DSI

Software Requirements Specification
for
Death Scene Investigation

Client: Dr. Sabrina Brown
Team: CS 499-001 Team 6 Fall 2019
Date Submitted: September 27th, 2019

Individual Contributions from Team 6 Members:

Authors	Content Contributed	WC
Evan Yang	Introduction, Project Overview (Sections 2.2-2.4), Development and Target Environments, System Model (Section 4.1), User Interaction (Section 5.1)	1936
Eura Shin	Introduction, Project Overview (Section 2.1), User Interaction (Section 5.2), System Model, Feasibility (Section 8.1)	1667
Matthew Phelan	Title Page, Table of Contents, Functional Requirements, Nonfunctional Requirements (Sections 7.1, 7.2.1, 7.2.3.3, 7.2.5.1), Conclusion	1525
Trent Woods	Nonfunctional Requirements (Sections 7.2.2, 7.2.3.1, 7.2.3.2, 7.2.4), Feasibility (Sections 8.2-8.4)	917

Table of Contents

1. Introduction **3**
2. Project Overview **4**
2.1 Introduction 4
2.1 System End Users and Stakeholders 4
2.2 Relevant Background 4
2.2.1 Terms and Definitions 4
2.2.2 Death Scene Investigation Process 5
2.2.3 Existing Frameworks 5
2.3 Main Features 6
2.4 Design Constraints 6
3. Development and Target Environments **8**
3.1 Introduction 8
3.2 Hardware Components 8
3.3 Software Components 8
4. System Model **9**
4.1 Introduction 9
4.2 Components of Existing vs. Proposed System 9
4.3 Text Descriptions of Diagrams 9
4.3.1 Entity Relationship Diagram (ERD) 9
4.3.2 Use Case Diagram 10
5. User Interaction **11**
5.1 Introduction 11
5.2 Use Case Scenarios 11
6. Functional Requirements **15**
6.1 Introduction 15
6.2 Functional Requirements List 15
6.2.1 Security 15
6.2.2 Functionality and Usability 16
7. Nonfunctional Requirements **18**
7.1 Introduction 18
7.2 Nonfunctional Requirements List 18
7.2.1 Security 18
7.2.2 Compatibility 18
7.2.3 Performance 18
7.2.4 Maintainability 19
7.2.5 Useability 19
8. Feasibility **20**
8.1 Introduction 20
8.2 Bare Bones Version 20
8.3 Enhanced Version 20
8.4 Team Member Capabilities 21
9. Conclusion **22**
10. Appendix **23**
11. References **25**

1. Introduction

Violence is a nationwide health problem that results in over 50,000 homicides and suicides each year. To obtain a better understanding of these violent deaths, the Centers 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. (Brown, 1)

Death scene investigation is an extensive process that requires collaboration between the coroner/deputy, the victim's associates, medical examiners, and law enforcement officers over long periods of time. It is critical that this Coroner/Medical Examiner (CME) data is collected to preserve accuracy and maintain an incredible level of detail that meet the standards of statutory jurisdiction (KRS 72.405). In the Commonwealth of Kentucky there are 120 counties, each with its own elected coroner and fluctuating number of deputy coroners. In addition, there is variation amongst the resources and methods that each county employs to implement the CME collection process, making it difficult - and sometimes impossible - to follow the process behind a ruling on the cause of death.

The Kentucky Coroner Association (KCA) Board of Directors, county coroners, and the State Medical Examiner (SME) director have worked together to massively transform the investigation process for violent deaths in Kentucky. Dr. Sabrina Brown, a professor of Epidemiology from the University of Kentucky's College of Public Health, has spearheaded efforts in developing the Coroner Investigation Reporting System (CIRS), which outlines a standard coroner reporting form for all violent death investigations and hosting this information online. However, the reporting form is currently collected on paper and manually entered into the online database, which makes the system vulnerable to insecure or misentered data.

This design project will contribute to this effort through an iOS mobile application for death scene investigators to report observations and characteristics of a death scene as well as an Android version of the same application. This cross-platform mobile application will enable coroners to work with the CIRS at the physical investigation scene and facilitate organization between all investigative parties. It will offer significant improvements over the old web based reporting system (CRI) by providing a responsive UI for the investigators, improving mobility of the application by caching forms that cannot be sent immediately to a database due to a lack of wireless connection and allowing investigators to generate real-time data and metrics about the death scene with on-device sensors. (Brown, 2-3)

2. Project Overview

2.1 Introduction

In this section we introduce the mobile Death Scene Investigation (DSI) application, which will streamline the way in which investigations collect, store, and interpret death scene related data. In the following paragraphs, we establish the relevant parties and expectations for the applications. We provide a brief overview of the current information collection strategies on site and the extent of the existing software which facilitates this process. In addition to overviewing our product's key features, we explain the specific design limitations which led to certain design decisions that we have made for the DSI mobile application when planning this project.

