# SchoolSafe

# Table of Contents

**Project Proposal/Motivation/Vision Statement**

For educators that want to provide the safest learning environment in an age where violence in the classroom has become more commonplace, SchoolSafe is a mobile application that allows current emergency alerts, headcounts, and live situation updates via chat to be silently exchanged between educators in the classroom and administrators in the office about active emergency situations on the campus. Unlike current solutions, our software allows administrators to provide updated information to police and rescue personnel while also keeping the rest of the staff informed during an event.

**Group Members**

Johnny Northrup – Project Lead

Daniel Leach – Lead Front End Developer

Alan Tessier – Front End Developer / Full Stack Developer

Claudio Herrera – Database Manager

Preston Small - Web Programmer / Research Manager

Brandon Deringer - Web Programmer / Documentation Manager

## Stakeholder Definitions

| Type | Description |
|---|---|
| Parents | Parents access the system to use its various modules including Messaging, get emergency alerts, access the Student Accountability module, and district calendar. |
| Educators | Educators can access and modify different modules in the system. They can send but cannot dismiss emergency alerts. They can modify the student count in the Student Accountability module. They can communicate using the messaging module. |
| Administrators | Administrators have full access to SchoolSafe. These users can assign user permissions, and view or modify all modules. Administrators can also dismiss security alerts. |
| Development Team | The development team also has access to the full functionality of the application. In addition, they have access to the source code and database that runs the system. |

**Statement on Developmental Model**

For our project we are using a hybrid of plan-driven and agile methods. The plan-driven portion of our method is based on the fact that we have deadlines we have to meet that are set by the course schedule. We chose to also go with an agile approach, because much of what we are doing will be tested as we go by both the front and backend teams. The agile method also allows us to fully utilize the incremental development model while also using some paired programming in the creation of the app and database. Our model is the incremental development model. We started by creating an outline of what our program is and then started into the specification phase. We chose this model because we all had some really great ideas for features, but we all understand that we don't have a ton of time to implement everything. This model gives us the freedom to design the app with the basic functions we originally want and then create newer versions with more updated features while keeping the ability to have a finished product if we run out of time.

**Statement on Distributed Model**

The system we are using is Software as a Service. We chose this as it made the most sense, mainly because we are utilizing a main server and thin web-based clients.

**1.      Resource sharing**

All of our resources are stored on the server and accessible from any of the client apps.

**2.      Openness**

With the app being connected to the server, a quick auto-update is all it needs to ensure the client has all new data as quickly as possible.

**3.      Concurrency**

By working with isolations, specifically locking a record while being edited, we should be able to cut down on quite a few issues with concurrent record updating. The system will grab the record, lock it for updating, validate it and then save it on update. If the system detects a concurrent update, it will alert the user and refresh their record.

**4.      Scalability**

As far as scalability goes, our project is using a database that allows the administrators to add new users and remove parents of students no longer attending their school. We have designed this portion to ensure ease of use for the customer. With our project being a web-based app, the ability to access the app isn't hindered by having to be at a set location which makes distribution easier.
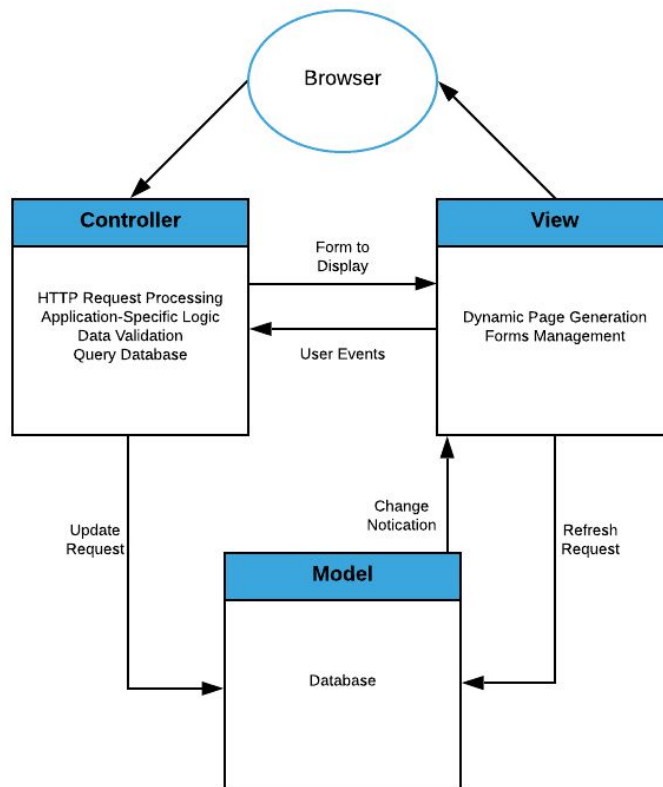
**5.      Fault Tolerance**

With our app being web-based, we are able to push updates and work problems in real time. Each piece of our app is modular so we can work them separately when needed, hopefully without too much interruption in the other services it provides.

**Statement on Security**

Security at SchoolSafe is one of our top priorities. Our team is devoted to not only keeping students safe, but also to ensuring the security of our users' data. We have taken several measures to ensure that their data will not fall into the wrong hands and potentially pose a threat to their privacy or become victims of stolen identity. The following are just a few of the methods that we have implemented to ensure the security of all the users on our platform:

We do not store any sensitive information in plain text. All passwords are salted and hashed using the SHA-256 hash function. The SHA-2 family of hash functions is one of the Secure Hash Algorithms required by law for use in certain US government applications. In addition, our engineers have implemented input validation functions which prevent SQL injection attacks. Along with input validation, the implementation of error handling allows the user to receive instant feedback should there be any errors while using SchoolSafe. Lastly, our user level feature protects users data, and prevents lower level users from accessing unnecessary features.

**System Architecture Model**



Browser

Controller
HTTP Request Processing
Application-Specific Logic
Data Validation
Query Database

View
Dynamic Page Generation
Forms Management

Form to Display

User Events

Update Request

Change Notication

Refresh Request

Model
Database

MVC Architectural Pattern

**User Requirements with Corresponding System Requirements**

## User Requirements

New Users

·       Must register in order to access web application's home page.

·       User level must be set to 0.

·       Cannot access any functionality of the web app until the user level has been upgraded by the admin.

Educator

·       Must log in with a verified account with the appropriate user level of 2 to access the web application.

·       Must be able to connect their district calendar via google calendars.

·       Able to update and access the student attendance.

·       Ability to message other users.

·       Ability to retrieve their user level.

