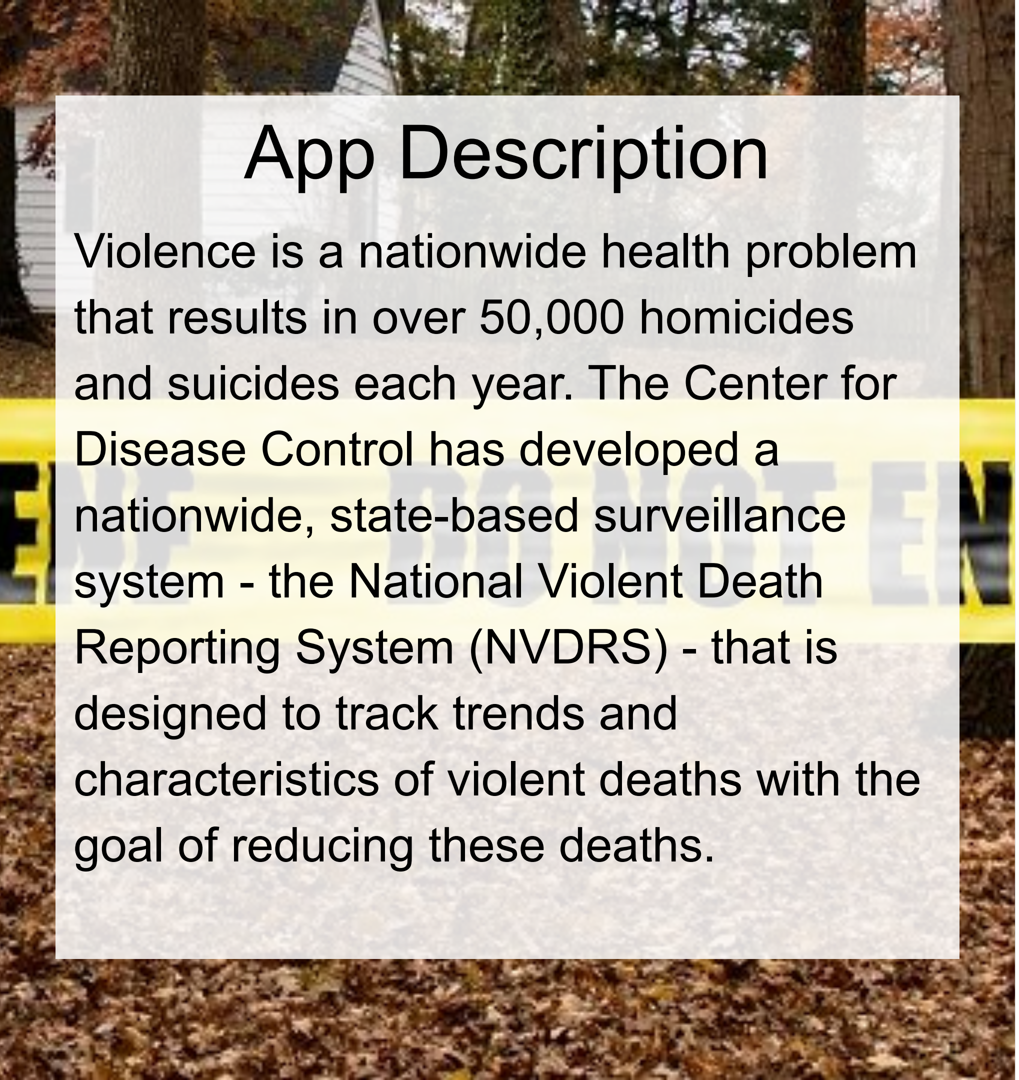
DSI

Death Scene Investigation Application

Team 6: Evan Yang, Eura Shin, Matthew Phelan, Trent Woods

Presentation Overview

- General App Description
- Code QA Review
- System Overview
- Video Demonstration
- Testing
- Story Points/Maintenance
- Lessons Learned



App Description

Violence is a nationwide health problem that results in over 50,000 homicides and suicides each year. The Center for Disease Control has developed a nationwide, state-based surveillance system - the National Violent Death Reporting System (NVDRS) - that is designed to track trends and characteristics of violent deaths with the goal of reducing these deaths.