2.1 System End Users and Stakeholders

Dr. Sabrina Brown is the client for the DSI application project. The system is intended for use with the CIRS, and therefore, the end users for the mobile application are coroners and medical examiners responsible for providing a ruling on the cause of death for the investigation. Occasionally, the KYVDRS staff will need to access the data submitted to medical examiners through the web database system.

2.2 Relevant Background

2.2.1 Terms and Definitions

Terms	Definition
Center for Disease Control (CDC)	The leading public health agency in the United States that is dedicated to saving lives and protecting the health of Americans.
National Violent Death Reporting System (NVDRS)	A nationwide state-based surveillance system designed to track trends and characteristics of violent deaths with the goal of reducing violent deaths.
Kentucky Violent Death Reporting System (KYVDRS)	The Kentucky subdivision of the National Violent Death Reporting System.
Medical Examiner (ME)	The death scene investigator responsible for performing post-mortem autopsies and completing the forensic pathology sections of the form.
Deputy/Coroner	The death scene investigator and government official who possesses the jurisdiction to conduct or order an

	inquiry into the manner or cause of death and to investigate or confirm the identity of an unknown individual who has been found dead.
Kentucky Coroner's Association (KCA)	An organization that supports the coroners of Kentucky through education, training, and providing resources to assist in death scene investigations.
Coroner Investigation Reporting System (CIRS)	The death investigation reporting system currently active in Kentucky. Involves form data entry on pen and paper. This information is transferred to a web system.

2.2.2 Death Scene Investigation Process

In this section we provide a general overview of the CME data collection process, highlighting the evolution of the data as it is passed between investigative parties. Currently, the CME data collection process involves several stages. First, the coroner or deputy in the county of a particular decedent conducts an investigation into the scene of the death of the decedent, which includes interviewing survivors, gathering the medical history and precipitating circumstances of the decedent, and involving other investigating agencies if applicable to the case. The MEs from a nearby regional office receives documentation of the death scene from the investigator and utilizes them to corroborate their findings during autopsy, from which they can assist the coroner and law enforcement agency in determining the cause of death that should be recorded on the death certificate for a particular case. The coroner will also consider the results of the investigation by law enforcement agency in order to determine the cause of death. (Brown, 1) A graphical overview of the death scene investigation process is given in Figure 2 of the Appendix.

2.2.3 Existing Frameworks

The existing framework for death scene investigations in the State of Kentucky consists of the CIRS along with a standardized reporting form on paper. The KYVDRS supplies each county with a notebook of the standardized forms as well as free access to the CIRS website. Coroners can submit the information collected on the CIRS paper forms from their cases in several ways, including via mail, faxing, email, or directly to the CIRS website. (Brown, 2) Recently, Dr. Brown has proposed a transition from this older framework to a newer one that utilizes smartphones and tablets at the scene of a death investigation to collect data from the field and send the data to a central webservice similar to the CIRS. This new proposed framework involves a mobile application that runs natively on the iOS and Android platforms and performs a variety of functions that will serve to replace the older framework, including authenticating users of the DSI mobile application, capturing data from the field to create a DSI report, submitting the reports to a centralized web service, and searching and reviewing previously submitted forms.

This new proposed framework has been partially implemented by a previous CSS 499 Senior design team, which includes the iOS application along with a SQL database schema. The iOS application in its current state, while usable, requires further work in order to be fully functional with all of the proposed features as outlined by Dr. Brown. Similarly, the SQL database along with the backend REST APIs that were implemented by the previous team are currently not usable. The tables containing the entities and their attributes were included within the MySQL database but the relations between the entities were nonexistent and none of the entities specified any primary keys. It was also discovered that the REST APIs that were implemented by the previous team were hardcoded to insert the same data to the database when called. Our proposed project will build upon the work accomplished by the previous CS 499 senior design team by finishing implementation of both the backend and frontend of the mobile application such that the application is fully functional. In addition to this, we will be adding new functionality to the application, such as data encryption, version control of the death scene investigation forms, and editing privileges for the form, among other features listed under our functional requirements in Section 6 of this document. We believe that our proposed project will provide a solution that fully accounts for every feature and functional requirements of the original project proposal as specified by Dr. Brown. We also believe that our solution should be developed rather than bought since there are currently no solutions in the market for this particular problem and the fact that there is an already existing and readily accessible solution that can be improved upon.

2.3 Main Features

The main features for our proposed project have been identified in the proposal submitted by Dr. Brown. The proposed mobile application will be native to the iOS and Android platforms and should run identically on the vast majority of smartphones or tablet devices currently in use within the United States. The application should be able to perform several key functionalities, among them are authenticating mobile users to access the application, collecting data from the field to create a DSI report, searching and reviewing reports available to the user in the DSI system, providing a responsive user interface for the investigators, improving mobility of the application by caching death scene investigation form data on the device, and allowing investigators to generate real-time data and metrics about the death scene with on-device sensors. (Brown, 2-3) While the user interface of the mobile application has been completed by the previous team - aside from several aesthetic-oriented tasks - the rest of the features listed have yet to be implemented. In addition to these features specified from the proposal submitted by Dr. Brown, our team has identified several other features that should be implemented after meeting with Dr. Brown and several other consultants on the project. These features can be found in the functional requirements section of this document (Section 6).

