

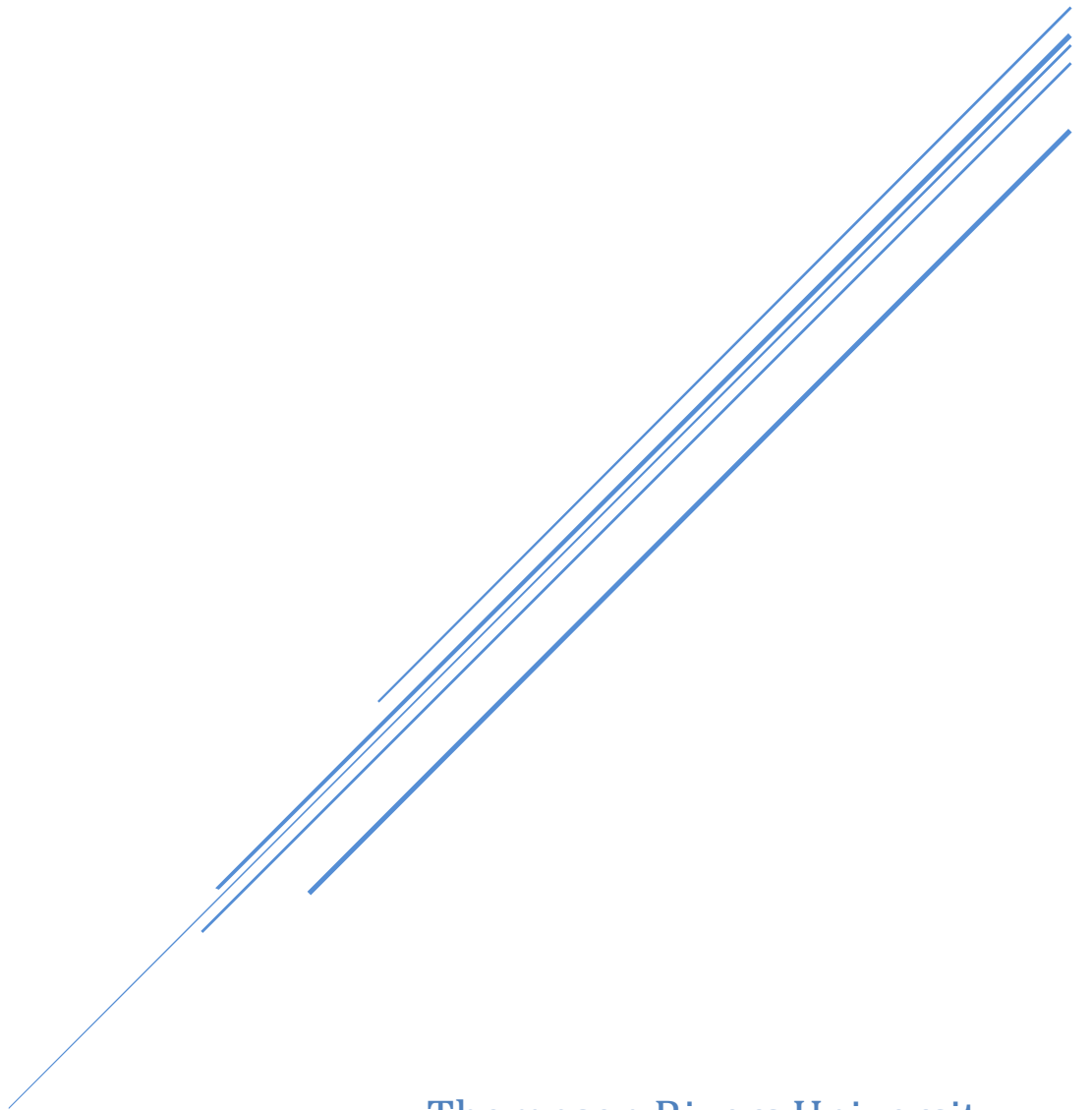
# FINAL PROJECT REPORT - SUPERFLY SWEARJAR

Group 9:

Kenneth Kouadio (T00633691)

Iskander Sermanizov (T00645991)

Mashruf Ahmed Zafri (T00657643)



Thompson Rivers University  
COMP-4910

## Table of Contents

Problem description.....	1
Company .....	1
People .....	2
Problem .....	2
Requirements .....	3
Functional .....	3
Nonfunctional .....	4
Analysis .....	5
Design .....	5
Chrome Extension .....	5
CI/CD Pipeline Integration.....	6
Firebase Backend .....	6
Implementation.....	6
Key Features .....	7
Technologies Used .....	7
Database .....	8
Chrome Extension Development.....	9
Testing .....	9
Branch Management.....	9
Appendixes .....	9
Weekly reports .....	9
Presentations .....	44

## Problem description

### Company

SuperFly Aerial has been a drone service company since 2015, and later expanded to engineering and industry. Key highlights:

- **Custom Thermographic Sensor:** Developed with specialized software, this sensor integrates real-time thermographic imagery with IMU data for precise 3D temperature location.
- **Target Market:** Focus on efficient thermographic inspections for the engineering and industry sectors.
- **Comprehensive Solution:** Provides end-to-end services, showcasing innovation in the drone industry.

## People

- Kenneth Kouadio
- Iskander Sermanizov
- Mashruf Ahmed Zafri

## Problem

- **Identifying Office Challenges:** SuperFly Aerial recognizes a need for addressing issues like profanity, late arrivals, and forgetting to unmute during meetings.
- **Project Objective:** Develop an integrated app or chatbot to track and manage these common office fouls efficiently.
- **Public Domain Access:** The goal is to create a freely available tool accessible to everyone, promoting a collaborative and positive work environment.
- **Streamlining Accountability:** The project aims to enhance workplace etiquette by providing a user-friendly solution for tracking and improving behavior in meetings.
- **Innovation for All:** SuperFly Aerial's commitment to fostering a productive and respectful workplace through accessible technology.

# Requirements

Our client provided us with the following requirements:

## Functional

### 1. Team Management

The system should allow users to **create, edit, and delete teams**. Each team should have the following attributes:

- **ID:** A unique identifier for the team.
- **Name:** Descriptive name for the team (e.g., “SuperFly Aerial,” “Swearjar Project Team,” “The Canadian Federal Government”).
- **Description:** Brief information about the team’s purpose or focus.

### 2. User Management

Users should be able to create, edit, and delete user profiles. Each user profile should include:

- **ID:** A unique identifier for the user.
- **Name:** User’s name.
- **Email Address:** Contact information for the user.

### 3. Category Management

The system should support the creation, editing, and deletion of **categories**. Categories represent different types of violations or incidents. Each category should have:

- **ID:** A unique identifier for the category.

- **Name:** Descriptive name for the violation category (e.g., “Number of times forgot to turn off mute”).
- **Description:** Additional details about the category.

#### 4. Multi-Tenant Functionality

The system must support **multiple teams** without data leakage between them. Each team’s data should remain isolated from others.

#### 5. Dashboard and Statistics

The system should provide a **dashboard** displaying team statistics over different time periods (30, 90, and 365 days). Implementations for stat display:

- **Chart:** Visual representation of team performance.
- **Leaderboard:** Ranking of users based on violations.

#### 6. Violation Logging

Users should be able to log violations for various categories through a simple interface. A straightforward mechanism (e.g., a big button) should allow users to **add violations** for specific users (e.g., “Andrew”) within a chosen category (e.g., “Forgot to turn off mute”).

### Nonfunctional

#### 1. Configurability

The system should be **highly configurable** to adapt to various scenarios and use cases.

#### 2. Self-Serve/Open Design

- The system should be designed to be as **self-serve/open** as possible.

- Personal information about existing teams must be **protected**.

### 3. Affordability and Safety

The system should be **cost-effective** (cheap) and prioritize user safety and privacy.

## Analysis

The problem statement identifies a gap in the market for a solution that can monitor and report on common office fouls such as profanity, forgetting to turn off mute and showing up late to meetings. These fouls can affect the productivity, communication, and morale of the team, and therefore need to be addressed. The problem statement also specifies that the solution should be easy to use, persistent, configurable, user-specific, and simple.

Our proposed solution would increase the awareness and accountability of the team members and encourage them to avoid or reduce office fouls. The solution would also be adaptable to different types of teams and workflows, as it would allow users to choose what they want to track and how they want to report it.