Coroner Investigation Reporting System (standardized form)

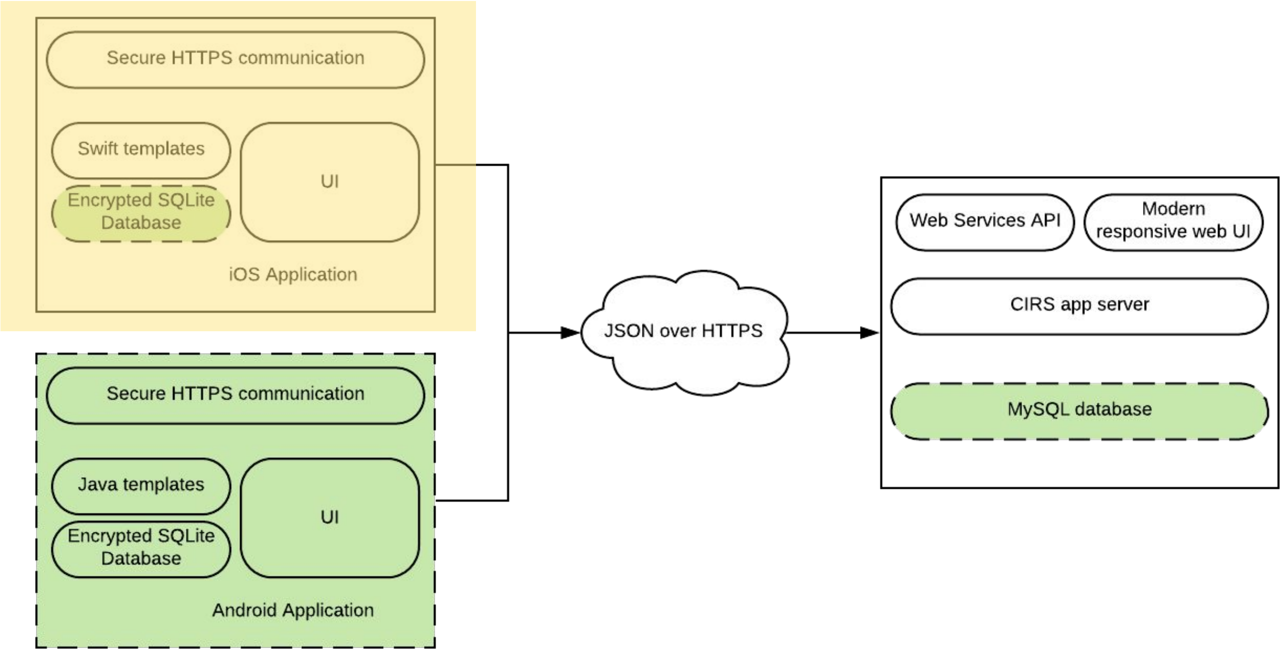
Online web form and database

DSI App for reporting on scene

Code QA Review Results

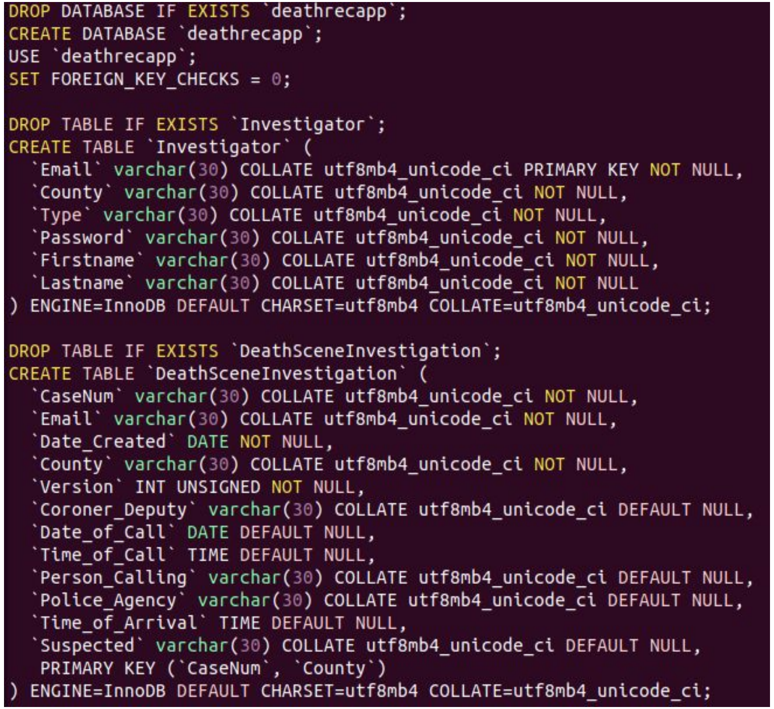
- Constraints not effectively used on the front end - view of the app is dependent upon the device of the user
- HTTP requests were made all at once using one long struct
    - Divided it up depending on form section
    - Allows us to save the form as we go rather than only at the end
- Version control not implemented yet
- View and functionality of app needs to change depending upon clearance level

System Overview



Architecture Diagram depicting a high level structure of our system

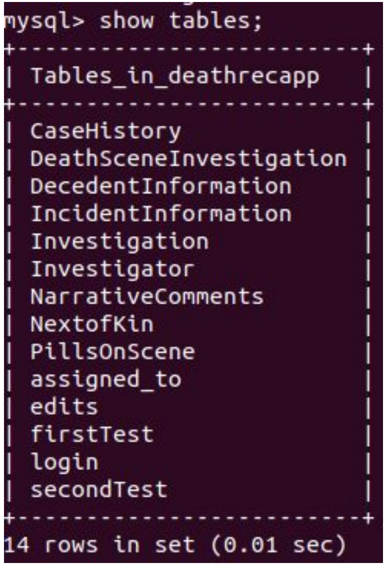
Backend Implementation



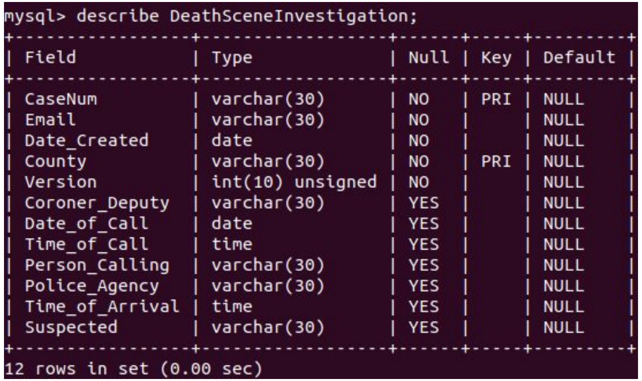
```
DROP DATABASE IF EXISTS `deathrecapp`;
CREATE DATABASE `deathrecapp`;
USE `deathrecapp`;
SET FOREIGN_KEY_CHECKS = 0;

DROP TABLE IF EXISTS `Investigator`;
CREATE TABLE `Investigator` (
  `Email` varchar(30) COLLATE utf8mb4_unicode_ci PRIMARY KEY NOT NULL,
  `County` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `Type` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `Password` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `Firstname` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `Lastname` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;

DROP TABLE IF EXISTS `DeathSceneInvestigation`;
CREATE TABLE `DeathSceneInvestigation` (
  `CaseNum` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `Email` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `Date_Created` DATE NOT NULL,
  `County` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `Version` INT UNSIGNED NOT NULL,
  `Coroner_Deputy` varchar(30) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
  `Date_of_Call` DATE DEFAULT NULL,
  `Time_of_Call` TIME DEFAULT NULL,
  `Person_Calling` varchar(30) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
  `Police_Agency` varchar(30) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
  `Time_of_Arrival` TIME DEFAULT NULL,
  `Suspected` varchar(30) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
  PRIMARY KEY (`CaseNum`, `County`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```



