Lab 2 Section 3 – Product Requirements

Team Copper

Old Dominion University

CS411W

Professor James Brunelle

March 19, 2021

Section 3 Version 2

Table of Contents

List of Figures

List of Tables

3 Specific Requirements

3.1 Functional Requirements

3.1.1 User Authentication (O: Yinka M: Grissom)

Upon opening the application, the user must be brought to a welcome page from which the user can select *Login*, *New User*, *Resources*, *Forgot Username*, or *Forgot Password*. The following requirements for Account Creation address the process a new user goes through to make an account when the user selects *New User* from the welcome page. The following requirements for Login address the process a user goes through to login to their account after selecting *Login* from the welcome page.

3.1.1.1 Account Creation

3.1.1.1.1 The application must require the user's first and last name.

3.1.1.1.2 The application must require the user to create a unique username between 2 and 20 characters in length.

3.1.1.1.3 The application must require the user to create a unique password 8-20 characters in length, containing at least one uppercase letter, one lowercase letter, one digit, and one special character.

3.1.1.1.4 The application must require the user to enter a valid email that may be used for account recovery in the event of a forgotten username or password.

3.1.1.1.5 The application must store user credentials in the Care Corner database.

3.1.1.2 Login

3.1.1.2.1 The application must collect the username from the user.

3.1..1.2.2 The application must collect the password from the user.

3.1.1.2.3 The application must confirm the provided username is stored in the database.

3.1.1.2.4 The application must confirm the provided password matches the password associated with the username in the database.

3.1.2 Panic Button (O: Turner M: Carpenter)

User's who have created and logged into their account must have access to the Panic Button. Access to this feature should be located from Care Corner's main menu, from the Fake Phone Call screen and from the Armed Safe Walk screen. Upon activation of the Panic Button, the application must:

1. Start a five second countdown timer before the button is activated, with an option to cancel the activation of the button.
2. Activate audio/video recording
3. Time stamp audio/video recording activation
4. Retrieve the user's location via GPS
5. Gather text message information including the pre-set message and user's location
6. Notify the user's pre-set emergency contacts via text message

3.1.2.1 Incident Creation (O: Turner M: Carpenter)

Upon deactivation of the Panic Button feature through the user ending the audio/video recording, the user must be prompted to input whether the scenario that caused them to activate the Panic Button was an incident that needs to be recorded, or if the scenario can be disregarded. If the user inputs that this scenario's details should be recorded, then a new incident must be added to Care Corner's cloud database. The user is then taken to a screen that has an option to go to the journal to write down any details from the incident, call any of the emergency contacts listed, or return to the main menu. The incident creation must include the following information to be stored as part of it:

1. The user's ID must be stored in the database.
2. The time that the Panic Button was activated must be stored in the database.
3. The time that the Panic Button was deactivated must be stored in the database.
4. The GPS data from the duration of the Panic Button's activation must be stored on the server.
5. The GPS location data should be downloaded.
6. The GPS location data should be stored in the Care Corner cloud.
7. The audio recording from the duration of the Panic Button's activation must be stored on the server.
8. The address of the audio recording on the servers must be stored in the database.

3.1.2.2 GPS Location (O: Turner M: Carpenter)

The application must activate the user's GPS location from the time the Panic Button feature is activated until it is deactivated. The user's location must be:

1. Shared with the user's in-app contacts upon activation.
2. Tracked from the time the Panic Button is activated until the Panic Button is deactivated.
3. Stored locally on the user's phone until the user's input whether the scenario that caused them to activate the Panic Button was an incident (3.1.2.1) or not.
4. Stored remotely in Care Corner's cloud database if user inputs that the scenario that caused them to activate the Panic Button was an incident (3.1.2.1).

3.1.2.3 SMS messaging (O: Turner M: Carpenter)

The application must send a pre-set sms message to the user's in-app contacts when the Panic Button feature is activated. The SMS message must include:

1. .The user's name
2. The pre-set message content
3. The user's location
4. The exact time the Panic Button was activated.

