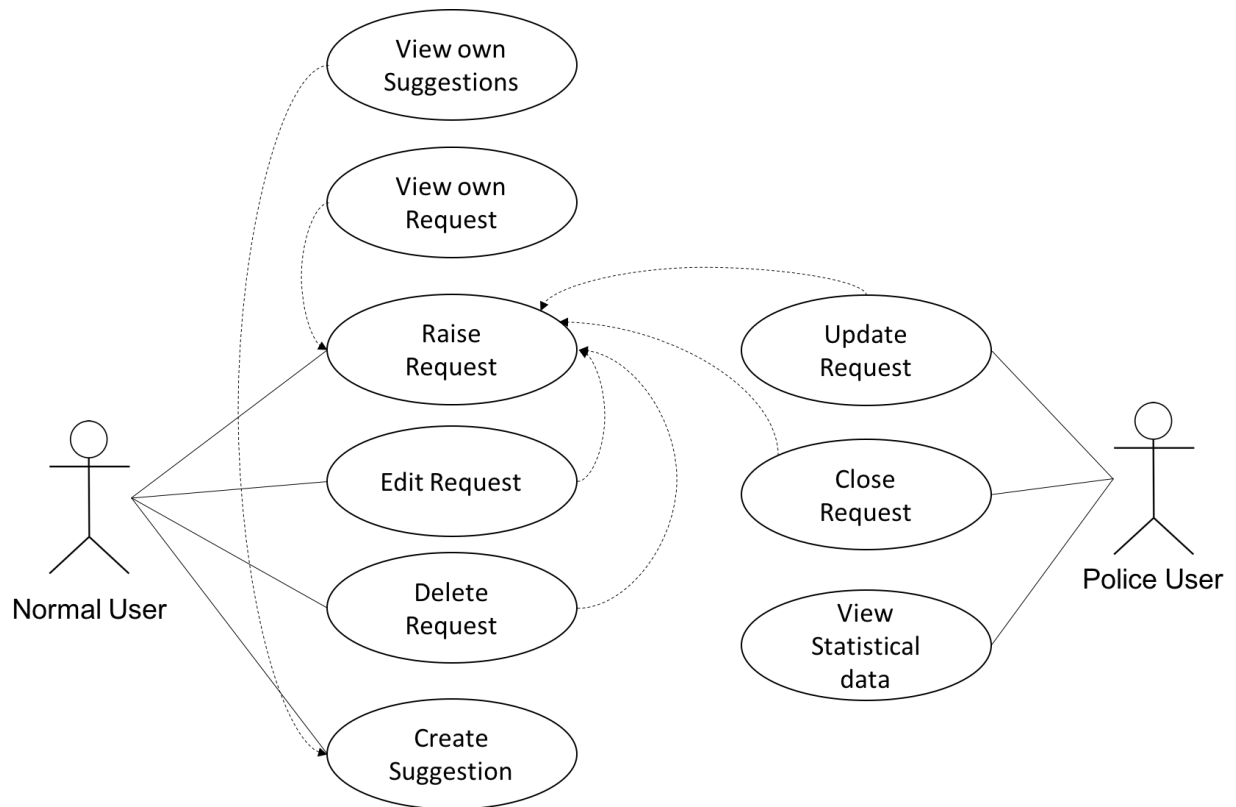
In Today's world, crime is on rise where we are faced with atrocities like theft, murder, homicide etc. Security has become number priority for administration more than ever. Police are bombarded with new challenges very now and then. There is a barrier between police and citizens where reporting crime can be a long process, consuming lot of time. With increase in crime, time has come to re wire our ways to approach it. It will be beneficial for police, if citizens are more connected to police in such a way that they can communicate on the go. The online application solves the problem and provides solution which benefits both police and citizens. This will result in saving time and efforts which would not have been possible by going through the conventional way of reporting a crime through police station.

Online application gives users option of lodging report of crime online where they can describe the nature of crime and location of incident. Following this report, police will able track down criminal events in the city immediately after event has occurred. This increase efficiency of police to handle crime and keep the city is safe. Police will be able to update the status of the report filed by the citizens based on the progress of the case. People can see the status of their report which has been filed online. This will help police to work more productively towards the felony.

Users will have option to post feedback or any suggestions on the application so that police will be able to connect with them making it interactive for both the parties. Police will have option to locate the affected area with the help of google maps feature on the application itself rather than doing in manually. Police will be able to analyze the criminal data from past years based on the criminal activity. Responsive data graphs will enable police to examine trends in crime more accurately.

Overall this application will assist police to work more efficiently towards felonies and bring out successful results in criminal cases.