2.4 Design Constraints

The most significant constraints that may influence design decisions for our project are system security and compatibility. System security constraints - such as only allowing authorized users to access death scene investigation data and restricting editing privileges to death scene

investigators - have directed us towards designing a new authorization scheme. This involves both changing and removing some of the features already implemented for authentication on the frontend, which involve removing sections of the frontend code that had usernames and passwords embedded within it and creating a SQLlite database to store usernames and passwords, as well as implementing the editing privileges feature on both the frontend and the backend. In terms of compatibility, our system is constrained in two ways. First, we must ensure that the mobile application that we are building is the same aesthetically and functionally across the two different platforms that we are developing for - iOS and Android. Second, we need to guarantee that when users retrieve a death scene investigation form from the backend the form will appear and behave identically to the same form retrieved on a different device. We have identified two other constraints that are not as influential in terms of design decisions - performance and maintainability. These constraints are detailed in the Nonfunctional Requirements section of this document (Section 7).

3. Development and Target Environments

3.1 Introduction

It is imperative that the application that we are developing is easy and intuitive to use for our clients in order to reduce the burden of collecting data in the field. Since the vast majority of death scene investigators in the state of Kentucky are supplied with a work smartphone, it is reasonable for us to develop our application on the two most popular mobile platforms - iOS and Android. The development and target environments for our proposed project will now be discussed. Specifically, we will identify both the hardware and software components that will be necessary to implement our system.

3.2 Hardware Components

The hardware components that will be necessary to build and run our system include mobile devices (smartphones and tablets) that can run both iOS and Android as well as a Linux based server to host our backend components. The mobile devices will be used to run the mobile application for use by the death scene investigators. Given data sent from the mobile devices via HTTPS requests, the backend Linux server will host the REST APIs that will read from and write to the MySQL database.

3.3 Software Components

For the development of the iOS version of the mobile application, we will be using Xcode as our primary IDE. The various frameworks that we have utilized thus far were imported using CocoaPods. On the other hand, we will be using Android Studio for the development of the Android version of the mobile application. On the backend side of this project, we will be using MySQL to store data from the submitted death scene investigation forms. Our REST APIs will consist of PHP scripts that will run on our server. These APIs will communicate with the front end via HTTPS requests and interact with the MySQL database via queries. The entire project will be subject to software configuration management. For this purpose, we will be using git for version control and GitHub for hosting a remote repository of our project’s codebase.

4. System Model

4.1 Introduction

The overall design of our system architecture is based on the client-server model. It includes a mobile application containing the frontend user interface as well as an encrypted local SQLLite database that can run on both iOS and Android along with a server that hosts the REST APIs that will receive and process data from a death scene investigation form and store it into a MySQL database. In this section, we will present a high level view of our system architecture to show its existing and proposed components and describe their relationships with one another.

4.2 Components of Existing vs. Proposed System

In the graphic below, we provide a high-level view of the current state of the system and features this project will add.

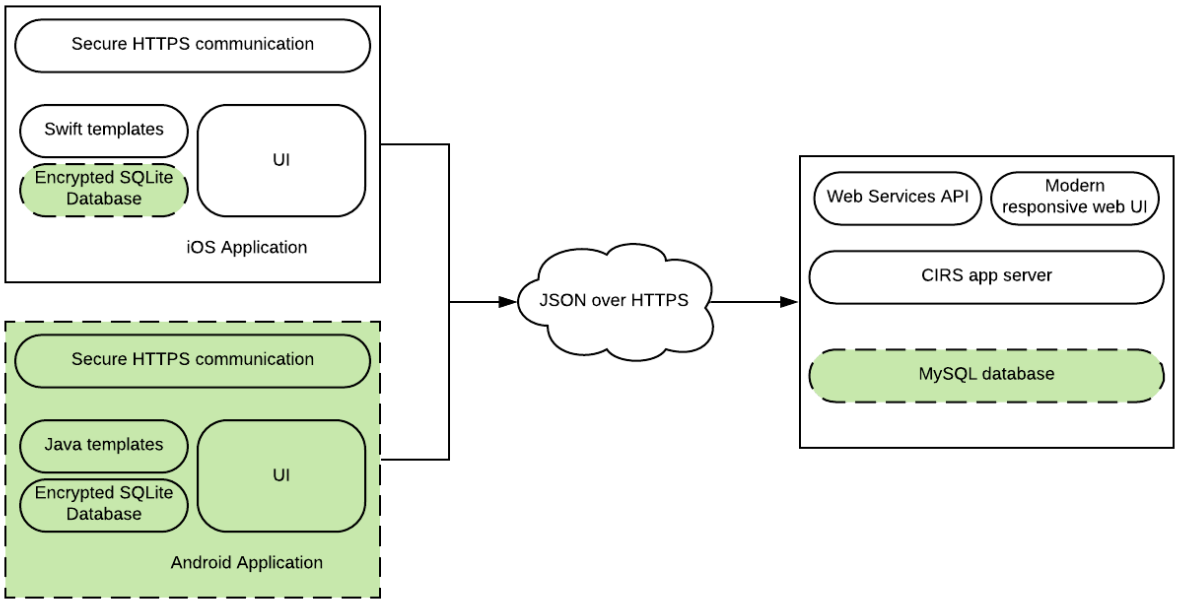

Figure 1. System overview. The features our group intends to add are shaded and dotted, while the existing framework is solid.