3.1.2.4 Audio and Video Recording (O: Turner M: Carpenter)

The application must start audio recording from the time the Panic Button is activated until the time that it is deactivated. This recording must be:

1. Stored locally on the user's phone until the user has input whether the scenario that caused them to activate the Panic Button was an incident (3.1.2.1) or not.
2. Stored remotely in Care Corner's cloud database if user inputs that the scenario that caused them to activate the Panic Button was an incident (3.1.2.1).

3.1.3 Armed Safe Walk (O: Prudner, M: Webb)

The Armed Safe Walk functional area provides monitoring of a user's walk and notifies contacts about the progress of the planned path. The following functional requirements must be provided.

Upon activating Armed Safe Walk, the application must:

1. Request the user's destination and estimated time of arrival:
   a. Display a prompt to capture the user's destination address.
   b. Display a prompt to capture the user's estimated time of arrival (ETA)
2. Send the destination and ETA to the Care Corner API.
3. Record the user's destination and ETA in the database.
4. Track the user's location throughout the walk.
5. Notify the user's contacts of the status of the walk.
6. Record audio and video of the walk.

7. Provide access to the Panic Button during an Armed Walk.

3.1.3.1 GPS

The application must begin recording the user's location when Armed Safe Walk is activated.

1. The location must be captured using the Android GPS API.

2. The application must ask permission to capture the user's location.

   a. The application must request permissions to access fine location and access background location capture.

3. The application must allow the user to select the following location options:

   a. "Allow all the time": Option to enable sharing the user's permission anytime the application is used.

   b. "Only this time": Option to provide location sharing one time.

   c. "Deny": Option to deny sharing user's location.

4. When the user permits sharing location:

   a. The application must send a location message to the Care Corner API.

   b. The API must record in the database the start of an Armed Walk for the user, recording the location and timestamp.

5. When the user denies sharing the location, the application must cancel the Armed Walk functionality.

3.1.3.2 Notifications

The application must notify and provide the user's location to all in-app contacts when Armed Safe Walk is activated, periodically throughout the walk, and upon arrival.

1. The application must send a location message to the Care Corner API.
2. The API must read the list of the user's contacts from the database.
3. The API must send a SMS to each contact in the list that contains.
4. The API must use Twilio to send a SMS message with:
    a. The user's name (potentially anonymized)

    b. The current location of the user

    c. A current timestamp of the user

    d. The estimated time of the user's arrival
5. The API must send a SMS message every 3 minutes with the state of the walk.
6. When the user arrives at the destination, the API must send an SMS message:
    a. The user's name (potentially anonymized)

    b.The current location of the user

    c. A current timestamp of the user

    d. A note that the user reached their destination

    e. The time of the user's arrival at their destination

3.1.3.3 Audio/Video Recording

The application must begin recording audio/video when Armed Safe Walk is activated.

1. The audio/video must be captured using the Android Media Recorder.
2. The application must ask permission the first time to record audio/video.

3. The application must remember the user's permission choice.

4. The application must begin recording audio/video.

5. The application must store the audio/video to a local file store.

6. When the user reaches their Armed Walk destination:

    a. The application must ask the user to back up the audio/video.

    b. When the user confirms they want to backup their audio/video:

        1. The application must stream the audio/video to the API.

        2. The API must store the audio/video in a AWS S3 bucket.

        3. The API must record the timestamp, file location, and name of the audio/video.

3.1.4 Fake Phone Call (O: Webb M: Carpenter)

The Fake Phone Call feature provides a way to get away safely from awkward or potentially dangerous situations by simulating a phone call for the user to have an excuse to leave. The following functional requirements must be provided. Upon activating the Fake Phone Call, the application must:

3.1.4.1 Collect what name is pre-set by the user when a Fake Phone Call is received.

3.1.4.2 Collect what phone number is pre-set by the user when a Fake Phone Call is received.

3.1.4.3 Simulate a phone call through initiating a ringer with options to answer or decline the phone call.

3.1.4.4 Simulate a phone call when the answer button is pressed through immediately starting the fake phone call audio.

3.1.4.5 Activate the microphone and record the audio of the Fake Phone Call to keep a record incase the situation transitions into panic mode

3.1.4.6 Activate the camera and record the video of the Fake Phone Call before the fake call starts.

3.1.4.7 Activate panic mode when the end call button is held down for a set amount of time.

3.1.4.8 Have multiple fake conversations for the user to choose from for different types of situations that will output from the user's receiver.