```
mysql> show tables;
+-------------------------+
| Tables_in_deathrecapp   |
+-------------------------+
| CaseHistory             |
| DeathSceneInvestigation |
| DecedentInformation     |
| IncidentInformation     |
| Investigation           |
| Investigator            |
| NarrativeComments       |
| NextofKin               |
| PillsOnScene            |
| assigned_to             |
| edits                   |
| firstTest               |
| login                   |
| secondTest              |
+-------------------------+
14 rows in set (0.01 sec)
```

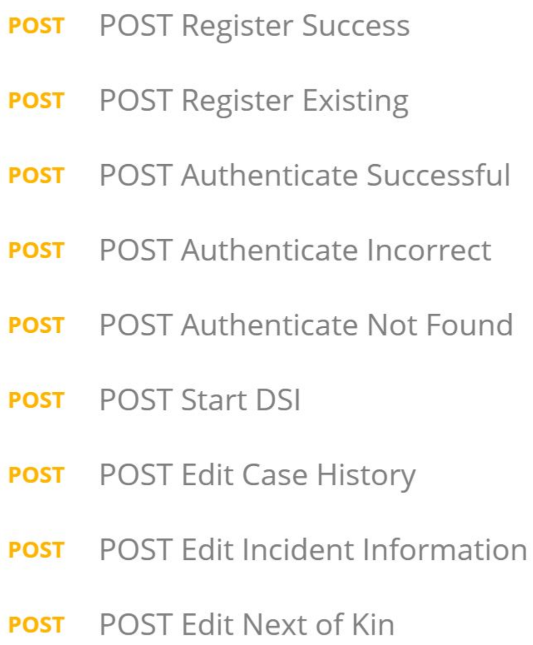