## Design

### Chrome Extension

We decided to create a standalone Chrome extension that enhances accessibility for SwearJar users by making it available on different platforms. This extension streamlines interactions, improves user experience, and provides essential features. We will publish our extension on the Chrome Web Store.

## CI/CD Pipeline Integration

To ensure seamless deployment of new features, we integrated a Continuous Integration/Continuous Deployment (CI/CD) pipeline. This automated process facilitates testing, building, and deploying updates.

## Firebase Backend

We have chosen Firebase for its cost-effectiveness and scalability. Here is why:

- **Database Storage:** Firebase offers 1GB of DB storage, accommodating our data needs efficiently.
- **Blob Storage:** With 5GB of Blob storage, we can securely store files, images, and other media.
- **Affordability:** Firebase's pricing aligns with our budget constraints.

## Implementation

SuperFly Aerial has identified a gap in the market for a comprehensive solution that monitors and reports common office violations, such as profanity usage, forgetting to turn off mute, and tardiness to meetings. Our project aims to fill this void by developing an integrated app that facilitates the tracking of these critical stats.

These fouls can affect the productivity, communication, and morale of the team, and therefore need to be addressed. Our proposed solution would increase the awareness and accountability of the team members and encourage them to avoid or reduce office fouls. The solution would also be adaptable to different types of teams and workflows, as it would allow users to choose what they want to track and how they want to report it.

## Key Features

- **Configurability:** Users can easily create, edit, and delete teams, users, and categories to tailor the app to their specific needs.
- **Multi-Tenant Functionality:** The app supports multiple teams while ensuring data privacy and security between them.
- **User-Friendly Dashboard:** A visually appealing dashboard displays team statistics over customizable timeframes, allowing for easy monitoring of office behaviors.
- **Violation Logging:** Users can log violations with a simple interface, promoting transparency and accountability.
- **Chart and Leaderboard Display:** Statistics are presented through interactive charts and leaderboards, providing insights into team performance.
- **Self-Serve/Open Access:** The system is designed to be accessible to all users while safeguarding personal information and maintaining affordability.

## Technologies Used

- **Development:** The project was developed using modern web technologies and tested locally before being deployed to the GitHub repository.
- **Frontend:** React.js and Next.js were utilized for building the user interface, providing a responsive and interactive experience while maintaining a single codebase with both backend and frontend components. Material-UI (MUI) was used for UI design and components.
- **Backend:** Connecting to Firebase Database (NoSQL) for secure data storage and management. Firebase offers larger storage space compared to other free services.



- **Deployment:** The project is deployed using Netlify for hosting the application, providing seamless deployment and scalability. Git + GitHub were used for collaborative development and version control.
- **CI/CD:** GitHub Actions were implemented for continuous integration and continuous deployment (CI/CD), ensuring automated testing and deployment processes. Unit tests were utilized to maintain code quality and reliability.
- **Other Tools:** ChatGPT was utilized for generating default violation types. The Agile methodology was adopted for the software development process, promoting iterative and incremental development.

## Database

SwearJar database consists of four tables: Users, Teams, Violation Type, and Report.

The Users table stores information about employees, including their ID, first and last names, email, profile picture, and the teams with which they are associated.

The Teams table contains details about different teams, including their ID, name, and team members.

The Violation Type table categorizes types of violations by ID, name, and a text description of the violation type.

The Report table logs instances of violations. Each record includes the report ID; IDs of the reporter and violator (employees); team ID and the specific type of violation; as well as a timestamp indicating when the violation was reported.

## Chrome Extension Development

To configure chrome extensions, the manifest.json file is used to set things like permissions, the name, the version and much more. To find out how you can further configure the extension check out the official documentation.

## Testing

GitHub Actions, Git and Jest were implemented for continuous integration and continuous deployment (CI/CD), ensuring automated testing and deployment processes. This continuous integration tool automatically runs our test suite whenever we push changes to our repository.

Unit tests were utilized to maintain code quality and reliability. We have chosen Jest as our testing framework because it is robust, reliable, and tailored for JavaScript. With Jest, we were able to write unit tests and reached the 50% testing threshold.

## Branch Management

We follow a clear branching strategy:

- **Main Branch:** Our production-ready code is stored here. It is stable and tested.
- **Feature (Dev) Branches:** Each feature or bug fix received its own branch. We developed, tested, and refined code here. Once ready, we merged into main.

## Appendixes

Weekly reports

## **Week 3: Weekly Report for COMP 4910 Project Course**

### **Summary:**

Our team is developing an integrated solution for tracking harmless office fouls like profanity, forgetting to turn off mute, etc.

Summary of last week (Jan 8th - January 14th) of the team:

Jan 9th - Group 9 was formed. Members met to select a project.

Jan 10th - Group met together for a basic overview.

Jan 11th - Group decided on software architecture and blueprint for the project.

Jan 12th - A basic outline with a database and basic UI was set.

Jan 14th - Group had a video meeting with Andrew Challenger and was given a basic understanding of the project. He received an overview of what the group has understood and decided for the project. The group received feedback and a point of start for the project.

We decided these so far:

A standalone program/chrome extension with user-friendly UI that will keep track of harmless workplace fouls from employees. A database will store this information and monthly/weekly send our reports. Mostly tracking reports will be done manually by admins with a few mouse clicks.

We will use Firebase for services like database (NoSQL) to store information, authentication, etc. We will use Node.js for the backend of the application. Vercel will be used for deployment. GitHub Desktop will be used for the project repository.

### **Task Completion from Last Week**

Need to set up the database with types of things to keep track of, dummy employees, test cases etc.

Finished a simple UI using Node.js.

Met with the client.

### **Task for This Week**

1. Develop the first prototype.

### **Problem**

It has come to SuperFly Aerial's attention that there are no good integrated solutions for tracking profanity, forgetting to turn off mute, showing up late to meetings and other harmless office fouls.

This project aims to correct this oversight, through the creation of some kind of integrated app or chatbot that can be used for tracking these critical stats. This app would be public domain, freely available for all to use.

## Requirements

### Functional

- Persistent tracking of stats overtime in a database.
- User configurable types of statistics to collect.
- Association of statistics with individual users (ideally without manual configuration).
- Simple reporting that would give updates on how all the users in the server are performing.

### Non-Functional

- Easy to use, without disrupting workflow or requiring more than the minimal amount of human effort.

## Analysis

The problem statement identifies a gap in the market for a solution that can monitor and report on common office fouls such as profanity, forgetting to turn off mute and showing up late to meetings. These fouls can affect the productivity, communication, and morale of the team, and therefore need to be addressed. The problem statement also specifies that the solution should be easy to use, persistent, configurable, user-specific, and simple.

Our proposed solution would increase the awareness and accountability of the team members and encourage them to avoid or reduce office fouls. The solution would also be adaptable to different types of teams and workflows, as it would allow users to choose what they want to track and how they want to report it.

## Design

1. Development: Writing code, implementing features and testing will be done locally and updated on to GitHub Desktop repository. Firebase, Node.js, and Vercel will be used to achieve the desired result.
2. Database: Node.js to create backend logic, connecting to Firebase services to store data.
3. Deployment: GitHub repository will connect to Vercel and push it for deployment.

4. User Interaction: Simple UI. Click user, click type of foul, click “+” to add to counter.
5. Real-Time data: Provide instant updates to admin.

#### **Week 4: Weekly Report for COMP 4910 Project Course**

##### **Summary:**

Our team has developed a basic outline of the project. We developed three components of the UI and have set up the database on Firebase. A chrome extension, though not linked yet, has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

Summary of last week (Jan 15th - January 21st) of the team:

Jan 15th - Developed certain component UI like user list, violation list, report, and report button.

Jan 16th - Dummy Chrome extension for SwearJar created. Data integration and testing remains.

Jan 17th - Database has been set up in Firebase.

Jan 18th - UI components slightly polished and pushed to GitHub repo.

Jan 19th - First meeting with Kevin

Jan 21st - Second meeting with Andrew

##### **New decisions/plans:**

The project will be a chrome extension rather than a standalone application. Reporting shall be kept manual for now.

The MVP: UI, logic, database, and deployment.

Would be nice: Razzle-dazzle.

Stretch: AI integration.