3.1.5 Journal (O: Carpenter M: Adegun)

The journal functions must provide the user a private place to put their thoughts into words. Access to the journal screen is found through a button on the Care Corner home screen once the user logs into their account. The following functional capabilities must be provided.

3.1.5.1 The application must keep the Journal protected by providing creation of personal identification number (PIN) upon initial attempt to access the journal.

3.1.5.2 The application must provide the user the option of resetting their PIN in case the user forgets their PIN.

3.1.5.3 The application must require the user to authenticate by logging into their Care Corner account again prior to resetting the pin.

3.1.5.4 The application must also require the user to authenticate by logging into their Care Corner account again after 30 minutes of inactivity prior to being able to access the journal.

3.1.5.5 The application must provide a time-stamped capability to allow the user to:

3.1.5.5.1 Create new entries.

3.1.5.5.2 Edit existing entries.

3.1.5.5.3 Delete existing entries.

3.1.5.5.4 View a list of previously created journal entries.

3.1.5.5.5 Save new entries to the Care Corner database.

3.1.5.5.6 Make newly saved entries accessible from the journal homepage.

3.1.5.5.7 Save edited entries to the Care Corner database.

3.1.5.5.8 Return to the main menu from the journal homepage.

3.1.6 Mombot (O: Grissom, M: Prudner)

The Mombot functions must provide the user with helpful advice in response to the user's plans or activities. The following functional requirements must be provided.

3.1.6.1 The application must request speech input from the user.

1. The application must display a ‘Tap on mic to speak’ button.
2. The application must start receiving speech when pressed.

3.1.6.2 The application must convert speech to text.

1. The application must use the Android Speech API to convert the speech.

3.1.6.3 The application must identify keywords from the input to return the related advice.

1. The application must accept a string as input.
2. The application must send the input to the Care Corner API.
3. The API must reference a set of keywords from the database.
4. The API must use lexical analysis to match the set of keywords.
5. The API must look up advice in the database for the matched keywords.

3.1.6.4 The application must provide the user with a suggested checklist of things to go over before the user goes out.

3.1.7 Reporting Assistance (O: Carpenter)

The reporting assistance functions must provide the user with information on the different ways to report the assault. The reporting assistance feature will be available inside of the resources feature in the form of a button the user has to click, or after panic mode has been deactivated. The following functional requirements must be provided.

3.1.7.1 The application must provide a questionnaire to properly record details on any incident.

3.1.7.1.1 The application questionnaire must include the following topics:

a. Date of the assault
b. Time of the assault
c. Location of the assault
d. Possible journal entry to include that talks about the assault.

3.1.7.2 The application must provide basic information that comes from the resources feature about how the user can go about reporting the assault.

3.1.8 Resources (O: Grissom, M: Prudner)

The resources function must provide the user with trusted resources related to assault. Trusted resources will be government sources or non-profits. The following functional requirements must be provided.

3.1.8.1 The application must provide the Resources feature without requiring authentication.

3.1.8.2 Upon activating Resources, the application must:

1. Obtain a current list of resources from the Care Corner API.
2. Cache the set of resources in local storage for later reference.
3. Display a list of resource categories for the user to choose from.

3.1.8.3 When the blog category is selected, the application must provide a listing of resources relating to assault in the form of trusted blogs.

3.1.8.4 When the national hotline category is selected, the application must provide a listing of resources relating to assault in the form of national hotlines.

3.1.8.5 When the government sources category is selected, the application must provide a listing of resources relating to assault in the form of government sources.

3.1.8.6 When the shelters or counselors category is selected, the application must capture a geofence of the user's current location.