```
mysql> describe DeathSceneInvestigation;
+-----------------+------------------+------+-----+---------+
| Field           | Type             | Null | Key | Default |
+-----------------+------------------+------+-----+---------+
| CaseNum         | varchar(30)      | NO   | PRI | NULL    |
| Email           | varchar(30)      | NO   |     | NULL    |
| Date_Created    | date             | NO   |     | NULL    |
| County          | varchar(30)      | NO   | PRI | NULL    |
| Version         | int(10) unsigned | NO   |     | NULL    |
| Coroner_Deputy  | varchar(30)      | YES  |     | NULL    |
| Date_of_Call    | date             | YES  |     | NULL    |
| Time_of_Call    | time             | YES  |     | NULL    |
| Person_Calling  | varchar(30)      | YES  |     | NULL    |
| Police_Agency   | varchar(30)      | YES  |     | NULL    |
| Time_of_Arrival | time             | YES  |     | NULL    |
| Suspected       | varchar(30)      | YES  |     | NULL    |
+-----------------+------------------+------+-----+---------+
12 rows in set (0.00 sec)
```

Video Demonstration

Testing

- Types:
  - Unit
    - White Box
  - API
  - Acceptance
    - Black Box
    - Beta
- Unit Testing is manual
- Most of the API testing is automated
  - Postman



POST POST Register Success

POST POST Register Existing

POST POST Authenticate Successful

POST POST Authenticate Incorrect

POST POST Authenticate Not Found

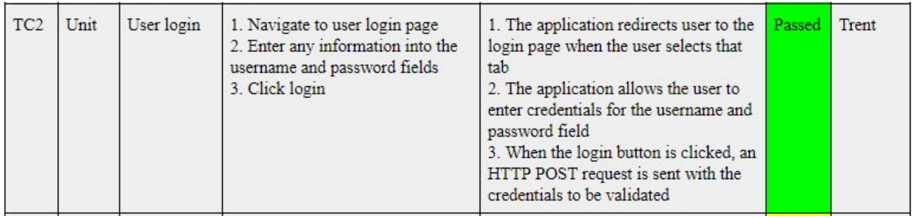
POST POST Start DSI

POST POST Edit Case History

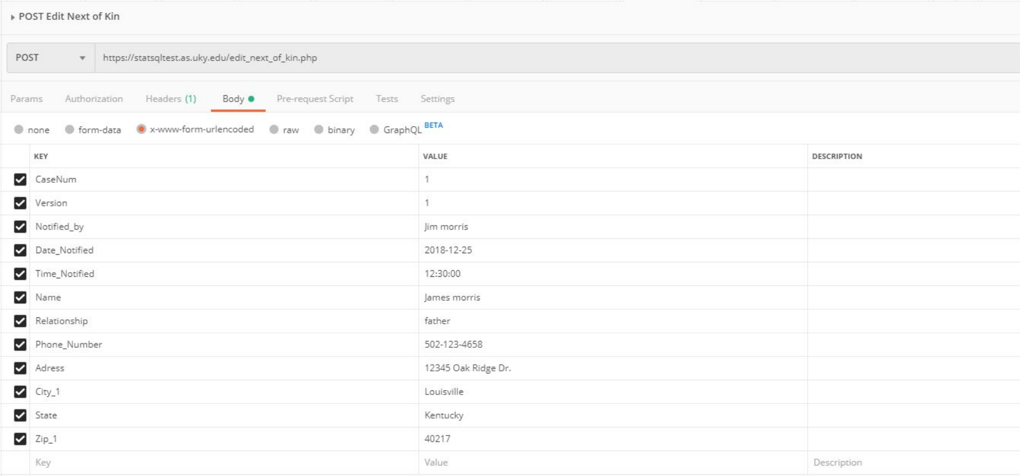
POST POST Edit Incident Information

POST POST Edit Next of Kin

Testing (cont.)



TC2	Unit	User login	1. Navigate to user login page 2. Enter any information into the username and password fields 3. Click login	1. The application redirects user to the login page when the user selects that tab 2. The application allows the user to enter credentials for the username and password field 3. When the login button is clicked, an HTTP POST request is sent with the credentials to be validated	Passed	Trent



▸ POST Edit Next of Kin

POST ▾ https://statsqltest.as.uky.edu/edit_next_of_kin.php

○ none ○ form-data ● x-www-form-urlencoded ○ raw ○ binary ○ GraphQL BETA

	KEY	VALUE	DESCRIPTION
☑	CaseNum	1	
☑	Version	1	
☑	Notified_by	Jim morris	
☑	Date_Notified	2018-12-25	
☑	Time_Notified	12:30:00	
☑	Name	James morris	
☑	Relationship	father	
☑	Phone_Number	502-123-4658	
☑	Adress	12345 Oak Ridge Dr.	
☑	City_1	Louisville	
☑	State	Kentucky	
☑	Zip_1	40217	
	Key	Value	Description

Story Points

User Story	**Story Points**
Work on the investigation form offline and sync when online	2
Autocomplete on the app so that I can quickly fill out the form	3
Three levels of login security clearance	5
Android and iPhone versions	5
Have access to previous versions of the forms	3
Secure connections to the server	6
Easily accessible on devices of different sized	3

⅔ Original Storypoints

Maintenance

- Github repository has a “backend,” “iOS,” and “Android” folder
- .sql dump file to edit the structure of the backend
- Current testing database is hosted at “linkblueid@statsqltest.as.uky.edu”
- Refer to User Manual and Administrator’s manual

Lessons Learned/Best Practices

- Incremental Development is key
- Have realistic timelines
- Establish an easy way for the team to communicate (Slack)
- Commit and push changes as soon as they’re ready
    - Ensures all team members are working with the same version of the app and know about recent changes

Summary

- General App Description
- Code QA Review
- System Overview
- Video Demonstration
- Testing
- Story Points/Maintenance
- Lessons Learned

Questions

¿ ?