##### **Task Completion from Last Week**

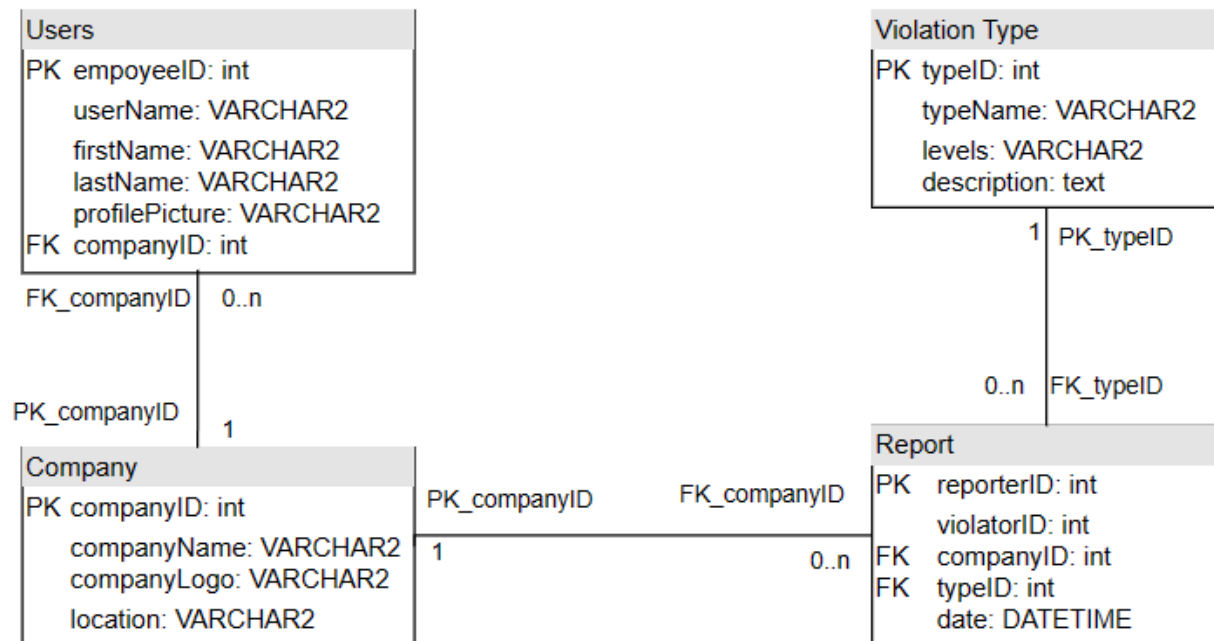
Finished setting up a database, a chrome extension, and some UI elements.

Had a second meeting with Mr. Challenger.

##### **Task for This Week**

1. Work on the logic.
2. Database needs to be populated with dummy data.
3. Decide on a few UI elements.
4. Merging the components and performing a few test runs.
5. Develop the first prototype.

UML diagram for Database:



The above UML diagram visualizes database design for a system that tracks and reports violations within a company. It consists of four tables: Users, Company, Violation Type, and Report.

The Users table stores information about employees, including their ID, username, first and last names, profile picture, and the company with which they are associated.

The Company table contains details about different companies, including their ID, name, logo, and location.

The Violation Type table categorizes types of violations by ID, name, levels of severity or type (as indicated by the 'levels' field), and a text description of the violation type.

The Report table logs instances of violations. Each record includes the report ID; IDs of the reporter and violator (employees); company ID and the specific type of violation; as well as a timestamp indicating when the violation was reported.

## Week 5: Weekly Report for COMP 4910 Project Course

### Summary:

Our team has developed a basic outline of the project. We developed three components of the UI and have set up the database on Firebase. A chrome extension, though not linked yet, has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

Summary of last week (Jan 22nd - January 28th) of the team:

Jan 22nd - continued developing component UI like user list, violation list, report, and report button.

Jan 23rd - first merge

Jan 24th - database tested.

Jan 25th - UI components polished.

Jan 26th - second meeting with Kevin

Jan 28th - third meeting with Andrew. Received requirements list from Andrew.

### **Task Completion from Last Week**

Finished setting up a database, a chrome extension, and some UI elements.

### **Task for This Week**

1. Finish the logic.
2. Test database.
3. Set up authentication.
4. Individual user details

### **Requirements**

Requirements for the SwearJar project:

The system should:

Be as configurable as reasonably possible to allow for application to various situations.

The user should be able to create/edit/delete the following:

- Teams (ex: SuperFly Aerial, Swearjar Project Team, The Canadian Federal Government). Should have ID, name, description.
- Users (ex: Andrew) Should have ID, name, email address.
- Categories (ex: Number of times forgot to turn off mute) Should have ID, name, description.

Have rudimentary multi-tenant functionality and administration.

- Multiple Teams should be supported by the app without information leaking between them.
- Recommend making it as self-serve/open as possible, while protecting the personal information of existing teams.

Have a dashboard showing team statistics over the past 30, 90, 365 days, with a simple interface for logging violations of the various categories for every user on the team.

- Possible implementation for stat display would be a chart or leaderboard.
- Possible implementation for adding a violation would be a big button that adds +1 to user "Andrew" in the "Forgot to turn off mute" category.

Be designed with the intention of making the system a free service open to all users. Cheap and safe please!

## UI Components

UI Framework prototype (version 1):



Violation List:



Select the Violation

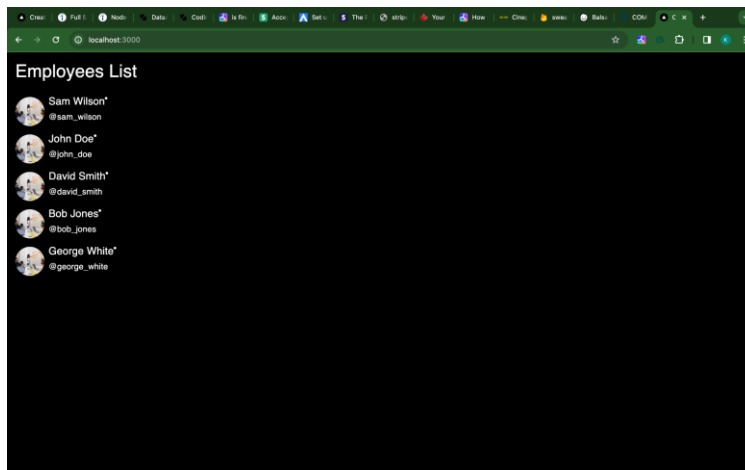
- ☐ Profanity
- ☐ Muted Microphone
- ☐ Late Arrival

Add new Swear Type

SUBMIT

REPORT

## Users List



## Week 6: Weekly Report for COMP 4910 Project Course

### Summary:

Our team continued development of a basic prototype. We developed three components of the UI and have set up the database on Firebase. A chrome extension,

though not linked yet, has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

Summary of last week (Jan 29th - Feb 4th) of the team:

Jan 29th - planned the changes needed to be done according to requirements.

Jan 30th - implemented few functional requirements.

Jan 31st - user authentication added.

Feb 1st - team viewer and nav bar added.

Feb 2nd - third meeting with Kevin.

Feb 3rd - bug fixes.

Feb 4th - fourth meeting with Andrew and decided on a few UI themes.

### **New decisions/plans:**

Apply a few UI components and styling. Fix layout. Produce (at least) a logo mock-up.

### **Task Completion from Last Week**

Finished correcting layout, program logic and user authentication. Integrated and made the database functional. Added new UI components: team viewer and navbar.

### **Task for This Week**

1. Decide UI theme and styling.
2. Fix bugs.
3. Polish individual UI components
4. Produce logo.
5. Develop user personas.

### **Additional Information**

Reminder to create a readme file.





We need to decide what should be on the Homepage Leaderboard view.