·       Can view and shortly dismiss the emergency banner on their screen only.
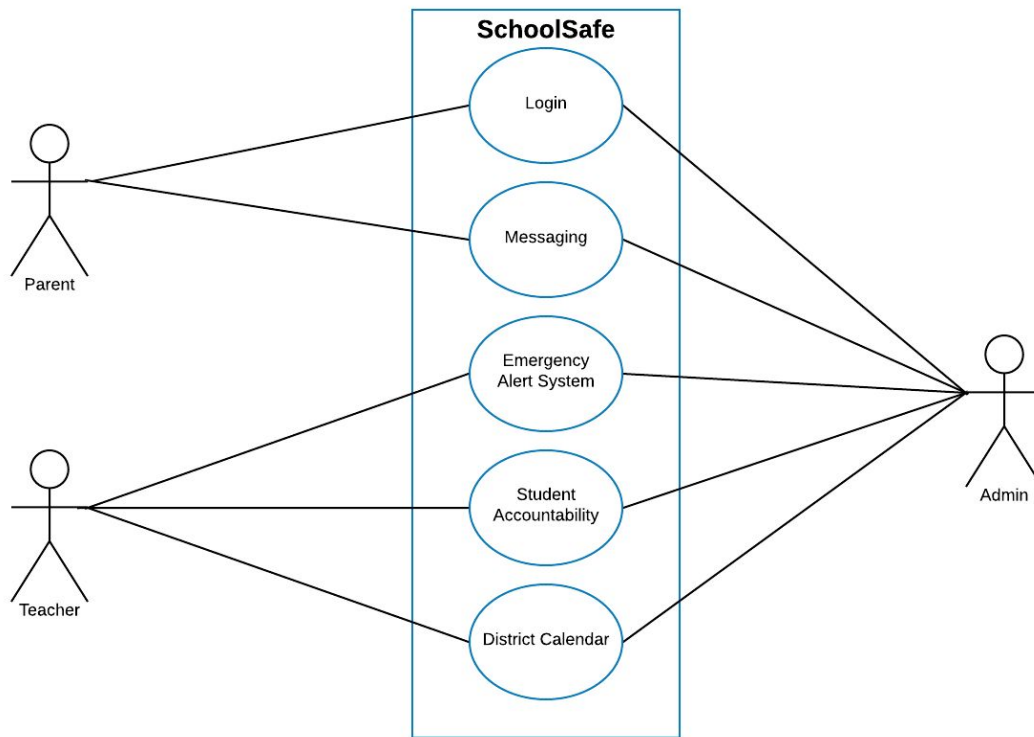
·       Can contact for help.

Admin

·       Must log in with a verified account with the appropriate user level of 1 to access the web application.

·       Has all the rights of an Educator (User Level 2).

·       Able to call and dismiss an emergency globally.

·       View and change a new user or educator user's level.

## System Requirements

· The system must be able to salt and hash passwords for security

· Create a database using Oracle DB for the logins, students, and alerts.

· Web application must be responsive for mobile views.

· There must be 3 user levels and every user must be assigned one.

   o  0 - Basic Access (no permission)

   o  1 – Admin (full Access)

   o  2 – Educator (limited access)

· New users cannot access the web application until their user level has been upgraded.

· Realtime database to store and update the user's messages.