1. The application must ask permission to capture the user's location.

   a. The application must request permissions to access fine location and background location capture.
2. The application must guide the user to their setting page to set the location permissions for the Care Corner application.

   a. Android API level 30 and up requires setting background location permissions via the user's setting instead of a dialog when basic location settings.
3. When the user permits sharing location:

   a. The application must create a geofence using the Android GeoFence API with a 90 miles radius..

3.1.8.6 When the shelter category is selected, the application must provide shelters based on the user's geofenced location.

3.1.8.7 When the counselors category is selected, the application must provide counselors based on the user's geofenced location.

3.2.Performance Requirements

3.2.1 Application Performance

3.2.1.1 The application must be written efficiently enough to land the user on the home screen of the application within 5 seconds of opening it. (O: Turner M: Grissom)

3.3.Assumptions and Constraints

3.3.1 The application requires internet access to store and retrieve data and files from the cloud servers and database. (O: Carpenter M: Adegun) If this constraint turns out to be false, it would affect the requirements by limiting the functionality of the application. Data will be stored locally on the user's device so previously located resources can still be accessed and the user can still record their trips with audio and video.

3.3.2 The application requires Android KitKat (4.4) OS or higher. (O: Adegun M: Grissom)

3.3.3 The application requires access to a functioning microphone on the user's device. (O: Grissom) If this constraint turns out to be false, it would affect the requirements by limiting the functionality of the application. Audio will no longer be recorded and stored.

3.3.4 The application requires the user's device to have a functioning rear facing camera. (O: Webb M: Carpenter) If this assumption turns out to be false, it would affect the requirements by limiting the functionality of the application through not being able to record and store video recordings.

3.4.Non-Functional Requirements

3.4.1 Security

3.4.1.1 Passwords (O: Prudner M: Webb)

Passwords must be used to secure protected areas of the application.

A personal identification number (PIN) must be used to access the journal area of the application.

3.4.1.1.1 Protection of account

The application must require a password to access an account.

When accessing an account, an authentication form will be presented for the usr to enter a password.

The application must prompt for a password when a user is logging in or a session has timed out.

3.4.1.1.2 Protection of journal

The application must require a PIN when accessing the journal.

When accessing the journal, a form must prompt the user to enter a valid PIN.

3.4.1.1.2 Password complexity requirements

Passwords must follow the Open Web Application Security Project (OWASP) guidance for complexity:

1. A password must be a minimum of 8 characters.

2. A password must be a maximum of 64 characters.

3. The application must allow usage of all characters for passwords.

3.4.1.2 API (O: Turner M: Grissom)

The APIs of the application will be managed through AWS API Gateway. The APIs will be RESTful and will conform to the REST architectural style.

3.4.1.2.1 API Keys (O: Turner M: Grissom)

The API gateway will be designed to require that the appropriate API keys be included for any API requests passed to the server.

3.4.2 Maintainability (O: Carpenter)

Care Corner has a low-maintenance procedure to keep updated through allowing an easy update and maintenance of system servers. Additionally, Care Corner follows internet security protocols and information security guidelines. Maintenance procedures also include verifying that websites and phone numbers stay up-to-date. These maintenance procedures are conducted semiannually.