4.3 Text Descriptions of Diagrams

4.3.1 Entity Relationship Diagram (ERD)

One of the first orders of business for this project team is to recreate the existing MySQL database. The ER-Diagram in Figure 2 does not include all of the attributes placed in the actual

database, because the DSI form itself is very large; we include only the attributes that are essential to an understanding of the backend structure.

Because the application is based on a form, the representation of this information in a database is straight forward (many attributes to a single entity). However, we chose to compartmentalize the form components based on section. Therefore, each DeathSceneInvestigation entity will *contain* various sections, such as PillsOnScene, DecedentInformation, etc. Because these sections are dependent on the existence of the DeathSceneInvestigation, they are considered weak entities. Additionally, we represent different users of the application as Investigators, who will have standard login data. Finally, the Clearance entity stores levels of clearance possible for each type of investigator.

4.3.2 Use Case Diagram

The Use Case Diagram in Figure 3 focuses on the user experience for three primary user types: coroners, medical examiners, and KYVDRS employees. Each of these users have different clearances in the investigative process, and as a result, will have vastly different experiences interacting with the application. As described in Section 2.2.2 (Death Scene Investigation Process), coroners have the ability to start and edit new investigations. They will be the users that spend the most time with our application. The ME is responsible for filling in the autopsy information on an existent investigation. In some cases, the KYVDRS employee works as a “middleman” to create an application for which the ME can enter autopsy information in cases where they do not have immediate access to the existing form information.

5. User Interaction

5.1 Introduction

The design of the user interface of the mobile application is heavily dependent on the vast possibilities of different use case scenarios. In order to design a robust user experience for the death scene investigators using the DSI application, we must consider and document the various use case scenarios that may arise. We document each use case scenario along with its primary actors, the main success scenario, and its failure extensions.

5.2 Use Case Scenarios

Death Scene Investigator Wishes to Create a New Account	
Actors	Coroner, ME, or KYVDRS Employee
Main Success Scenario	1. DSI clicks the “create new account” button. 2. The system reroutes the user to a registration form. 3. The agent fills his/her information and submits. 4. The backend verifies that the username is unique. 5. The backend verifies that this is an authorized user. 6. The backend assigns a security clearance to the user. 7. The user is logged in and rerouted to the home page.
Failure Extensions	3a. The system does not have connectivity. 1. The system with display a “no connectivity!” message and stay on the filled form page. 3a1. The DSI resubmits or backs out. 4a. The username is not unique. 1. The system displays a “this username is taken” message. 3a1. The user chooses a unique username. 4b. The passwords do not match. 1. The system prompts the user to reenter the passwords. 5a. This is not an authorized user. 1. The system displays a “user not authorized” message.

Death Scene Investigator Wishes to Login	
Actors	Coroner, ME, or KYVDRS Employee
Main Success Scenario	1. DSI submits username and password. 2. The backend checks the credentials. 3. The user is logged in and rerouted to the home page.

Failure Extensions	2a. The password is incorrect for the given username. 1. The system prompts user to re-enter the password. 2a1. The password is incorrect. 2. The system redirects the user to a “forgot password” page. 2b. The username does not exist. 1. The system prompts the user to re-enter the username. 2b1. The username does not exist. 2. The system reroutes the user to the registration page.

Coroner Wishes to Create and Fill a New Death Scene Investigation Form	
Actors	Coroner
Main Success Scenario	1. Coroner clicks the “create new investigation” button. 2. The system verifies that the coroner is authorized to start an investigation. 3. The system creates a new DSI entity in the database. 4. The system assigns the creator to the DSI.
Failure Extensions	1a. Coroner does not have clearance to start a new investigation. 1.The screen displays a message denying access. 1a1. Coroner backs out or tries again.

Coroner/Medical Examiner Wishes to Edit an Existing Death Scene Investigation Form	
Actors	Coroner or Medical Examiner
Main Success Scenario	1. The system fetches the cases to which the coroner is assigned. 2. The application displays the cases. 3. The coroner chooses a case and edits relevant fields. 4. The system saves this new case version in the backend. 5. The page exits to the home screen.
Failure Extensions	1a. The user is not assigned to any cases. 1. The system does not display any cases loaded on screen. 3a. The user changes no fields. 1. The system returns to the home screen without making changes. 4a. Backend is not updated due to connectivity issues. 1. The system will exit to the home screen and locally store edits. 4a1. The system connects to the internet. 2. The system saves the local edits in the backend.