Use Cases



Actors:

- a) Police
- b) Normal user

Actor descriptions:

Police

- Updates the status of request filed by user.
- Closes the request upon completion of case.
- Checks graphical data of crimes in past years based on criminal activity.

User

- Files a request.
- Gives suggestion.

Platform

- Business logic: PHP,
- Front end: HTML5, CSS3, Bootstrap.
- Scripting language: JavaScript, AJAX, JSON, JQuery
- Database: MySQL
- Additional Libraries: In house graph library, Google map API, geocomplete.js
- Tools used for coding: Sublime text.

Illustration of Functionality

1) Login

- Users and Police must sign into the application to use the application. Similarly, credentials assigned to both users and police are required to log into the application.

2) Files a request (User)

- On account of any crime occurrence users can file a report stating nature of crime, location of scene, name and address.
- Here we have used Google API to get the exact latitude and longitude of the entered address, which will be then stored in the database which can be view later on the map by the police user.

3) Edit the Request:

- The user who submitted the request can edit or delete the submitted request
- **Note:** The deletion can be done only when the created request in Active status and not when It is picked for service or when it is closed.

4) View Own Request:

- The user can view all the request submitted by him.

5) Suggestions/Feedback option

- Users have option of posting suggestions in forum included in the application. Through this police can connect with citizens and have better understanding. Of what they are expecting.

6) View Suggestion submitted:

- Can view the suggestion which are submitted by him.

7) Graphical Data/ statistical data

- Police have feature of viewing criminal data in form of graph such as bar graph and donut charts. It can display data of past years based on occurrence of crime. This will basically display the crime count based on year which can be then drill down to check each year's data base of different type of crimes and to compare how many arrests were made which type of crime is increasing or decreasing.

8) Google Maps

- While entering location for the crime user get options based on google maps suggestions. Police can locate the accurate details of location with help of the feature in the application.

Overall design:

We have 2 major parts in the system

- 1) Role based system
- 2) Statistical data analysis

And when a request is created by any user it will follow a life cycle which is discussed below:

Request Life cycle:

- 1) Active

When the user creates a new request it is by default created in the active state, which means, in this stage the user can edit and delete the request if he wants. And It also means that the Police can pick this request for service.

- 2) In Service

When the police user picks a request to service it the status is changed to in-service. And it cannot be edited or deleted.

- 3) Closed

Finally let's say that the request is complete then the police user will have to enter some more fields, like arrest made etc. And this time the record is entered in the main crime table which is used to the statistics. And the request status is changed to close. Which means this request cannot be edited or deleted but can be just viewed and can be used by the user who entered the request to check the extra information added by the police.

Summary Discussion

Challenging parts

- Time spent in choosing a data set for the project
- Learning and understanding technologies like AJAX, JSON, JQuery
- Had to spend time in understanding the details of database like IUCR code, Nature of crime, term like beat and FBI Code.
- Due to the large size of database, we faced problems to load data from CSV file
- Had to normalize the database and index them because of its size which initially took time to reload in the application.

Easy parts

- Created normalized tables without spending much of the time
- Made rough plan for the application like features and functionality