# Use-Case Diagram

# Sequence Diagrams

# Class Diagram

**Validate**

+ $conn : db connection
+ $error : boolean
+ $username : String
+ $email : Email
+ $pass : String
+ $firstname : String
+ $lastname :String
+ $salt : String
+ $pasword : hash
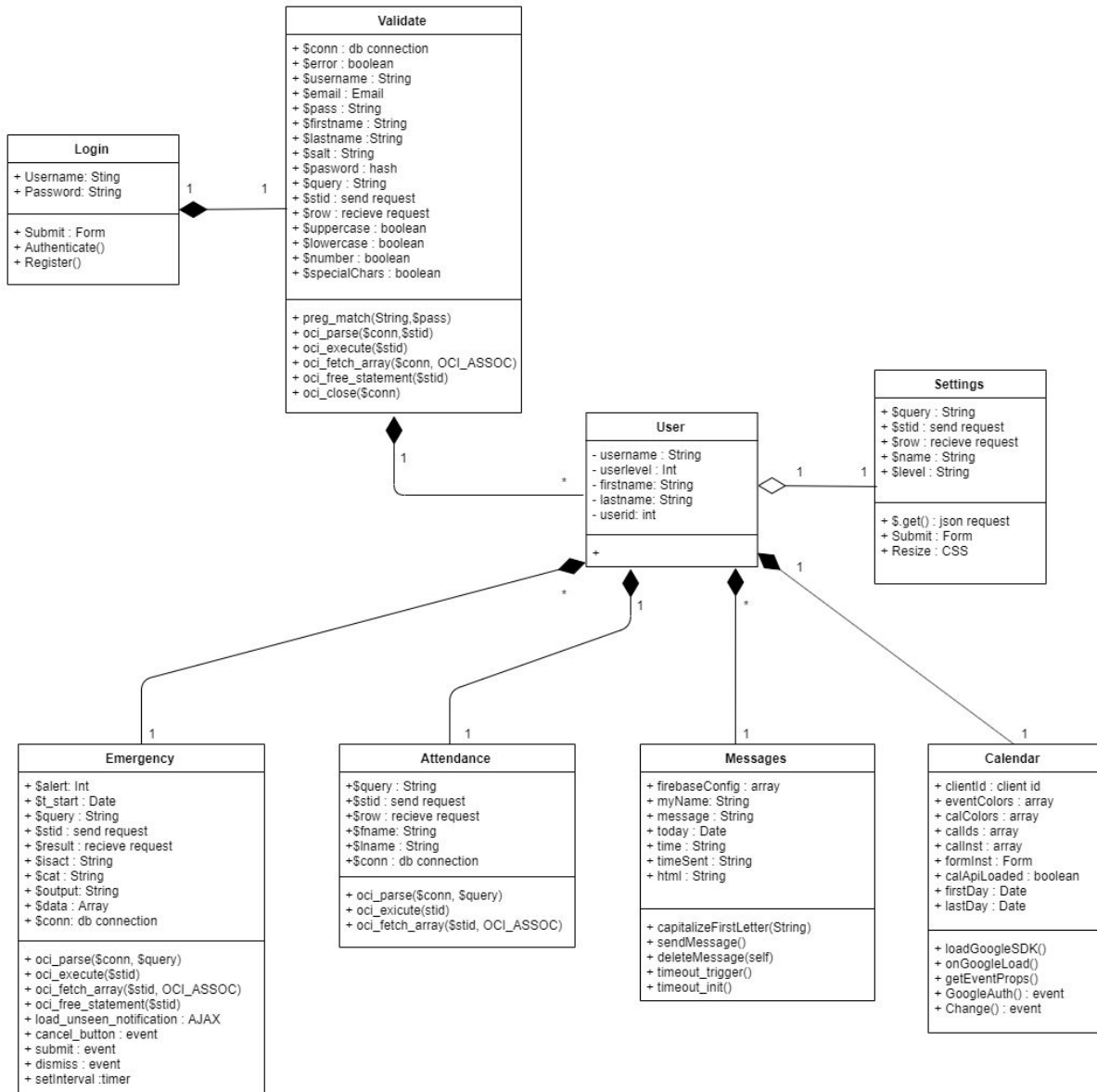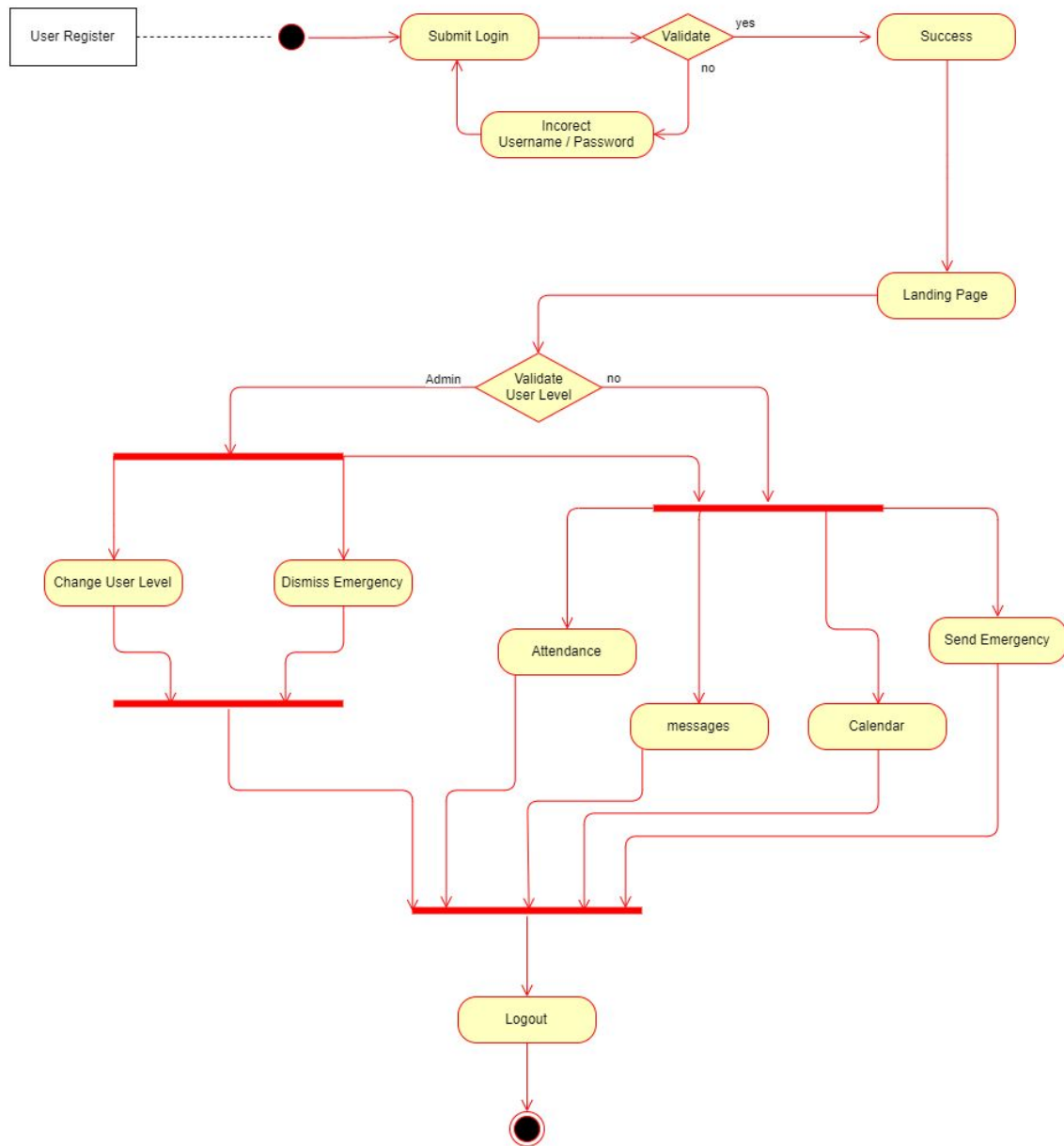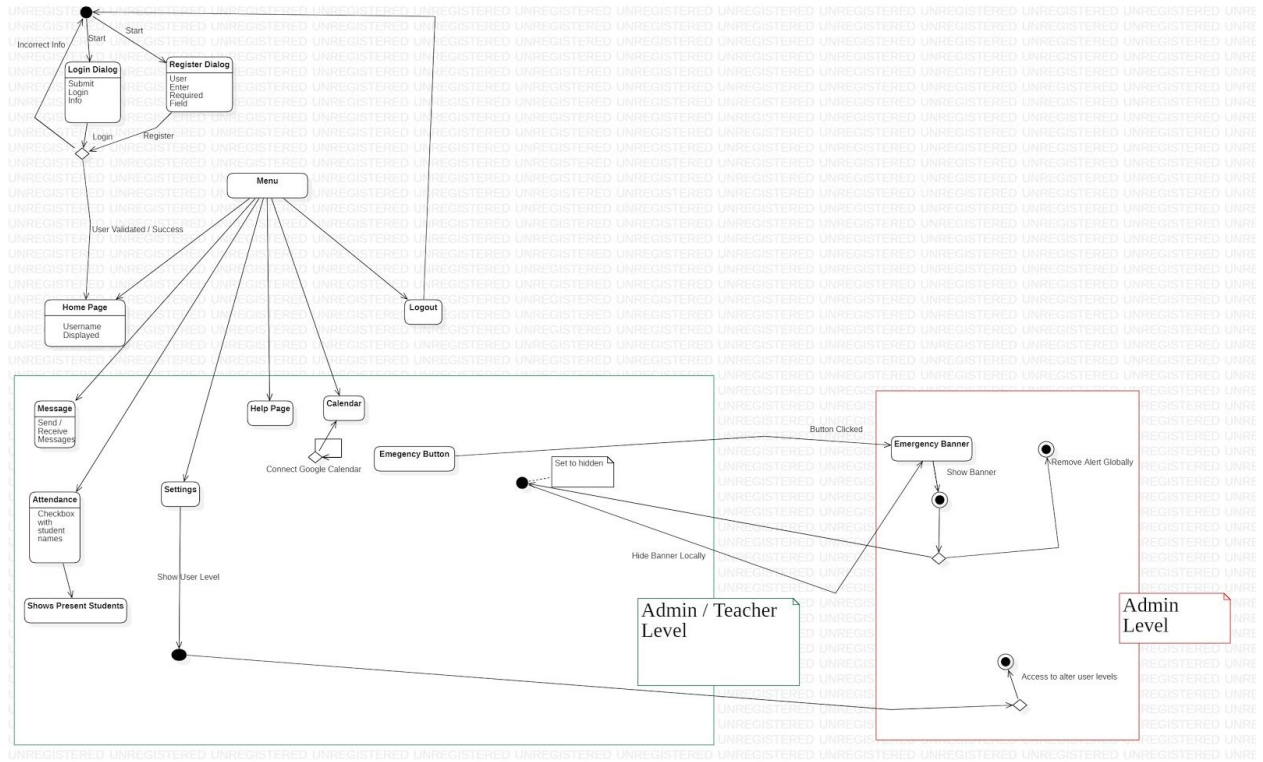+ $query : String
+ $stid : send request
+ $row : recieve request
+ $uppercase : boolean
+ $lowercase : boolean
+ $number : boolean
+ $specialChars : boolean