Navigation Bar added:

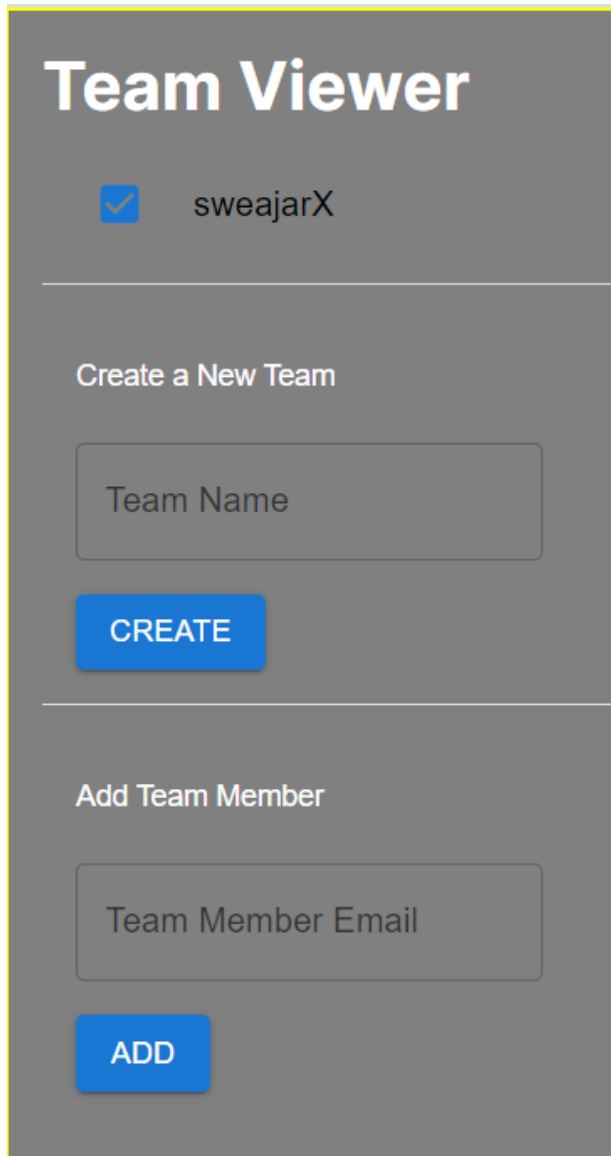
[Leaderboard](#)[Teams](#)[New Report](#)[Profile](#)[Logout](#)

## Homepage Leaderboard

SORT

	Bobby Boy	Profanity	20
	Sobby Boy	Profanity	20
	Lobby Boy	Profanity	20
	Ken K	Forgot to unmute	1000

Team Viewer:

A UI mockup for a 'Team Viewer' application. The title 'Team Viewer' is at the top in white on a dark grey background. Below it, a blue checkmark icon is next to the text 'sweajarX'. A horizontal line separates this from the 'Create a New Team' section. This section has a text input field labeled 'Team Name' and a blue 'CREATE' button. Another horizontal line separates this from the 'Add Team Member' section. This section has a text input field labeled 'Team Member Email' and a blue 'ADD' button.

# Team Viewer

☒ sweajarX

---

Create a New Team

Team Name

CREATE

---

Add Team Member

Team Member Email

ADD

## Week 7: Weekly Report for COMP 4910 Project Course

### Summary:

Our team continued development of a basic prototype. We developed three components of the UI and have set up the database on Firebase. A chrome extension, though not linked yet, has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

### Summary of last week (Feb 5th - Feb 11th) of the team:

Feb 5th - Produced a logo mockup. Developed user personas.

Feb 6th - Team viewer and Report page redesign.

Feb 7th - restructured the database.

Feb 8th - Added styles.js and color.js files.

Feb 9th - Added Leaderboard charts. Fourth meeting with Kevin. Redux for state management and caching

Feb 10th - team meeting, updated Leaderboard charts, CI/CD pipeline with GitHub actions, basic unit test integrated with CI/CD pipeline.

Feb 11th - fifth meeting with Andrew.

### **New decisions/plans:**

- Begin to write unit test for classes.
- Leaderboard and user details UI update

### **Task Completion from Last Week.**

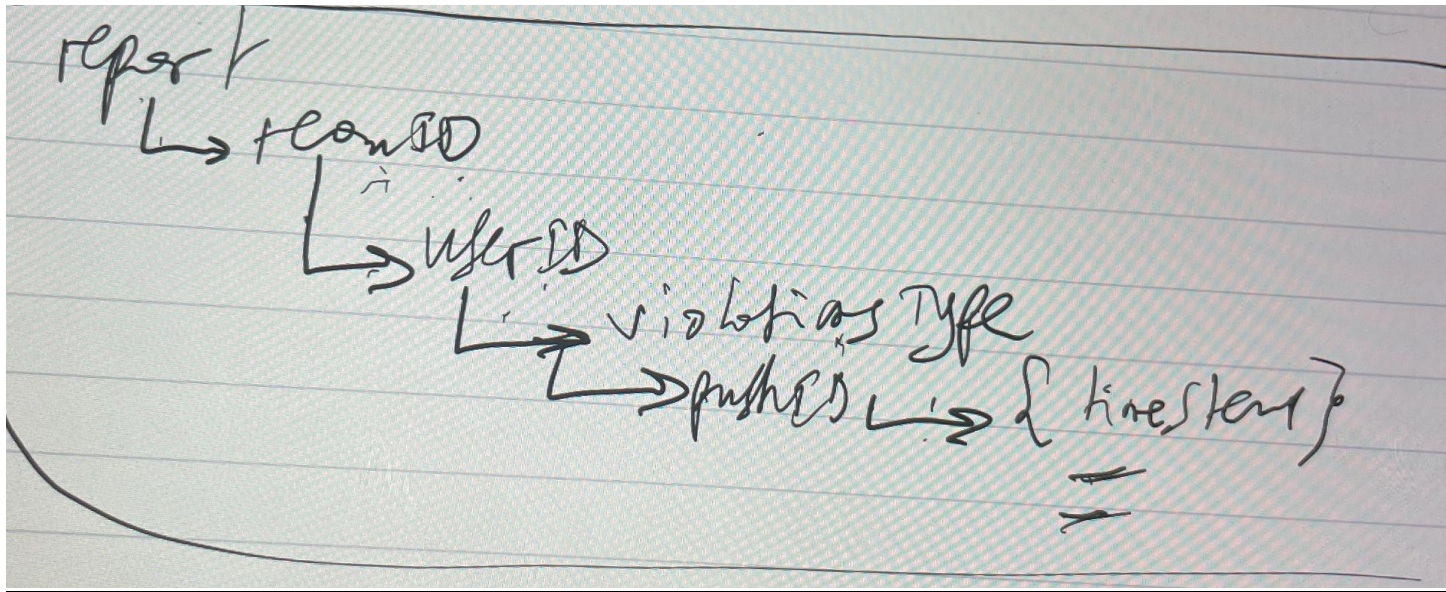


1. Produced logo mock.
2. Developed user personas.
3. Decided UI theme and styling.
4. Fixed bugs.

### **Task for This Week**

1. Update styles and color files.
2. Fix bugs - Ensure that the new database design does not affect team (create team and add team member) and new report (Creating new swear type)
3. Fix SORT button.
4. User stories (using Balsamiq)
5. Flow diagrams.
6. Configure the extension.
7. Add team Details component.
8. Fix the user Details component.

### **Report Database Structure Update**



Sign In / Sign Up:

Leaderboard Teams New Report Profile Logout

## Welcome to SwearJar! 🎉

Are you tired of your foul-mouthed habits? 🤔🤔

Well, fear not! SwearJar is here to help you keep those profanities in check. 😊😊

SIGN UP NOW SIGN IN

Leaderboard Teams New Report Profile Logout

Email

Password

SIGN IN BACK

[Leaderboard](#) [Teams](#) [New Report](#) [Profile](#) [Logout](#)

First Name

Last Name

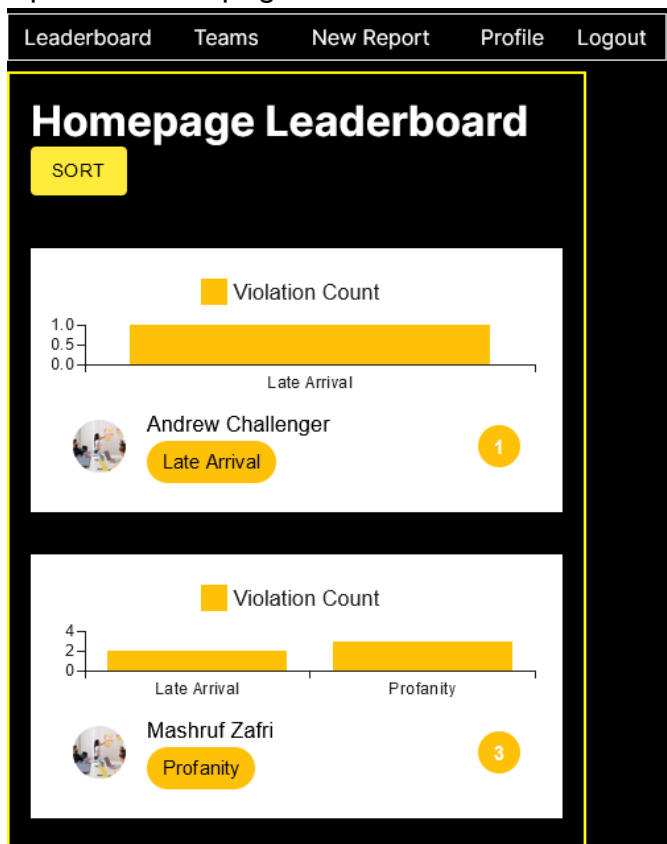
Email

Password

SIGN UP

BACK

### Updated Homepage Leaderboard:



New report page:

## New Report

Select the Violation

☐ Profanity

☐ Muted Microphone

☐ Late Arrival

**Next**

---

Add new Swear Type

**SUBMIT**

Team Viewer:

## Team Viewer

Your Teams:

☒ sweajarX

---

Create a New Team

**CREATE**

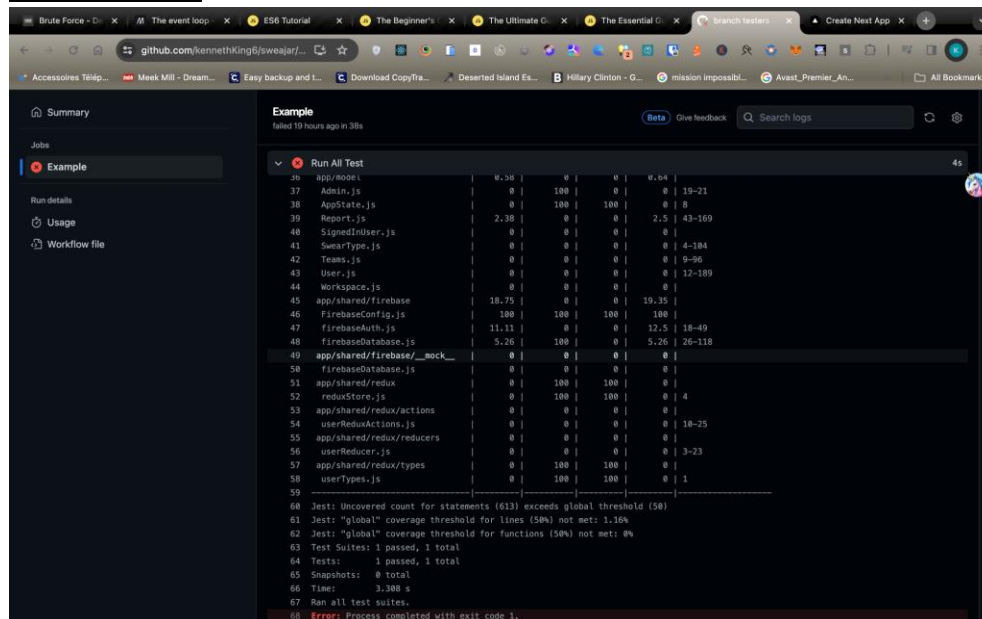
---

Add Team Member

**ADD**



## CI/CD Report



## Week 8: Weekly Report for COMP 4910 Project Course

### Summary:

Our team continued development of a basic prototype. We refined several UI components and have set up the database on Firebase. A chrome extension has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

### Summary of two last weeks (Feb 12th - Feb 25th) of the team:

Feb 12th - Added team Details component.

Feb 13th - Updated styles and color files.

Feb 14th - Updated TeamViewer, Team Details and New Report pages with icon buttons that allow users to switch between different components.

Feb 15th -

Feb 16th - fifth meeting with Kevin. Scheduled the Midterm review.

Feb 17th -

Feb 18th - sixth meeting with Andrew.

Feb 19th - Added test Line Chart to User Details.

Feb 20th -

Feb 21st - restructured the database.

Feb 22nd - Added styles.js and color.js files.

Feb 23rd - Added Leaderboard charts. Redux for state management and caching

Feb 24th - updated Leaderboard charts, CI/CD pipeline with GitHub actions, basic unit test integrated with CI/CD pipeline.

Feb 25th - team meeting.

### **New decisions/plans:**

- Begin to write unit test for classes.
- Leaderboard and user details UI update

### **Task Completion from Last Week.**



1. Produced logo mock.
2. Decided UI theme and styling.
3. Fixed bugs.
4. Added team Details component.
5. Added test Line Chart to User Details.
6. Updated styles and color files.

### **Task for This Week**

1. Fix bugs - Ensure that the new database design does not affect team (create team and add team member) and new report (Creating new swear type)
2. Fix SORT button.
3. User stories (using Balsamiq)
4. Flow diagrams.
5. Configure the extension.
6. Fix the user Details component.

New report page:

## New Report

Select the Violation

- ☐ Profanity
- ☐ Muted Microphone
- ☐ Late Arrival

**Next**

## New Report

Add new Swear Type

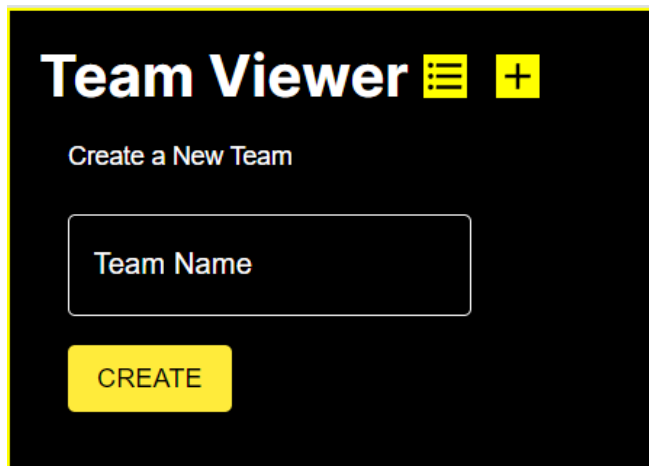
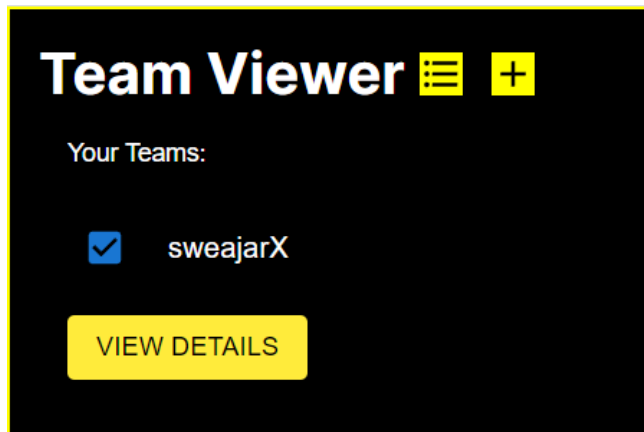
Name

Description

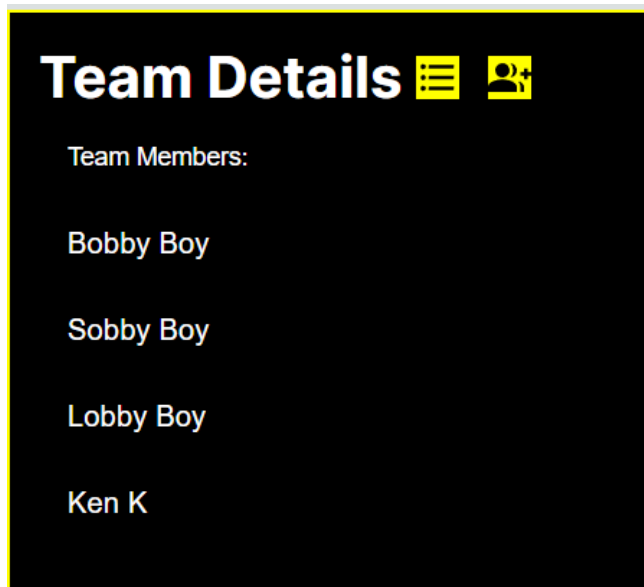
Select Level

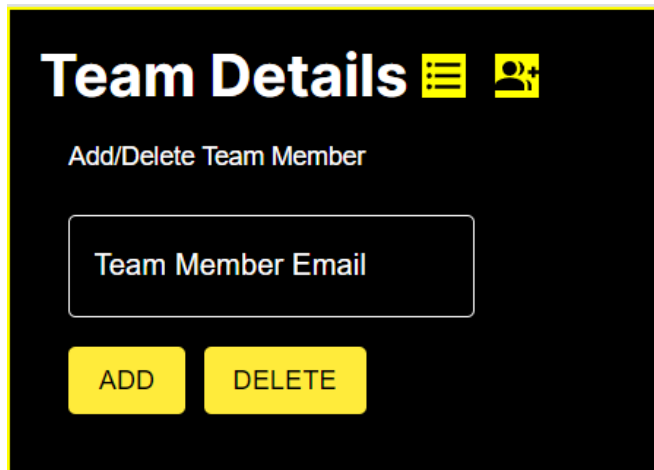
SUBMIT

Team Viewer:



Team Details:





## Week 9: Weekly Report for COMP 4910 Project Course

### Summary:

Our team continued development of a basic prototype. We refined several UI components and have set up the database on Firebase. A chrome extension has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

### Summary of two last weeks (Feb 26th - Mar 3rd) of the team:

Feb 26th - continued writing unit tests.