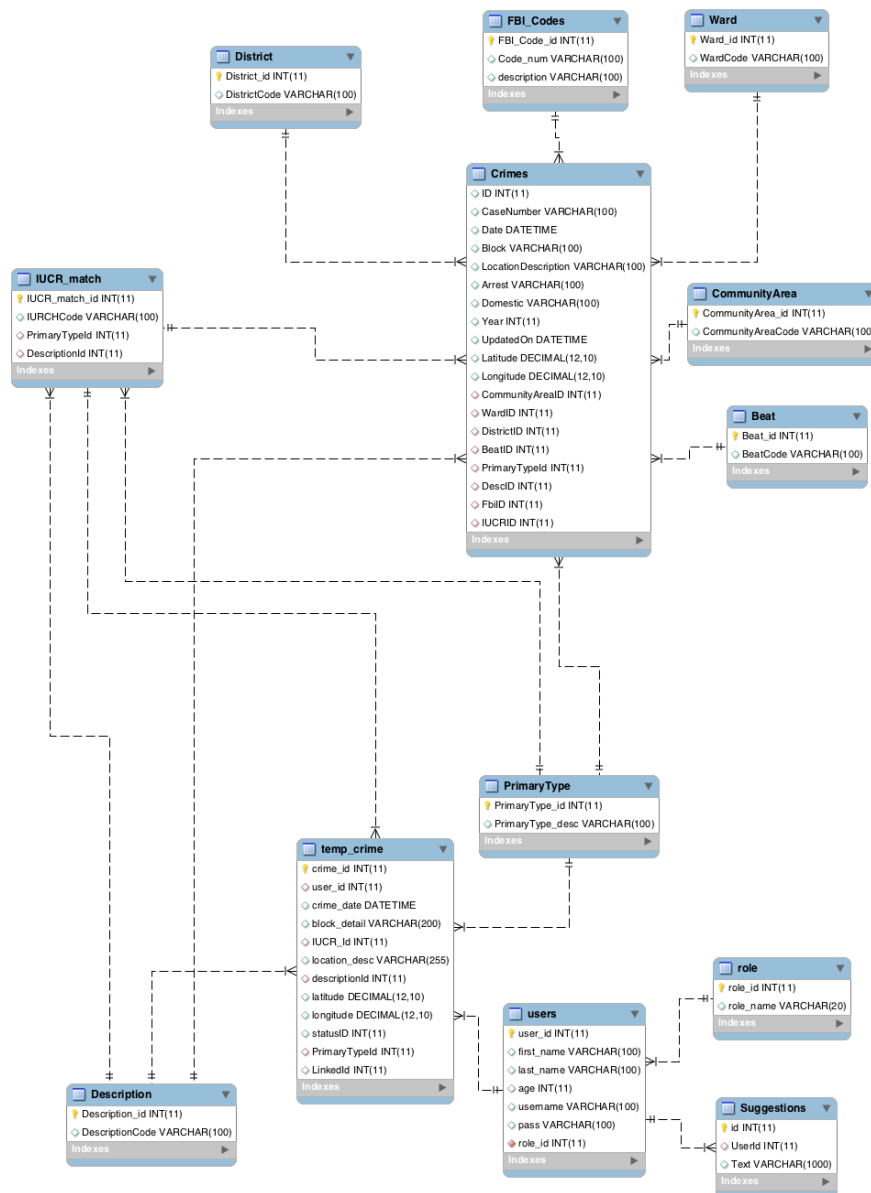
Division of labor

Pushparaj – Used libraries for displaying graph data into the application. Integrated business layer of the application with database and front end. Performed AJAX, and JSON scripting on the application to dynamically generate data from the application.

Rujuta- Designed front end PHP HTML forms for the application and created few tables in database for the application. Assisted Pushparaj in building up the business layer of the application. Involved in Normalization of database.

Urmil- Designed front end PHP HTML forms for the application and created few tables in database for the application initially. Assisted Pushparaj in creating business layer of the application. Involved in normalization of database.

Database Design:



We have 2 major tables:

Crimes table: This table is the main dataset that is considered for the application. This tables will have all the crime data since 2001. And we have used this tables as the final data source where only clean crime data is entered, as we have explained earlier in this document. Normal user will create request which will be first stored in the temp table and then by the police to service and finally close. And while closing we get all the necessary data and store in these table. We have normalized this table.

Temp Crime: Table stores all the data which is entered by the users, as explained above and at the time of closing the request, we change the status and stored the id from the main crime table into this table so that we can link this request to the final data in the crime table.

And the Remaining supporting tables

FBI Codes- indicates the crime classification as outlined in the FBI's national Incident based reporting System(NIBRS).

Rule: FBI code is associated with crime type, so one FBI Code is assigned to particular crime

Ward - Ward is a city council district where incident occurred

District: It indicates police district where incident occurred

IUCR Match: The Illinois uniform crime reporting code. This is linked to primary type table and description. So one IUCR code is assigned to a pair of description and primary type

Rule: We have a dedicated table which stores the information like IUCR_ID, IUCR_CODE, Primary type id and Description id. And based on the user input i.e. based on the Primary type and description we have a unique IUCR code which gets assigned to it.

For example: IUCR Code for Burglary and forcible entry is 0610

Community Area- Indicates community area where incident occurred Chicago currently has 77 such areas

Beat- A beat is smallest police geographic area and each beat has dedicated police beat car. Three to five beats make up a police sector and three sectors make up a police district

Primary Type: It indicates primary type of crime based on IUCR Code

Description- It describes the type of crime in more precise way for example Forcible entry, Armed handgun

***Note:** Reasons for storing all 3 columns in the main crime table: we know that all we need is the IUCR code and based on that we can get the primary type and the description. But unfortunately in the dataset we are not having all the combinations for the primary type and description which gives us the IUCR code.*

Users: It contains citizens who are registered on the application, and each user will be assigned a role.

Role: It shows type of users like whether it is police or citizen

Suggestions: This table stores the suggestions entered by users on the application

Demo video Link

<https://youtu.be/t6gBlj6o2qg>