+ preg_match(String,$pass)
+ oci_parse($conn,$stid)
+ oci_execute($stid)
+ oci_fetch_array($conn, OCI_ASSOC)
+ oci_free_statement($stid)
+ oci_close($conn)

**Login**

+ Username: Sting
+ Password: String

+ Submit : Form
+ Authenticate()
+ Register()

**User**

- username : String
- userlevel : Int
- firstname: String
- lastname: String
- userid: int

+

**Settings**

+ $query : String
+ $stid : send request
+ $row : recieve request
+ $name : String
+ $level : String

+ $.get() : json request
+ Submit : Form
+ Resize : CSS

**Emergency**

+ $alert: Int
+ $t_start : Date
+ $query : String
+ $stid : send request
+ $result : recieve request
+ $isact : String
+ $cat : String
+ $output: String
+ $data : Array
+ $conn: db connection

+ oci_parse($conn, $query)
+ oci_execute($stid)
+ oci_fetch_array($stid, OCI_ASSOC)
+ oci_free_statement($stid)
+ load_unseen_notification : AJAX
+ cancel_button : event
+ submit : event
+ dismiss : event
+ setInterval :timer

**Attendance**

+$query : String
+$stid : send request
+$row : recieve request
+$fname: String
+$lname : String
+$conn : db connection

+ oci_parse($conn, $query)
+ oci_exicute(stid)
+ oci_fetch_array($stid, OCI_ASSOC)

**Messages**

+ firebaseConfig : array
+ myName: String
+ message : String
+ today : Date
+ time : String
+ timeSent : String
+ html : String

+ capitalizeFirstLetter(String)
+ sendMessage()
+ deleteMessage(self)
+ timeout_trigger()
+ timeout_init()

**Calendar**

+ clientId : client id
+ eventColors : array
+ calColors : array
+ calIds : array
+ calInst : array
+ formInst : Form
+ calApiLoaded : boolean
+ firstDay : Date
+ lastDay : Date

+ loadGoogleSDK()
+ onGoogleLoad()
+ getEventProps()
+ GoogleAuth() : event
+ Change() : event

# Activity Diagram

# User Interface Diagrams



### Login / Register Flow

- Start
- Incorrect Info
- **Login Dialog** — Submit Login Info
- **Register Dialog** — User Enter Required Field
- Login
- Register
- User Validated / Success

### Menu

- **Home Page** — Username Displayed
- **Message** — Send / Receive Messages
- **Attendance** — Checkbox with student names
- Shows Present Students
- **Settings** — Show User Level
- **Help Page**
- **Calendar** — Connect Google Calendar
- **Emergency Button**
- Set to hidden
- **Logout**

### Admin / Teacher Level

- Button Clicked
- **Emergency Banner** — Show Banner
- Hide Banner Locally

### Admin Level

- Remove Alert Globally
- Access to alter user levels

16

# Database Design Diagram

### STUDENT

| | | |
|---|---|---|
| P * | SID | NUMBER (*,0) |
| | FNAME | VARCHAR2 (50 BYTE) |
| | LNAME | VARCHAR2 (50 BYTE) |
| | PID | VARCHAR2 (100 BYTE) |
| | ATTENDANCE | NUMBER (*,0) |
| | TID | NUMBER (*,0) |
| | PRESENT | CHAR (1 BYTE) |

☞ SID_PK (SID)

◆ SID_PK (SID)

### LOGIN

| | | |
|---|---|---|
| P * | USER_ID | NUMBER (*,0) |
| * | USERNAME | VARCHAR2 (255 BYTE) |
| * | PASSWORD | VARCHAR2 (255 BYTE) |
| * | EMAIL | VARCHAR2 (50 BYTE) |
| * | FNAME | VARCHAR2 (30 BYTE) |
| * | LNAME | VARCHAR2 (30 BYTE) |
| * | USER_LVL | NUMBER (*,0) |

☞ UID_PK (USER_ID)

◆ UID_PK (USER_ID)

### ALERT_LOG

| | | |
|---|---|---|
| P * | ALERT_ID | NUMBER (*,0) |
| * | T_START | VARCHAR2 (50 BYTE) |
| | T_END | VARCHAR2 (50 BYTE) |
| * | CAT | NUMBER (*,0) |
| * | IS_ACT | CHAR (1 BYTE) |

☞ ALERT_LOG_PK (ALERT_ID)

◆ ALERT_LOG_PK (ALERT_ID)

## Glossary

- Stakeholder - person of interest using the application. Parents, administrators, and educators.

- Model - Manages the Data, Logic, and Rules of an application.

- View - User interface displaying objects within an application.

- Controller - updates both model and view. Accepts input and performs the corresponding updates.

- SHA-256 hash - SHA-2 family of hash functions is one of the Secure Hash Algorithms required by law for use in certain US government applications

- Resource Sharing- A distributed system allows the sharing of hardware and soft-ware resources

- Scalability- distributed systems are scalable in that the capa- bilities of the system can be increased by adding new resources to cope with new demands on the system.

- Concurrency - several processes may operate at the same time on separate computers on the network.

- Openness - Systems designed around Internet protocols so that equipment and software from different vendors can be combined.