Feb 27th - team meeting. Discussed the direction of the project.

Feb 28th - updated the design of TeamViewer, Report and Details.

Mar 1st - sixth meeting with Kevin.

Mar 2nd - team meeting.

Mar 3rd - Midterm review.

### New decisions/plans:

- Continue writing unit test for classes.
- Leaderboard and user details UI update

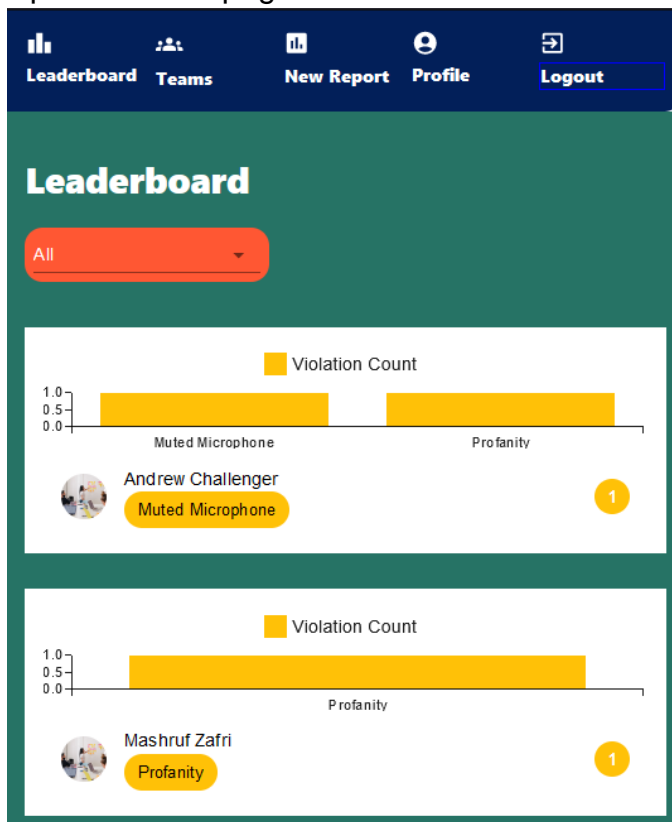
### Task Completion from Last Week.

1. Decided UI theme and styling.
2. Fixed bugs
3. Updated team Details component.
4. Updated styles and color files.
5. Prepared midterm review presentation.

### Task for This Week

1. Fix bugs - Ensure that the new database design does not affect team (create team and add team member) and new report (Creating new swear type)
2. Fix SORT button.
3. User stories (using Balsamiq)
4. Flow diagrams.
5. Configure the extension.
6. Fix the user Details component.
7. Add admin chip - picture, name and chip that says admin.
8. Resize Team Viewer, Report and Details, show pictures.
9. Add graphs to Team Details, nicely place the name list. Show picture.
10. Legend and clickable description for the violations in User Details Charts
11. Categorize the violations so that it is not a huge list to scroll through.
12. Report page, next button should show below the CHECKED box(es).
13. Just show the email on the employees list on the report page.
14. Midterm review feedback.

Updated Homepage Leaderboard:



New report page:

## New Report

Select the Violation

- ☐ Profanity  
Choose this option if the user used inappropriate language
- ☐ Muted Microphone  
Choose this option if the user's microphone was muted
- ☐ Late Arrival  
Choose this option if the user joined the meeting late
- ☐ Goofy 🤪🤪  
Choose this option if the user is goofy on meeting

Select users to report.

## Employees List

REPORT	
 Andrew Challenger	<input type="checkbox"/>
 Mashruf Zafri	<input type="checkbox"/>
 name lname	<input type="checkbox"/>
 Kenneth Kouadio	<input type="checkbox"/>

### Week 10: Weekly Report for COMP 4910 Project Course

#### Summary:

Our team continued development of a basic prototype. We refined several UI components and have set up the database on Firebase. A chrome extension has been

developed as well. Andrew was briefed on our progress and has given us insightful feedback.

#### Summary of the last week (Mar 4th - Mar 10th) of the team:

Mar 4th - presented Midterm review presentation in class.

Mar 5th - team meeting - decided on UI restructuring.

Mar 7th - team meeting. Worked on unit tests.

Mar 8th - seventh meeting with Kevin. Created a plan for the remaining weeks.

Mar 10th - team meeting/seventh meeting with Andrew.

#### **New decisions/plans:**

- Continue writing unit test for classes.
- Leaderboard, team viewer, new report, and user details UI update

#### **Task Completion from Last Week.**

1. Decided UI theme and styling.
2. Fixed bugs.
3. Wrote more unit tests.
4. Updated Profile component.
5. Updated styles and color files.
6. Midterm review feedback.
7. Fixed login issue.
8. Added pictures to Team Details.

#### **Task for This Week**

1. Fix bugs - Ensure that the new database design does not affect team (create team and add team member) and new report (Creating new violation Type)
2. Legend and clickable description for the violations in User Details Charts
3. Just show the email in the employees list in the report page
4. Write more unit tests.
5. Leaderboard
  - a. 3d card view
  - b. Remove chart for selected violation sort.
6. TeamViewer
  - a. Square Avatar view - <https://dribbble.com/shots/18286703-Team-Page-UI-Design>
  - b. Show team selection as shadow.
  - c. When no data show a nice text and icon description
  - d. View Details Button reposition.
  - e. Icon buttons reposition.



- f. Add function to delete a team.
- 7. Navbar
  - a. Show tab selection as shadow.
- 8. Report
  - a. Categorize all the violations.
  - b. Let users add own violations (Your Category)
  - c. Bubble view for violation types.
  - d. Report page, next button should show below the CHECKED box(es).
- 13. Team Details
  - a. Card view
  - b. Reposition buttons.
  - c. Add admin chip - picture, name and chip that says admin.
- 14. Add new swear type
  - a. Resize input text.
  - b. Labels should provide more instruction.

## **Week 11: Weekly Report for COMP 4910 Project Course**

### **Summary:**

Our team continued development of a basic prototype. We refined several UI components and have set up the database on Firebase. A chrome extension has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

### Summary of the last week (Mar 11th - Mar 17th) of the team:

Mar 13th - team meeting - decided on UI restructures.

Mar 15th - eighth meeting with Kevin.

Mar 16th - team meeting.

Mar 17th - team meeting/eighth meeting with Andrew.

### **New decisions/plans:**

- Continue writing unit test for classes.
- Leaderboard, team viewer, new report, and user details UI update

### **Task Completion from Last Week.**

1. Decided UI theme and styling.
2. Fixed bugs
3. Wrote more unit tests.
4. Updated Profile component.
5. Updated styles and color files.

6. Fixed login issue.
7. Created a Close Out Plan for the rest of the weeks left with the client.
8. Implemented legend and clickable description for the violations in User Details Charts.
9. Add new swear type.
  - a. Changed size of the input text.
  - b. Labels provide more instruction.
10. TeamViewer
  - a. Square Avatar view - <https://dribbble.com/shots/18286703-Team-Page-UI-Design>
  - b. Shows team selection as a border.
  - c. When no data shows a nice text and icon description.
  - d. View Details Button reposition.
  - e. Icon buttons reposition.
11. Report page.
  - a. Users can add a new violation category.
  - b. Next button is fixed in the bottom right corner.