Coroner Wishes to Restore a Previous Version of a Death Scene Investigation Form	
Actors	Coroner

Main Success Scenario	1. The coroner chooses the case to restore. 2. The page displays the previous versions. 3. The user selects a version to restore. 4. The application verifies the user's decision to restore. 5. The backend updates to the previous version and deletes the upstream versions.
Failure Extensions	2a. The case does not have any previous versions. 1. The system displays no previous versions. 4a. The user chooses not to revert the case. 1. The system exits and displays the current case

Medical Examiner Wishes to Fill Out the Autopsy Related Fields	
Actors	ME
Main Success Scenario	1. The medical examiner enters a case number. 2. The backend verifies the case number. 3. The ME fills out the autopsy data and saves. 4. The backend updates the form with autopsy data.
Failure Extensions	2a. An investigation does not exist for the case number. 1. The system displays a "case not found" error

Coroner Wishes to Review and Submit the Final Form	
Actors	Coroner
Main Success Scenario	1. The coroner chooses to submit the case. 2. The application prompts a final submission verification. 3. The submission is updated in the backend.
Failure Extensions	1a. Fields are missing in the final form. 1. The system highlights the missing fields and asks for verification. 2a. The system receives verification of empty form. 2. The system finalizes the form.

KYVDRS Employee Wishes to Abstract the Form Data for the Medical Examiner	
Actors	KYVDRS Employee
Main Success Scenario	1. KYVDRS clicks the "create new investigation" button. 2. The system verifies that the KYVDRS is authorized to start an investigation. 3. The system creates a new DSI entity in the database.

	4. The system assigns the creator to the DSI.
Failure Extensions	1a. KYVDRS does not have clearance to start a new investigation. 1.The screen displays a message denying access. 1a1. KYVDRS backs out or tries again.

6. Functional Requirements

6.1 Introduction

Our product deals with government investigations of deaths. As such, many stipulations are inherent to these investigations, and, naturally, many functional requirements need to be fulfilled to meet the needs of our customer. Our first priority is security, as sensitive government data will be passed through our application. The functional requirements related to security deal with user authentication, security clearances, and read-only constraints for individuals who are not legally allowed to edit death scene investigation data (district attorneys, for example). Confirmation of correctness with our customer and extensive testing will be required for all security related functional requirements to ensure they comply with all legal guidelines on the security of death scene investigation data.

Our next priority involves features associated with application usability and functionality. We are developing this application to provide a more convenient and quicker way to report death scene investigation data in lieu of the traditional, slower death scene investigation paper form. Thus, functional requirements like locally saving data when the device is offline and uploading data from the form to the database when it is back online or form version control are critical to making our application a tool investigators want to switch to. There has to be an incentive to switch from paper forms for our application to be successful, and useability and convenience will provide that incentive.

6.2 Functional Requirements List

Functional requirements appear in order of highest priority to lowest. Note that identifiers for requirements are based on their position in the document. For example, the requirement below is located in section 6 (functional requirements), subsection 2 (functional requirements list), subsection 1 (security), and is the first requirement in the list. This allows us to trace requirements more easily.

6.2.1 Security

6.2.1.1 Editing Permissions: The system shall prevent editing from individuals who do not have editing permissions.
Rationale: Dr. Brown informed us that, legally, if the death scene investigation data needs to be reviewed for a court case, the reviewer can only view the document. Thus, read only constraints are required to prevent this from occurring.

6.2.1.2 Security Clearance: The system shall reject or accept users at login based on their clearance to view data in the form they are trying to access.
Rationale: We learned in our meeting with Dr. Brown that there are three levels of security clearance for viewing death scene investigation data; they are: coroners have universal access,

access for deputies varies from county to county, and district attorneys have limited access (when reviewal is necessary).

6.2.1.3 Data Encryption: The system shall encrypt all communication and connection between the application and the database so all transmitted data is secure.
Rationale: Death scene data is sensitive by nature and contains HIPAA Privacy Rule protected information. Encryption is a must to protect this data from unauthorized individuals.

6.2.2 Functionality and Usability

6.2.2.1 Application Structure: The system shall be structured (data input methods and flow of input) to reflect the death scene investigation form provided by Dr. Brown.
Rationale: It is easier for coroners to switch from the paper form to the electronic version if both methods are similar in structure. Also, the form is provided by the government and data should be stored in accordance with form format. The electronic form is just an extension of this, so it should reflect that.

6.2.2.2 Local Storage and Backup: The system shall store input data locally and backup to the database once internet connection is reestablished.
Rationale: Most death scene investigators will be using tablets without cellular function to input data. Many death scenes will not have WiFi available, so it is important to store their input data locally then sync to the backend once the device is connected to WiFi.

6.2.2.3 Version Control: The system shall save each version of the form and allow access to these versions at any time.
Rationale: If any mistakes are made in the revision or collection of forms, previous versions of the form should be available for viewing to correct the mistake.

6.2.2.4 Navigability Between Form Sections: The system shall include a “table of contents” which allows users to jump between sections of the form.
Rationale: The DSI form is extensive and will require a long period of time to complete. Users should be able to quickly access the part of the form they are interested in filling out, so they don’t have to scroll through the form linearly every time they wish to edit.