**Brochure**

**FAU** DEPARTMENT OF COMPUTER & ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

College of Engineering & Computer Science
Florida Atlantic University

## Theme: Educational Systems

## Project Overview

### Project Name: SchoolSafe

For educators that want to provide the safest learning environment in an age where violence in the classroom has become more commonplace, SchoolSafe is a mobile application that allows current emergency alerts, headcounts, and live situation updates via chat to be silently exchanged between educators in the classroom and administrators in the office about active emergency situations on the campus. Unlike current solutions, our software allows administrators to provide updated information to police and rescue personnel while also keeping the rest of the staff informed during an event.

Our team: Johnny Northrup – Project Lead, Daniel Leach – Lead Front End Developer, Alan Tessier – Front End Developer / Full Stack Developer, Claudio Herrera – Database Manager, Preston Small - Web Programmer / Research Manager, Brandon Deringer - Web Programmer / Documentation Manager

### Course Instructor
Lofton A. Bullard, Ph.D.

Ph.D., Computer Science, Florida Atlantic University, Boca Raton, FL, 2008
M.S., Computer Science, Florida Atlantic University, Boca Raton, FL, 1996
M.S., Education, University of Miami, Miami, FL, 1991
B.S., Computer Science, Florida International University, Miami, FL, 1985
Florida Professional Educator's Certification

Specializing in finding talent in FAU students

The primary objective of this course is to introduce the student to core concepts in software engineering in an effort to prepare the student for industrial scale software systems development and the ability to work in software development teams.

**FAU**
DEPARTMENT OF COMPUTER & ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
College of Engineering & Computer Science
Florida Atlantic University

www.ceecs.fau.edu

# SCHOOLSAFE

## How to Use SchoolSafe

### User Instruction Manual

Presented by

SchoolSafe

# SCHOOLSAFE

## Purpose

For educators that want to provide the safest learning environment in an age where violence in the classroom has become more commonplace, SchoolSafe is a mobile application that allows current emergency alerts, headcounts, and live situation updates via chat to be silently exchanged between educators in the classroom and administrators in the office about active emergency situations on the campus. Unlike current solutions, our software allows administrators to provide updated information to police and rescue personnel while also keeping the rest of the staff informed during an event.

# SCHOOLSAFE
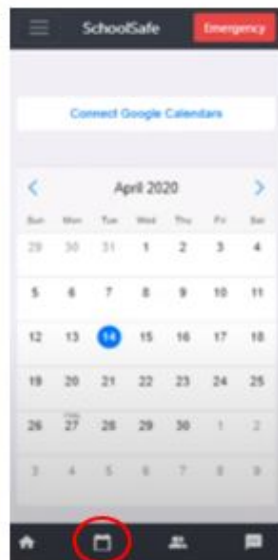


## Calendar

The calendar screen allows you to keep track of the school district calendar, your own upcoming meeting and anything else you would like to place on your schedule. You can also connect your own Google calendar to personalize it even further.

# SCHOOLSAFE



## Accountability

On the accountability screen, you can easily keep track of which students are currently accounted for. Once submitted, the list updates in the database so that the admin level accounts can see who has full accountability in an emergency.

4
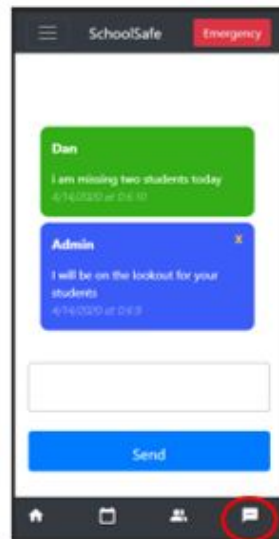
# SCHOOLSAFE



## Messaging

The app also comes with a messenger built in to allow immediate updates between staff members during an event. Parents and teachers can also communicate to provide updates on a student's progress in school.

25

# SCHOOLSAFE



## Emergency

An easy access emergency button is located on the top right corner and is accessible from everywhere in the application. Here you can select from a list of emergencies you may experience in a classroom setting. Upon pressing submit, you are asked to confirm your emergency before it is broadcast to all active user accounts.

(Admin only) Admin are able to dismiss emergencies once they have been cleared by the proper authorities. Only admin accounts are able to see the dismiss button.

# SCHOOLSAFE



## Settings (Admin only)

The settings screen is accessible by selecting it from the drop-down menu on the top left of the application. Here you can set user levels for anyone currently signed up for the app.

# Jon Northrup

1177 Lake Victoria Dr., APT P
West Palm Beach, FL, 33411

(919) 221-5054
r33drift@gmail.com
GitHub
LinkedIn

## OBJECTIVE

I am a Computer Science Major interested in mobile applications, game design, and augmented and virtual reality.

## ADDITIONAL SKILLS

❖ Languages - Java, Python, C, C++, HTML5, CSS3, JavaScript
❖ Data Science - MATLAB, TensorFlow
❖ Highly Skilled Public Speaker

## PROJECTS

SchoolSafe

This was a team project to design an application for more expedient response to school emergencies. The application allows administrators and educators to send real time emergency alerts, class accountability, and messages to all active users.

Cocktail Genie

This is a solo project, stand-alone Android application that acts as a pocket bartender's guide. Cocktail Genie is written in Java, uses an API to store the drink database, and can be searched by name, ingredient, category, etc.

## EDUCATION

### FLORIDA ATLANTIC UNIVERSITY

Boca Raton, FL

*B.S. Computer Science Candidate (Expected graduation Dec 2020)*

- Relevant Coursework: Computer Operating Systems, Data Structures, Mobile Application Development, Algorithm Analysis, Stochastic Models

### PALM BEACH STATE COLLEGE

West Palm Beach, FL

*A.A. Transfer (May 2018)*

### AIR UNIVERSITY

Montgomery, AL

*Completed coursework towards A.S. Aviation Mechanics (Jul 2015)*

## PROFESSIONAL EXPERIENCE

### BOEING, Palmdale, CA

*Aircraft Maintenance Inspector | Nov 2015 - Jul 2016*

- Inspect work of aircraft mechanics performing maintenance, modification, or repair and overhaul of aircraft and aircraft mechanical systems to ensure adherence to standards and procedures.
- Inspect new, repaired, or modified aircraft to identify damage or defects and to assess airworthiness and conformance to standards, using checklists, hand tools, and test instruments.
- Prepare and maintain detailed repair, inspection, investigation, and certification records and reports.
- Examine maintenance records and flight logs to determine if service and maintenance checks and overhauls were performed at prescribed intervals.