### **Task for This Week**

1. Fix bugs - Ensure that the new database design does not affect team (create team and add team member) and new report (Creating new violation Type)
2. Also show the email in the employees list in the report page
3. Write more unit tests.
4. More description to guide new users.
5. Leaderboard
  - a. 3d card view
  - b. Remove chart for selected violation sort.
6. TeamViewer
  - a. Add function to delete a team.
7. Navbar
  - a. Show tab selection as shadow.
8. Report
  - a. Categorize all the violations.
  - b. Bubble view for violation types.
13. Team Details
  - a. Card view
  - b. Reposition buttons.
  - c. Add admin chip - picture, name and chip that says admin.

## Plan for the remainder of the course.

### Timeline

	Week 11	Week 12	Week 13	Week 14
	Mar 11 - Mar 17	Mar 18 - Mar 24	Mar 25 - Mar 31	April 1 - April 5
UI improvements				
Unit Testing				
Documentation & Final Code Review				

### Unit Testing

Aiming to achieve 50% threshold (currently 37%)

### Documentation

- Readme
- Final report

**Final Presentation Date:** April 7th (Tentative)

### Week 12: Weekly Report for COMP 4910 Project Course

#### Summary:

Our team continued development of a basic prototype. We refined several UI components and continued writing unit tests. A chrome extension has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

#### Summary of the last week (Mar 18th - Mar 24th) of the team:

Mar 20th - UI update.

Mar 21st - rewrote several unit tests.

Mar 22nd - ninth meeting with Kevin.

Mar 23rd - team meeting.

Mar 24th - team meeting/ninth meeting with Andrew.

**New decisions/plans:**

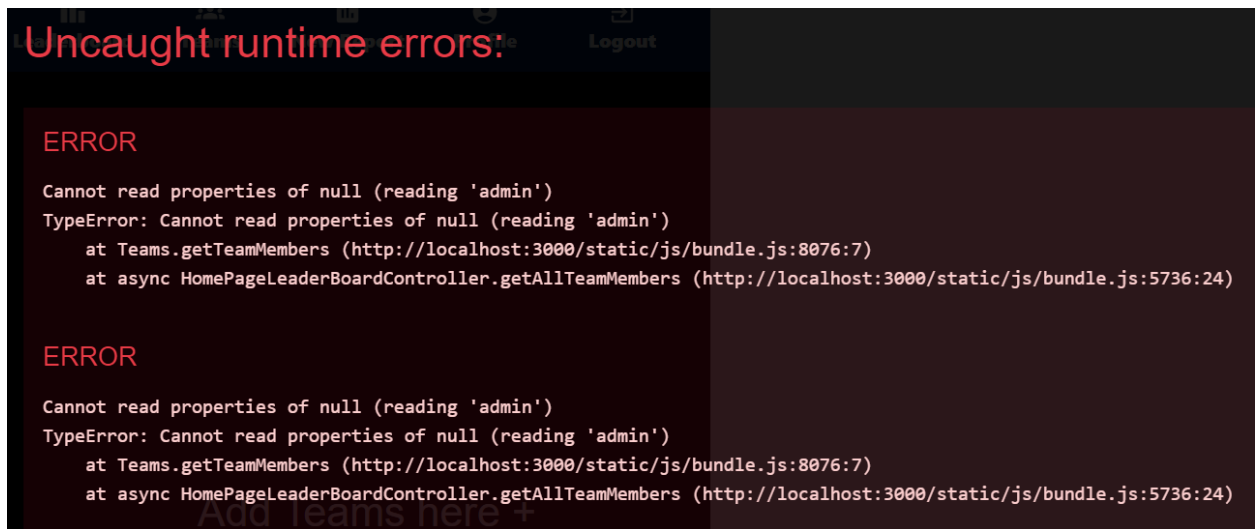
- Continue writing unit test for classes.

**Task Completion from Last Week.**

1. Decided UI theme and styling.
2. Fixed bugs.
3. Wrote more unit tests.
4. Updated styles and color files.
5. Added more description to guide new users.
6. Report
  - a. Categorized all the violations.
7. TeamViewer
  - a. Added function to delete a team.
8. Navbar
  - a. Shows tab selection as shadow.
9. Leaderboard
  - a. Removed chart for selected violation sort.
10. Team Details
  - a. Card view.
  - b. Repositioned buttons.
  - c. Added admin chip - picture, name and chip that says admin.

**Task for This Week**

1. Fix bugs - Ensure that the new database design does not affect team (create team and add team member) and new report (Creating new violation Type)
2. Also show the email in the employees list in the report page
3. Write more unit tests.
4. Add/Delete categories and violations.
5. After clicking NEXT from violations, it goes to the Employees List. It should show the emails along with the employees' names.
6. Add tutorial.
7. Readme: description, how it works, how to set it up, how to get it started for someone new
8. Bug: Switching to Leaderboard after deleting a team



9. Report

- a. Bubble view for violation types.

Sign In / Sign Up:

---

## Welcome to SwearJar



Are you tired of your foul-mouthed habits? 🤬 🤬  
Well, fear not! SwearJar is here to help you keep those profanities in check. 😊 😊

SIGN UP NOW

SIGN IN

## Sign In

Build your company culture

Email

Password

SIGN IN

BACK

---

## Sign Up

First Name

Last Name





Email

Password

SIGN UP

BACK













Updated Homepage Leaderboard:

Leaderboard			All
	name lname	0	
	Mashruf Zafri	0	
	John Doe Muted Microphone	1	
	Iskander Sermanizov	0	

New report page:





## New Report

Pick the category that best describes the behaviour of your coworker

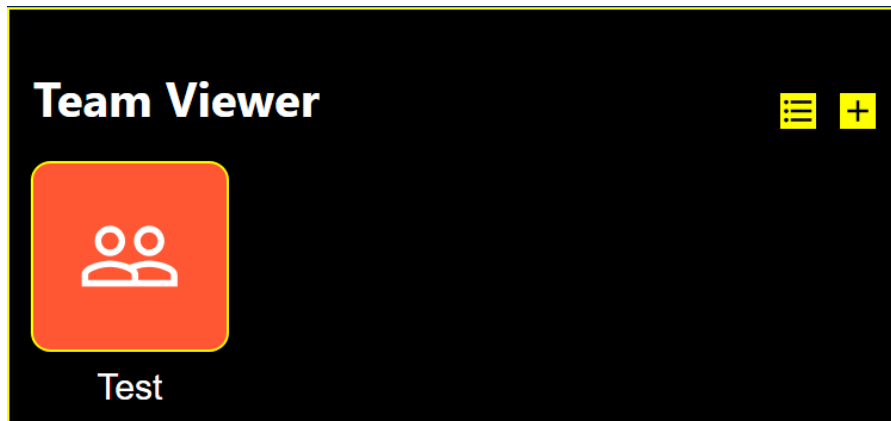
 Office Culture Enthusiast	 Digital Explorer	 Idea Maverick	 Rhythmic Maestro
 Idea Maverick	 Rhythmic Maestro	 Virtual Costume Artist	 Office Detective
 Virtual Cheerleader	 Meeting Scheduler Extraordinaire	 Office Puppet Master	 Microclimate Meteorologist

Select users to report.





Employees List		REPORT
	name lname	<input type="checkbox"/>
	Mashruf Zafri	<input type="checkbox"/>
	John Doe	<input type="checkbox"/>
	Iskander Sermanizov	<input type="checkbox"/>

Team Viewer:



# Team Viewer





Create a New Team / Delete a Team

CREATE


DELETE

Team Details:

## Team Details




Team Members:




name lname  
mail@gmail.com


Admin



Mashruf Zafri  
mzafri@gmail.com


Member






John Doe  
jdoe@gmail.com


Member

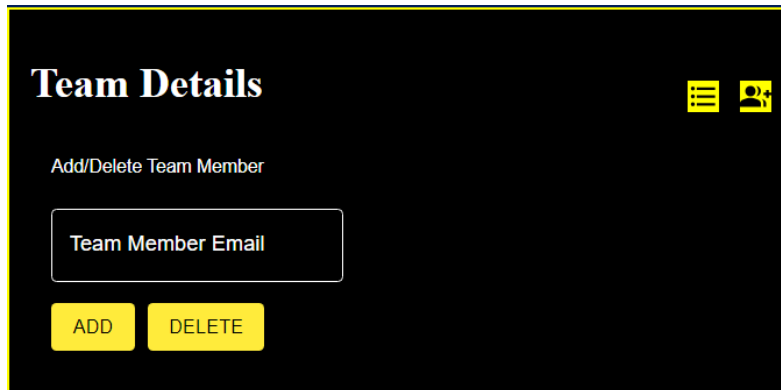




Iskander Sermanizov  
isermaniz@gmail.com

Member





## Week 13: Weekly Report for COMP 4910 Project Course

### Summary:

Our team continued development of a basic prototype. We refined several UI components and continued writing unit tests. A chrome extension has been developed as well. Andrew was briefed on our progress and has given us insightful feedback.