6.2.2.5 Autocomplete Feature: The system should offer autocomplete suggestions as users fill out fields of the form.
Rationale: Some of the field elements are very long and wordy. Because these field elements will likely be an item from a consistent list, users will save time and minimize typos using an autocomplete function.

6.2.2.6 Death Scene Photograph Support: The system, via the device’s camera functionality, should allow the taking and storage of death scene photos so they can be attached to the form.
Rationale: Photographs play an important role in documenting the scene of a death, and the native camera functionality of each death scene investigator’s device would make this a convenient addition to the application.

Traceability Matrix For the Functional and Nonfunctional Requirements

Functional Requirements		Nonfunctional Requirements			Test Case ID
FR ID	FR Description	NFR ID	NFR Description	Priority	We have not begun testing
6.2.1.1	**Editing Permissions**	**7.2.1.2**	**Court Case Editing**	HIGH	
6.2.1.2	**Security Clearance**	**7.2.1.1**	**Access Restrictions**	HIGH	
6.2.2.1	**Application Structure**	**7.2.2.2**	**Consistent UI Across Mobile Platforms**	HIGH	
6.2.2.2	**Local Storage and Backup**	**7.2.3.3**	**Local Storage**	HIGH	
6.2.2.3	**Version Control**	**7.2.4.1**	**Detailed Change Documentation**	HIGH	
6.2.2.4	**Navigability Between Form Sections**	**7.2.3.2**	**Quick Navigation Within the Application**	MEDIUM	
6.2.2.6	**Death Scene Photograph Support**	**7.2.5.1**	**Investigator Photographs**	LOW	

Figure 2. This traceability matrix serves to connect our requirements and categorize them based on priority. Once we begin testing, we can connect each requirement to a test.

7. Nonfunctional Requirements

7.1 Introduction

In this section we will be outlining the constraints under which our applications will operate. The nonfunctional requirements we will be outlining include security, compatibility, performance, and maintainability and will be describing the characteristics of our product. Similar to the functional requirements, each non-functional requirement will have an identifier based on its section, subsection, and list position. This will enhance traceability between requirements.

7.2 Nonfunctional Requirements List

7.2.1 Security

7.2.1.1 Access Restrictions: Only authorized users should be able to access the death scene investigation data.
Rationale: Sensitive government data should only be available to those who have authorization to access.

7.2.1.2 Court Case Editing: Individuals reviewing DSI forms for legal proceedings should only be allowed to read the form.
Rationale: When death scene data is complete and being reviewed for legal proceedings it cannot be edited.

7.2.2 Compatibility

7.2.2.1 Cross-platform Collaboration: Different users should have editing access to the same form across all mobile platforms.
Rationale: Multiple parties may be working in tandem on the same case investigation. They should be able to contribute evidence to the same form regardless of mobile platform.

7.2.2.2 Consistent UI Across Mobile Platforms: Both applications (Android and iOS) shall have the same visual elements and appearance.
Rationale: There should be no confusion in the way information is presented between mobile platforms. This will facilitate consistent data collection independent of mobile device.

7.2.3 Performance

7.2.3.1 Quick Form Data Loading There should be a maximum of a five second lag when retrieving and/or updating form data to the server.
Rationale: The form data elements are not large, especially if each section of the form is loaded individually (as they are viewed). As a result, retrieving or sending this data to the server should be quick.

7.2.3.2 Quick Navigation Within Application: Transitions between pages should occur without observable lag or delay.
Rationale: The application is not computationally intensive, so there should be no difficulty navigating from one view to the next.

7.2.3.3 Local Storage: Every entry from a death scene investigator should be saved locally and backed up when they have a connection.
Rationale: Offline editing is vital to keeping death scene data when no internet connection is available.

7.2.4 Maintainability

7.2.4.1 Detailed Change Documentation: A detailed log should be kept of all changes to the document.
Rationale: Because the project group will only last a semester, we want to ensure that future students will be able to understand the history of changes that have been made to the app and the motivation behind these changes.

7.2.4.2 Expected Attendance at National CDC Convention: At least one representative is expected to attend the national CDC convention in the Spring of 2020, along with Dr. Brown, to present the application.
Rationale: The software developer is able to accurately speak to the mobile applications functionality and can answer any technical questions posed at the conference.

7.2.5 Useability

7.2.5.1 Investigator Photographs: Death scene investigators should be able to attach photographs to the electronic form.
Rationale: Photographs are an important part of the paper form, so the electronic form should include this too.

8. Feasibility

8.1 Introduction

The project we are undertaking involves the completion of both an iOS and Android application. The previous student group was able to develop a front-end to partial completion and implement a basic database. We expect to make minor edits to the iOS application, major changes to the database, and to implement an Android front-end from scratch. The existing state of the software (passed on from the previous team) is described in section 2.2.23. We expand on this framework to describe our viable products, both bare bones and enhanced, in the following subsections. The product requirements listed correspond with those mentioned in the Functional Requirements section of this document (Section 6).