### US AIR FORCE, Washington, DC

*Flight Line Maintenance Manager (Expediter) | Jun 2002 - Jul 2015*

- Confer with personnel, such as management, engineering, or quality control, to coordinate work activities, resolve employee grievances, or identify and review resource needs.
- Determine schedules, sequences, and assignments for work activities, based on work priority, quantity of equipment, and skill of personnel.
- Counsel employees about work-related issues and assist employees to correct job-skill deficiencies.
- Recommend or initiate personnel actions, such as hires, promotions, transfers, discharges, or disciplinary measures.

## REFERENCES

References available upon request

Alan Tessier
Hollywood, FL
https://github.com/alantess
97atessier@gmail.com

## OBJECTIVE

Obtain a position as a software engineer.

## TECHNICAL SKILLS

| Languages | Database Management | Data Science |
|---|---|---|
| C / C++ | MongoDB | MATLAB |
| Python | MySQL | TensorFlow |
| R | | R Studio |
| PHP | | |
| JavaScript | | |

## PROJECTS

**Cat and Dog Detection**                                                          MATLAB

*Utilize a deep learning network for image detection*
- A deep learning network that detects whether an image is a cat or dog.
- Built on AlexNet (Convolutional Neural Network).

**Smart Baby Car Seat**                                                               C

*Build a car seat to ensure the safety of the child..*
- Detects whether a child is left alone in a vehicle for a long period of time.
- Alerts caregiver and local authorities with GPS location of the child.
- Detects the temperature of the vehicle and state of the child.
- Disables alerts once the child is safe.

**Telemetric LED Strip**                                                    C / JavaScript

*Use multiple sensor to control a led strip based on readings.*
- Changes color based on the temperature.
- Can access sensor readings / data wirelessly.
- Patterns dictated by a touch switch

## EDUCATION

**Florida Atlantic University**                                                      2020
*Boca Raton, Florida*
- Bachelor of Science: Computer Science

# Daniel Leach

Software Developer

(954) 701-9338
dleach2018@fau.edu
LinkedIn
GitHub
Portfolio

## Objective

I am a Computer Science Major interested in developing new applications to help further research in Computer Science.

## Skills

C++, Python, JavaScript, Java, HTML5, CSS3, PHP, MySQL, GIT, Mobile App. Development

## Education

FLORIDA ATLANTIC UNIVERSITY
BOCA RATON, FL
B.S. IN COMPUTER SCIENCE
Expected Fall 2020

## Course Work

Foundation of Computer Science
Intro. to Logic Design
Stochastic Models for Computer Science
Data Structures and Algorithm Analysis
Intro. To Internet computing
Intro. to Microprocessor Systems
Design and Analysis of Algorithms
Introduction to Artificial Intelligence
Python Programming
Computer Operating Systems
Intro. To Database structures
Java Programming
Mobile App. Development
Data Science and Analytics
Principles of Software Engineering
Formal Languages and Auto. Theory

## Experience

### Maverick Pet Foods | Production Manager

Aug 2014 – OCT 2015
- Oversaw the production of all company products from beginning to end.
- Managed new employee training and work hours.
- Generated daily progress reports.

### Clean Air Services | HVAC Technician

March 2012 – Dec 2015
- Maintenance and cleaning of ventilation systems on marine vessels.
- Extensive knowledge of how ventilation systems operate.
- Installation of access points to assess the condition of certain areas of a ventilation system.
- Preparation of equipment for current and future jobs.

## Projects

### Remote Controlled Raspberry Pi Car

The objective of this project was to redesign an original project based on the SunFounder Smart Car for the Raspberry Pi. This project removes all Wi-Fi enabled capabilities and instead uses Bluetooth to be controlled by a PlayStation 3 DualShock controller. This was done because of the original design had major latency issues and poor design. By changing from Wi-Fi to Bluetooth I was able to reduce the latency to be almost unnoticeable.

### SchoolSafe

This project was developed as a team to introduce a more efficient and alternate solution to school related emergencies. The application allows administration and teachers to send an emergency to all active users and be able to communicate between them as well as mark students accounted. SchoolSafe uses the Bootstrap framework to cut down development time, Firebase to handle real time messaging, and Mobiscroll to handle calendar events.

## Accomplishments

### President of the Broward College Robotics Club

Spring 2018
During my time as president of the Robotics club, I was able to teach fellow students the basics of robotics. Held weekly meetings and workshops. I helped put on a robotics competition based on the autonomous robot that I had created.

# Claudio J. Herrera

Fort Lauderdale, FL · (786)606-6696 · cherrera@herreratech.com

**EDUCATION**

Florida Atlantic University                                           Boca Raton, FL
Bachelor of Science in Computer Science                  December 2020
GPA: 3.5/4.0

**RELEVANT COURSEWORK**

Intro to Programming, Data Structures, Intro to Internet Computing, Computer Operating Systems, Applied Database Systems

**TECHNICAL SKILLS**

- *Languages:* C/C++, SQL, Python
- *Software:* SQL Server, ORACLE, Web Development, Relational Databases
- *Operating Systems:* Windows, Linux/Unix
- *Hardware:* Hardware Troubleshooting, Servers, Networking, Cabling

**ACADEMIC PROJECTS**

**Voting App for Popular Baby Names** (Intro to Internet Computing, Spring 2019)
Designed a Web application that allows users to vote on their favorite baby names and displays updated statistics of the most popular baby names. The Front-end uses HTML, CSS, Bootstrap and JavaScript. The Back-end uses PHP and MySQL.

**EXPERIENCE**

Herrera Tech LLC                                                      Hollywood, FL
**Owner/Operator**                                              January 2017 – Present

- Worked closely with clients to solve IT issues. Determined changes, recommended quality software, projected modifications of software, hardware and networking.
- Provided 24/7 on call remote and on-site IT support for clients in various cities.
- Implemented solutions that provided value and increased efficiency for the client.

Centuric Cloud Computing                                     Fort Lauderdale, FL
**Cloud Specialist**                                           April 2012 – January 2017

- Supervised, operated, programmed and maintained Asterisk based PBX systems ensuring minimum outages and down time.
- Managed user accounts using Active Directory 2003/2008 to create new domain accounts, troubleshoot user account lockouts and provide controlled access to existing network shares.
- Deployed and maintained several VMs in a VMware vSphere Infrastructure.

**LANGUAGES**

Fluent in the written and oral communications of English and Spanish

# Preston Small

Ft. Lauderdale, FL 33312

(954) 993-5158

psmall2018@fau.edu

---

**EDUCATION**

**FLORIDA ATLANTIC UNIVERSITY**         Boca Raton, FL

Bachelor of Science in Computer Science         May 2018-present

**Relevant Courses:**

| | |
|---|---|
| Design and Analysis of Algorithms | Principles of Software Eng. |
| Introduction to Internet Computing | Data Structures and Algorithms |
| Deep Learning | Intro to Database Structures |

**SKILLS**

**Languages:** C++, C, Python, HTML, JavaScript, PHP, SQL

**Operating Systems:** Unix, Windows, Linux

**Software:** Visual Studios, CLion, Eclipse

**PROJECTS/LABS**   **Stock Trading Simulation**

Created a simulated stock trading program that read a user's stock shares with prices from a file. The program would help user decide when to sell stocks for a profit.