3.4.3 Reliability (O: Adegun)

3.4.3.1 Fault tolerance

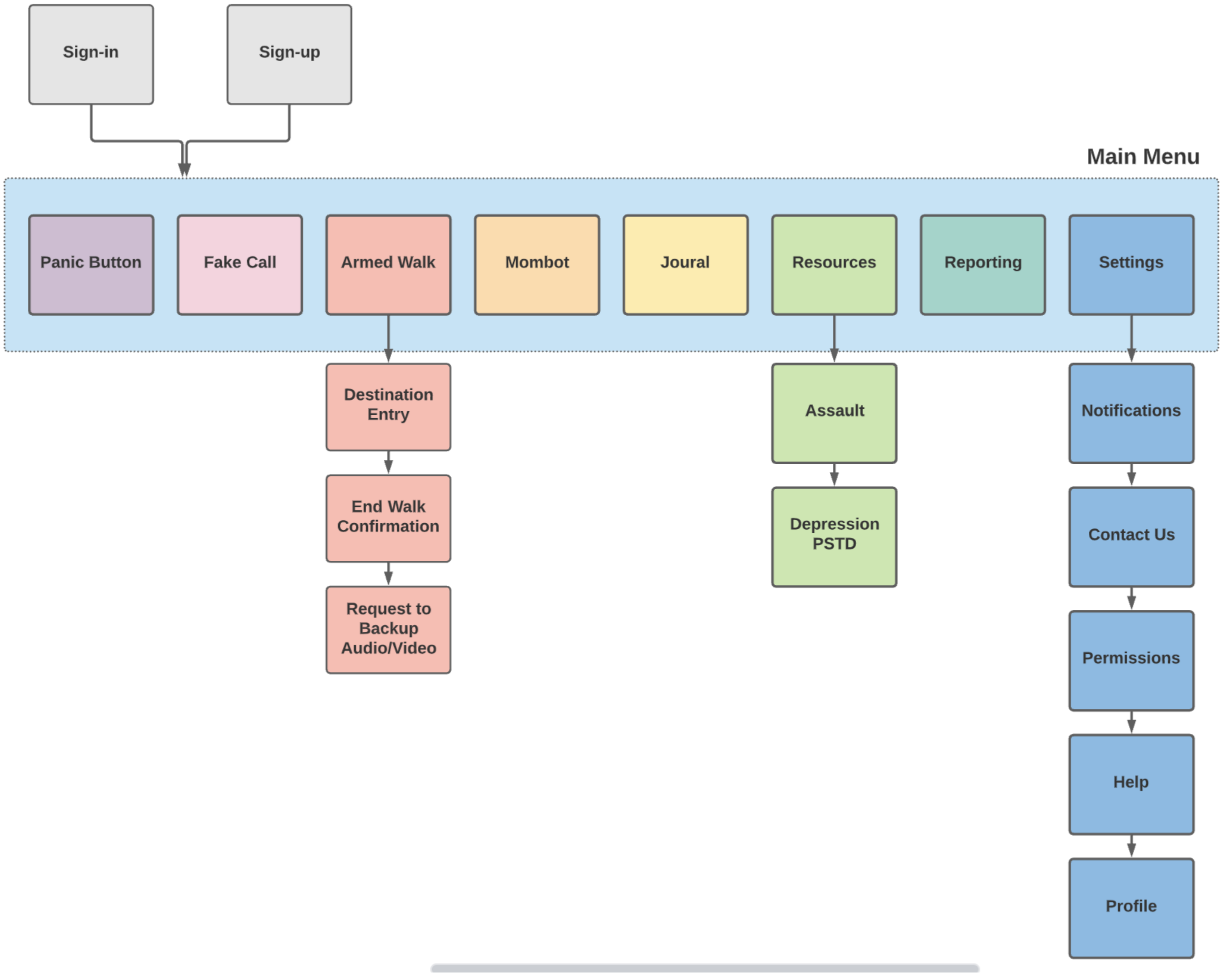
3.4.3.1.1 AWS Serverless

3.4.3.1.2 Five nines: 99.999%

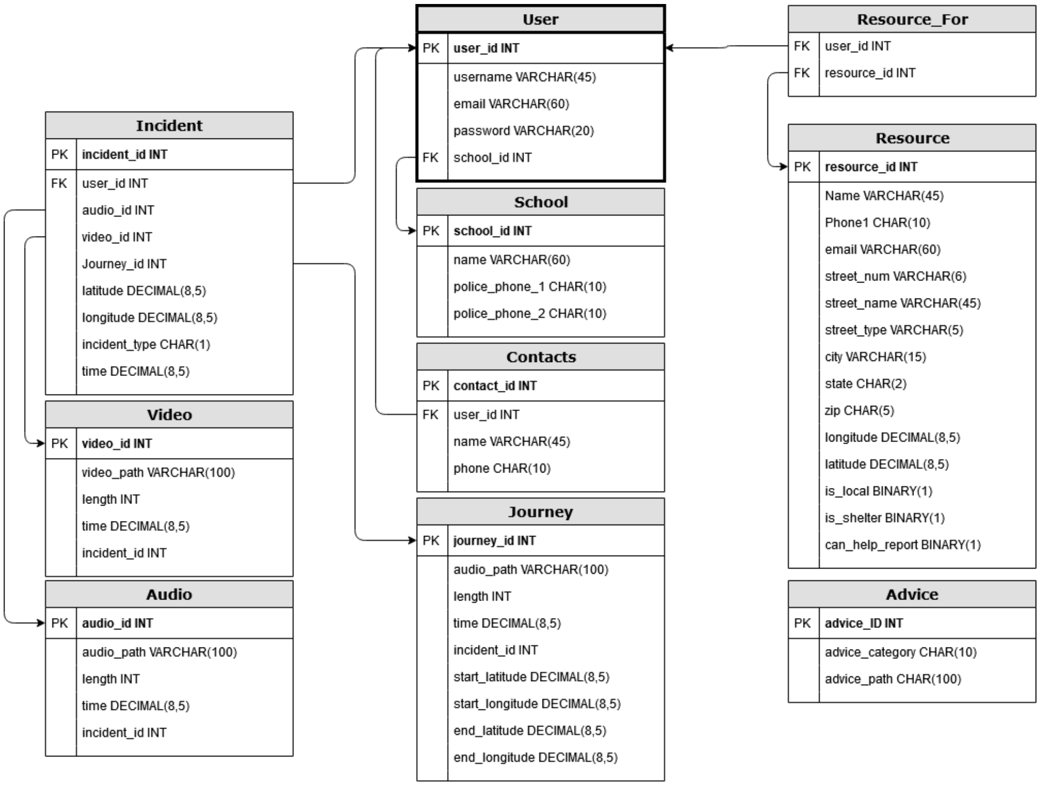
3.4.3.1.3 Multi-region

3.4.3.2 Database backups

Appendix A - Site Map



Appendix B - Entity Relationship Diagram



User	
PK	**user_id INT**
	username VARCHAR(45)
	email VARCHAR(60)
	password VARCHAR(20)
FK	school_id INT

Resource_For	
FK	user_id INT
FK	resource_id INT

Incident	
PK	**incident_id INT**
FK	user_id INT
	audio_id INT
	video_id INT
	Journey_id INT
	latitude DECIMAL(8,5)
	longitude DECIMAL(8,5)
	incident_type CHAR(1)
	time DECIMAL(8,5)

School	
PK	**school_id INT**
	name VARCHAR(60)
	police_phone_1 CHAR(10)
	police_phone_2 CHAR(10)

Resource	
PK	**resource_id INT**
	Name VARCHAR(45)
	Phone1 CHAR(10)
	email VARCHAR(60)
	street_num VARCHAR(6)
	street_name VARCHAR(45)
	street_type VARCHAR(5)
	city VARCHAR(15)
	state CHAR(2)
	zip CHAR(5)
	longitude DECIMAL(8,5)
	latitude DECIMAL(8,5)
	is_local BINARY(1)
	is_shelter BINARY(1)
	can_help_report BINARY(1)

Contacts	
PK	**contact_id INT**
FK	user_id INT
	name VARCHAR(45)
	phone CHAR(10)

Video	
PK	**video_id INT**
	video_path VARCHAR(100)
	length INT
	time DECIMAL(8,5)
	incident_id INT

Journey	
PK	**journey_id INT**
	audio_path VARCHAR(100)
	length INT
	time DECIMAL(8,5)
	incident_id INT
	start_latitude DECIMAL(8,5)
	start_longitude DECIMAL(8,5)
	end_latitude DECIMAL(8,5)
	end_longitude DECIMAL(8,5)

Audio	
PK	**audio_id INT**
	audio_path VARCHAR(100)
	length INT
	time DECIMAL(8,5)
	incident_id INT

Advice	
PK	**advice_ID INT**
	advice_category CHAR(10)
	advice_path CHAR(100)