8.2 Bare Bones Version

We chose to focus our main efforts on *polishing* and *finalizing* the Apple mobile application. Our customer's primary goal is to get a robust, functional version of the application deployed within Jefferson County, Kentucky for Beta testing. All of the coroners from Jefferson County reportedly have iPhones, so we focus on iOS development to meet immediate user demands. As a result, our minimal viable product will maximize its chances of getting published in the Apple store and its robustness for field use.

1. Editing Permissions
2. Security Clearance
3. Application Structure (implemented for both iPhone)
4. Local Storage and Backup (implemented for both iPhone)
5. Version Control

8.3 Enhanced Version

The latter part of the semester will focus on developing the Android application to mimic the polished iOS application. This is another reason we aim to perfect the iOS application before beginning development in Android; the team will now have a reference template for exactly how the front-end should appear and behave.

1. Application Structure (implemented for Android)
2. Local Storage and Backup (implemented for Android)
3. Navigability between form sections (implemented for both iPhone and Android)
4. Autocomplete feature (implemented for both iPhone and Android)
5. Death Scene Photograph Support (implemented for at least mobile platform)
6. Data Encryption

8.4 Team Member Capabilities

iOS: *Evan and Trent*
Although no team members have extensive iOS development experience, the majority of development on this end is complete. Evan and Trent plan on implementing the missing features from the iOS application and work on polishing the aesthetics of the user interface to meet Apple's Human Interface Guidelines since they are the only two members on the team with Macs and access to Xcode.

Database and REST APIs: *Eura and Matthew*
Eura and Matthew have taken a databases course and have experience using MySQL and PHP, which are the same technologies that the previous team used for developing the backend of the application.

Android: *All members*
Once the bare-bones product is complete with the database and iOS edits, all team members will flock to developing the Android application. All members of the team are familiar with Java and have some previous Android application development experience.

9. Conclusion

We are seeking to equip Kentucky's counties with the first (nationwide, according to Dr. Brown) mobile application for filling out death scene investigation data. We first need to finish the iOS application and its associated backend and begin testing it in Jefferson County—Kentucky's largest county. Once that testing is complete, we will begin working on the Android application. When we have both platforms completed and refined, the application will be released to the rest of the counties within Kentucky. With respect to scalability, we hope that all states will adopt this method of reporting to expedite and standardize death scene reporting.

Having a nationwide, standardized method of reporting death scene data will enhance interstate collaboration on investigations. More metrics on death scenes will be available nationwide, which could be analyzed to prevent similar, future death cases. Nationwide onboarding may be a task for later groups or even larger organizations, but Team 6 is humbled to be a part of this initiative.