**Logic Design Lab**

Analyzed and tested combinational and sequential circuits, while using the fundamentals of Logic Design and Boolean Algebra.

**Internet Computing**

Created various web projects using HTML, CSS, JavaScript, and PHP.

**EXPERIENCE**   **Juice Bar & Cafe**         Ft. Lauderdale, FL

Kitchen Leader         June 2019 - Present

- Use communication skills and adaptability to train new employees.
- Delegate tasks and work as part of a team to ensure productivity is maintained.
- Taking inventory and placing orders for necessary items in the kitchen.

# Brandon Deringer

Plantation, FL
(954) 909-3016
brandonderinger@gmail.com

## EXPERIENCE

**iFixandRepair,** Fort Lauderdale, FL — *Repair Technician*

July 2016 - August 2017

- iPhone, Android, Tablet, iPad, Laptop, Mac, and PC repairs.
- Packaging and shipping of large quantities of valuable electronics. Inventory management. Laboratory upkeep. Communication with senior staff.

**Quarterdeck,** Fort Lauderdale, FL — *Food Runner/Busser*

May 2016 - July 2016

## EDUCATION

**Florida Atlantic University,** Boca Raton, FL — *Bachelor of Science in Computer Science*

Expected Spring 2021

**Broward College,** Davie, FL — *Associate of Arts*

May 2018

## PROJECTS

### SchoolSafe

January 2019 - April 2019

Web-based application developed to provide a tool for handling school emergencies. It allows faculty to send an alert to all users and provides a channel for communication via messaging.

## SKILLS

C/C++

Python

HTML5/CSS

JavaScript

## RELEVANT COURSEWORK

Intro to Programming

Data Structures & Algorithms

Intro to Internet Computing

Intro to Database Structures

Intro to Deep Learning

Computer Operating Systems

Formal Languages and Automata Theory

# SchoolSafe:

## A Web Application for Increasing Safety in Schools During Emergencies

Johnny Northrup

Daniel Leach

Alan Tessier

Claudio Herrera

Preston Small

Brandon Deringer

**Abstract**

Within the past few decades school shootings have been on a steady rise in the U.S. These tragic occurrences cause fear in schools, where teachers and students should otherwise feel safe and able to foster learning and growth. With security measures expanding in schools and technology becoming more integral in the classroom, we present our application SchoolSafe. SchoolSafe is a web application that allows emergency alerts and live messaging between educators in the classroom and other faculty about active emergency situations on the campus. We will analyze previous security applications aimed at schools and present our software solution to this problem. Further, we will explore the future of safety applications and examine the effectiveness and ethical implications involved with these proposed solutions.

**1. Introduction**

As school shootings have become more commonplace over the recent decades in the United States, there has been a demand to find immediate solutions to prevent these tragedies from occurring. Various security measures have been implemented in schools to help mitigate the severity of these shootings such as assigning School Resource Officers (SROs), access control measures (locking doors, screening visitors, requiring identification badges), and metal detectors (Jonson). But even with these measures put in place, the effectiveness of these solutions is still not certain as these tragedies continue to occur. With the rise of technology in the classroom, as well as wide access to smartphones and the internet, technology can play an important role in securing the classroom and providing a safe environment for students and teachers. Our team at SchoolSafe believes that technological solutions can play a pivotal role in

communication during these crisis scenarios. We will explore previous attempts at school safety

applications and discuss how we intend to create a better solution. We will discuss the features of

our application as well as the development, architecture, and security measures used. Finally, we

will plan our future aspirations for SchoolSafe, and see what the future holds for safety

applications in the educational domain.


## 2. Background

To make the best solution possible, we first researched some existing applications that

offer similar services. VoiceShot is a system that allows users to alert people via phone call or

text message (Ikegwu et al.). However, this would require a separate solution for first reporting

incidents, whereas in our application, reporting an emergency is directly tied to the alert system.

This will cut out the need for a second application and improve response times.

The University of São Paulo developed a mobile app that was most similar to what we

envisioned for our application (Ferreira et al.). It allows students, faculty, and staff to report

emergencies. The report would then be reviewed by the university's security team, and a

decision would be made on how to proceed. Because of the grim reality that school shooters are

often students (Jonson), we do not allow students to access the application. This will reduce false

alarms and prevent students with nefarious intentions from using the application to manipulate

other users. Instead of only alerting security, our application will alert all faculty and parents in

the event of an emergency, allowing immediate action and facilitating more effective

communication through the messaging feature.

**3. Our Application School Safe**

SchoolSafe is an easy to use, responsive application that can be used on any desktop computer as well as any smartphone. We decided developing a web application would be a feasible solution to the problem we are addressing.Based on the literature reviewed, we determined what features we wanted to include in our application.

In an emergency situation it is important to alert the faculty of the school as quickly as possible so that students and staff can take necessary precautions. The most important feature of the SchoolSafe App is the alert system, which notifies teachers, parents, and other faculty of emergency events taking place as soon as they occur. For example, if a shooting were to take place, the teacher within seconds can press the emergency button and select what kind of emergency is happening from a list of options. The faculty and parents with the app will be notified immediately and be able to quickly take action.

Another core functionality of the application is the messaging feature, which provides instant communication between teachers, faculty, and parents. Because communication is crucial during emergencies, live messaging will be a vital tool in providing real-time updates to other classrooms, SROs (student resource officers), and parents. Rapid communication between different classrooms can aid in quickly determining a course of action.

Keeping track of students during an emergency is an important task. Students who may not be present on a given day need to be accounted for, as well as students who might be elsewhere in the school. We included an attendance feature which allows teachers to easily access their student roster and update the attendance which will be stored in the database. This feature will be a useful tool during fire drills and also throughout ordinary daily activities.

We've included a monthly calendar as an additional feature that would include important district school events. This feature is to keep teachers and parents informed on important dates and events.

Administrators are able to assign roles to users to determine which features they can access. The lowest user level is for parents, who will be allowed to view emergency alerts and the calendar. Teachers will be able to send emergency alerts, take attendance, and send messages. Administrators are at the highest level; they will have the ability to change the database and dismiss alerts. By restricting lower level accounts we can prevent features of the app from being abused. False alarms will be less likely to occur and because only the admin can dismiss alerts, they will remain active as long as there is an ongoing emergency.

Because of the time constraints and size of our team, we used an agile incremental approach for our developmental model. Plan-driven approaches, such as the waterfall method, require a lot of time to be dedicated to planning and requirement engineering. We felt that time would be better used toward iterating our application and implementing new features. Also, strict documentation would make it difficult to adapt to change. Because our team was working remotely, we found it easier to delegate tasks to certain members, but at the same time have the flexibility for team members to switch between tasks as needed. Each member having different strengths and backgrounds made extreme programming practices, such as pair programming, useful for debugging and testing.

SchoolSafe was designed using a model-view-controller (MVC) architectural pattern. The MVC pattern allows the implementation and development of the application to be modular. For example, the model part of the application handles the database. In our case, we chose to

implement our database using Oracle because of its robustness and reliability. The view in the MVC pattern refers to the front-end design of the application. SchoolSafe uses HTML, CSS, and JavaScript to display all of its data to the user. The controller was written using PHP and was used as the interface between all the client-side operations with the server side and database operations. PHP was chosen for its scalability and performance (Adam and Andolo). It was used to handle things such as input validation, password hashing, error handling, database querying, and to dynamically populate the user interface.

The calendar in this project uses the Mobiscroll API, which is responsible for syncing the Google calendar onto the display. The API first sets up and designs a dynamic calendar using HTML, CSS, and JavaScript and retrieves the user's Google calendar events once the user signs in. A separate API key must be generated by Google calendar for the user to log into their Google account and sync any holidays or events on their Google Calendar onto the Mobiscroll calendar. The messaging system utilizes Firebase because it is a real time database, which is helpful since the user can immediately send, retrieve, and delete their messages in real time. The database grabs the time the message is sent, the session username, and the message that the user wants to send. As soon as the user presses send, the data gets appended to the database, added to a message (list item) on the screen and visible to all users on the message board. When the user decides to delete a message, the message will no longer be visible to users or anyone on the message board.

Security at SchoolSafe is one of our top priorities. Our team is devoted to not only keeping students safe, but also to ensuring the security of our users' data. We have taken several

measures to ensure that their data will not fall into the wrong hands and potentially pose a threat to their privacy or become victims of stolen identity. The following are just a few of the methods that we have implemented to ensure the security of all the users on our platform: We do not store any sensitive information in plain text. All passwords are salted and hashed using the SHA-256 hash function. The SHA-2 family of hash functions is one of the Secure Hash Algorithms required by law for use in certain US government applications (Dang 6). In addition, our engineers have implemented input validation functions which prevent SQL injection attacks. Along with input validation, the implementation of error handling allows the user to receive instant feedback should there be any errors while using SchoolSafe. Lastly, our user level feature protects users data, and prevents lower level users from accessing unnecessary features.

**4. Future**

In the future, we hope to build upon the functionality of the SchoolSafe App to enhance the user experience. A few features we would like to include in our application would be push notifications to update users of current emergencies and events, and a location tracker to aid in decision making of whether to lock-down or evacuate during an emergency.

While there are existing applications in the safety domain that have similar functionality to our app, there is innovative research and development being done in this area. For example, researchers and educators from the University of Nevada, Las Vegas and surrounding high schools have developed software that can increase security levels in school by using computer vision and machine learning to detect intruders and monitor the emotional state of students (Deniz et al.). Another proposed solution is to have robotic mounts installed in schools that could

be capable of detecting sudden movements of large groups of students, and could be used to neutralize an intruder by firing off an electrically charged dart (Ikegwu et al.). Although these solutions are hypothetical, they may prove to be useful as areas of machine learning and computer vision forge ahead.

However, with this rapid progress in technological capabilities, it is important to consider the ethical implications of these technologies. According to an analysis of 45 essays written by senior year computer engineering students about the societal impacts of computer vision technology, 48.9% of the students discussed concerns about privacy/anonymity (Cote and Albu).


## 5. Conclusion

Emergency situations in schools, particularly school shootings, remain an ongoing crisis and there is no empirical evidence to show that certain security measures are proven to stop school shootings (Ojasalo et al.). However, it is our hope that the SchoolSafe application will be a useful tool to teachers, faculty, and parents in reacting promptly to emergencies. The role of technology in the classroom is not only integral to learning, but can be utilized in security and safety measures. The outcry after school shootings shows the demand for a solution; while there is no clear solution to this problem, we believe that SchoolSafe will be a mechanism for faculty and staff to handle these unfortunate incidents more effectively, mitigating the risks involved in these scenarios.

Works Cited


Adam, Stenly Ibrahim and Andolo, Stevani. "A New PHP Web Application Development

      Framework Based on MVC Architectural Pattern and Ajax Technology," 2019 1st

      International Conference on Cybernetics and Intelligent System (ICORIS), Denpasar,

      Bali, Indonesia, 2019, pp. 45-50.

Cote, Melissa and Albu, Alexandra Branzan. "Teaching Computer Vision and Its Societal

      Effects: A Look at Privacy and Security Issues from the Students' Perspective," 2017

      IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW),

      Honolulu, HI, 2017, pp. 1378-1386.

Dang, Quynh. "Recommendations for Applications Using Approved Hash Algorithms." NIST,

      Special Publication 800-107, Revision 1, Aug. 2012 p.6.

Deniz, Sarah, et al. "Computer Vision for Attendance and Emotion Analysis in School Settings,"

      2019 IEEE 9th Annual Computing and Communication Workshop and Conference

      (CCWC), Las Vegas, NV, USA, 2019, pp. 0134-0139.

Ferreira, João Eduardo, et al. "Smart services: A case study on smarter   public safety by a

      mobile app for University of São Paulo," 2017 IEEE SmartWorld, Ubiquitous

      Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing &

      Communications, Cloud & Big Data Computing, Internet of People and Smart City

      Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA,

      2017, pp. 1-5.

Ikegwu, Kelechi M., et al. "Exploring technological preventive methods for school shootings,"

SoutheastCon 2015, Fort Lauderdale, FL, 2015, pp. 1-6.

Jonson, Cheryl Lero. "Preventing School Shootings: The Effectiveness of Safety Measures."

*Victims & Offenders*, vol. 12, no. 6, Nov. 2017, pp. 956–973. *EBSCOhost*,

doi:10.1080/15564886.2017.1307293.

Ojasalo, Jukka, et al. "Responsibility and decision making transfer in public safety and security

emergencies - A case study of school shootings," *2009 IEEE Conference on Technologies

for Homeland Security*, Boston, MA, 2009, pp. 358-365.