10. Appendix

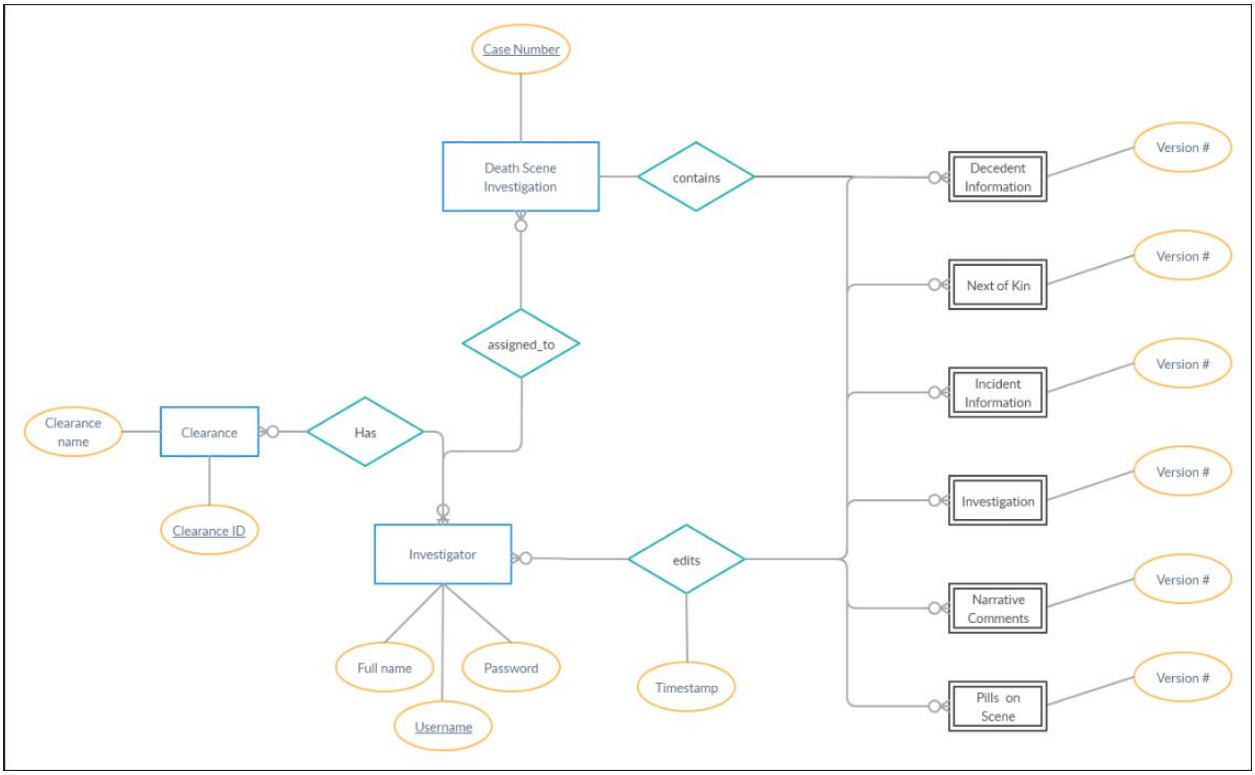

Figure 3. An Entity-Relation Diagram highlighting the major entities, relationships, and attributes in our backend for the application.

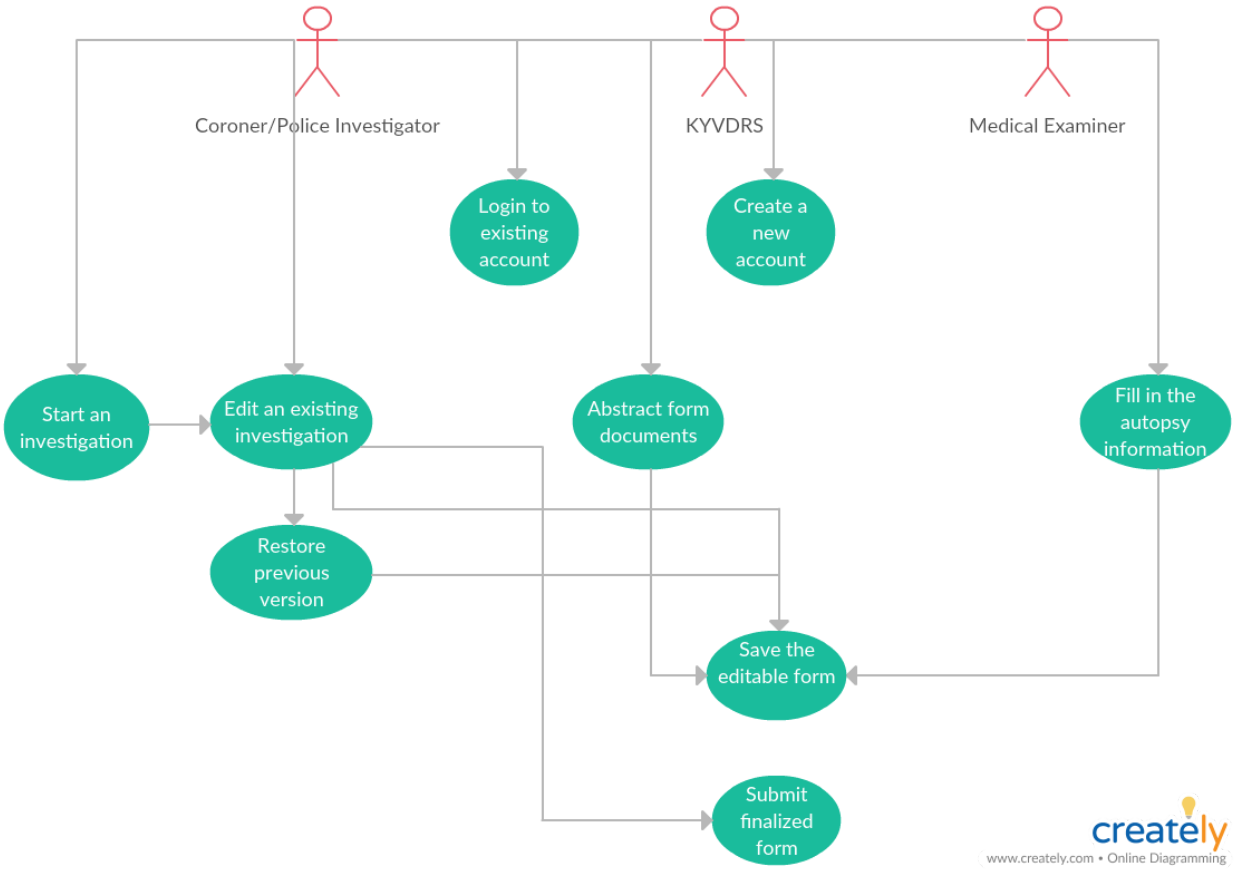

Figure 4. Use case diagram showing users of our application and how they interact with it to fill out the death scene investigation form.

11. References

Brown, Sabrina. "*Death Scene Investigation (DSI) Mobile Application."* Senior Design Project Proposal. University of Kentucky. College of Engineering, Dept. of Computer Science; College of Public Health, Dept. of Epidemiology. Lexington, KY. n.d. Print.