### Summary of the last week (Mar 25th - Mar 31st) of the team:

Mar 26th - created a Readme file.

Mar 27th - minor UI updates.

Mar 28th - fixed broken tests.

Mar 29th - added new unit tests and minor UI updates.

Mar 31st - team meeting.

### New decisions/plans:

- Continue writing unit test for classes.

### Task Completion from Last Week.

1. Decided UI theme and styling.
2. Fixed bugs
3. Wrote more unit tests.
4. Updated styles and color files.
5. Added more description to guide new users.
6. Fixed Bug: Switching to Leaderboard after deleting a team.
7. Created a Readme file.
8. The Employees List shows the emails along with the employees' names.

### Task for This Week

1. Write more unit tests.
2. Add/Delete categories and violations.

3. Add a tutorial.
4. Report
  - a. Bubble view for violation types (Tentative)

# GROUP 9

## Assignment 2

Kenneth Kouadio (T00633691)  
Iskander Sermanizov (T00645991)  
Mashruf Ahmed Zafri (T00657643)

## Company

### SuperFly Aerial

- SuperFly Aerial: Drone service since 2015, expanded to engineering and industry.
- Custom Thermographic Sensor: Developing with specialized software.
- Functionality: Integrates real-time thermographic imagery with IMU data for 3D temperature location.
- Target Market: Focus on efficient thermographic inspections for engineering and industry.
- Comprehensive Solution: End-to-end services, showcasing innovation in the drone industry.

## Problem Description

- Identifying Office Challenges: SuperFly Aerial recognizes a need for addressing issues like profanity, late arrivals, and forgetting to unmute during meetings.
- Project Objective: Develop an integrated app or chatbot to track and manage these common office fouls efficiently.
- Public Domain Access: The goal is to create a freely available tool accessible to everyone, promoting a collaborative and positive work environment.
- Streamlining Accountability: The project aims to enhance workplace etiquette by providing a user-friendly solution for tracking and improving behavior in meetings.
- Innovation for All: SuperFly Aerial's commitment to fostering a productive and respectful workplace through accessible technology.

## Project - SuperFly SwearJar

A standalone program/chrome extension with user-friendly UI that will keep track of harmless workplace fouls from employees. A database will store these information and monthly/weekly send our reports. Mostly tracking reports will be done manually by admins by a few mouse clicks.

## Workflow

Firebase Database (NoSQL) to store information. Firebase has a larger storage space over any other free services.

Next.js to maintain single code base which has both backend and frontend.

Vercel for deployment

Git + Github for collaborations

Agile for software development process

## Questions?



# GROUP 9

## Assignment 3

Kenneth Kouadio (T00633691)  
Iskander Sermanizov (T00645991)  
Mashruf Ahmed Zafri (T00657643)

### Company

#### SuperFly Aerial

- SuperFly Aerial: Drone service since 2015, expanded to engineering and industry.
- Custom Thermographic Sensor: Developing with specialized software.
- Functionality: Integrates real-time thermographic imagery with IMU data for 3D temperature location.
- Target Market: Focus on efficient thermographic inspections for engineering and industry.
- Comprehensive Solution: End-to-end services, showcasing innovation in the drone industry.



## Problem Description

- Identifying Office Challenges: SuperFly Aerial recognizes a need for addressing issues like profanity, late arrivals, and forgetting to unmute during meetings.
- Project Objective: Develop an integrated app or chatbot to track and manage these common office fouls efficiently.
- Public Domain Access: The goal is to create a freely available tool accessible to everyone, promoting a collaborative and positive work environment.
- Streamlining Accountability: The project aims to enhance workplace etiquette by providing a user-friendly solution for tracking and improving behavior in meetings.
- Innovation for All: SuperFly Aerial's commitment to fostering a productive and respectful workplace through accessible technology.

## Project - SuperFly SwearJar

A standalone program/chrome extension with user-friendly UI that will keep track of harmless workplace fouls from employees. A database will store these information and monthly/weekly send our reports. Mostly tracking reports will be done manually by admins by a few mouse clicks.

## Workflow

Firebase Database (NoSQL) to store information. Firebase has a larger storage space over any other free services.

Next.js to maintain single code base which has both backend and frontend.

Vercel for deployment

Git + Github for collaborations

Agile for software development process

## Requirements

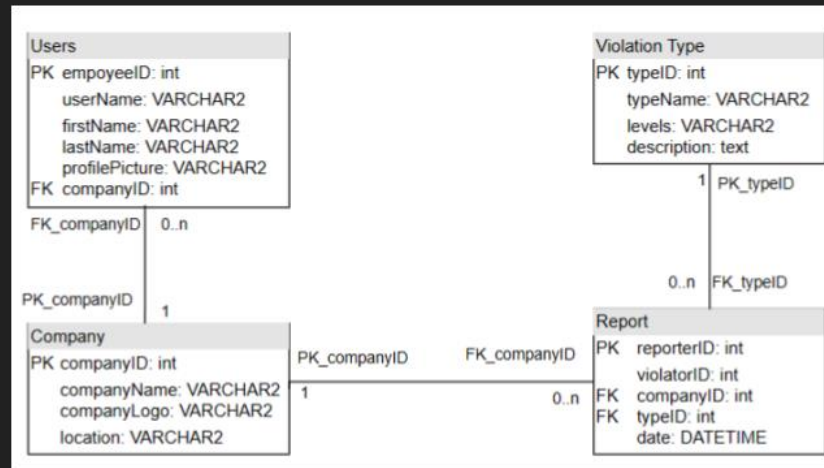
Be as configurable as reasonably possible to allow for application to various situations. The user should be able to create/edit/delete teams, users and violation categories.

Have rudimentary multi tenant functionality and administration. Multiple Teams should be supported by the app without information leaking between them. Recommend making it as self-serve/open as possible, while protecting the personal information of existing teams.

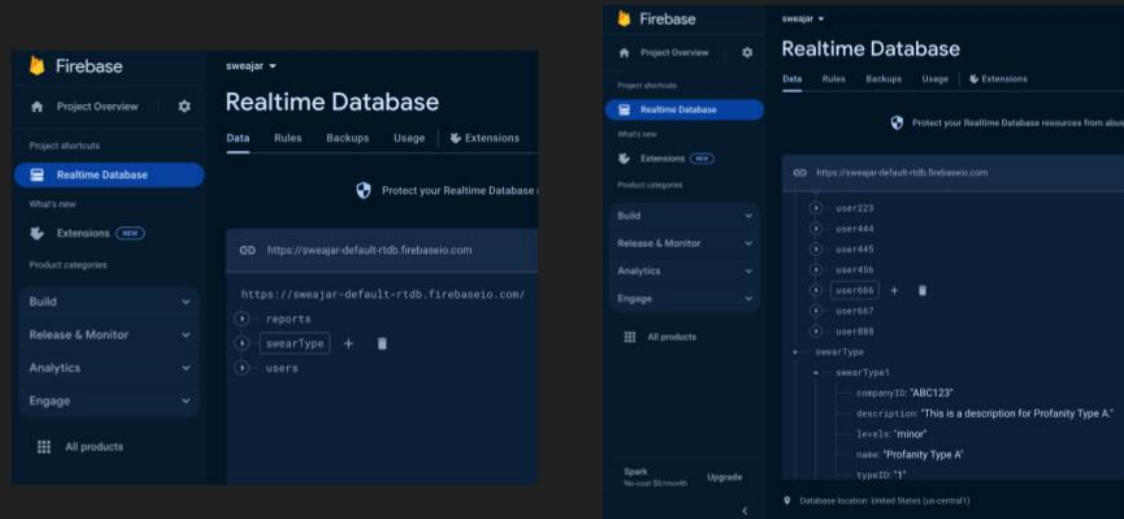
Have a dashboard showing team statistics over the past 30, 90, 365 days, with a simple interface for logging violations of the various categories for every user on the team.

Be designed with the intention of making the system a free service open to all users.

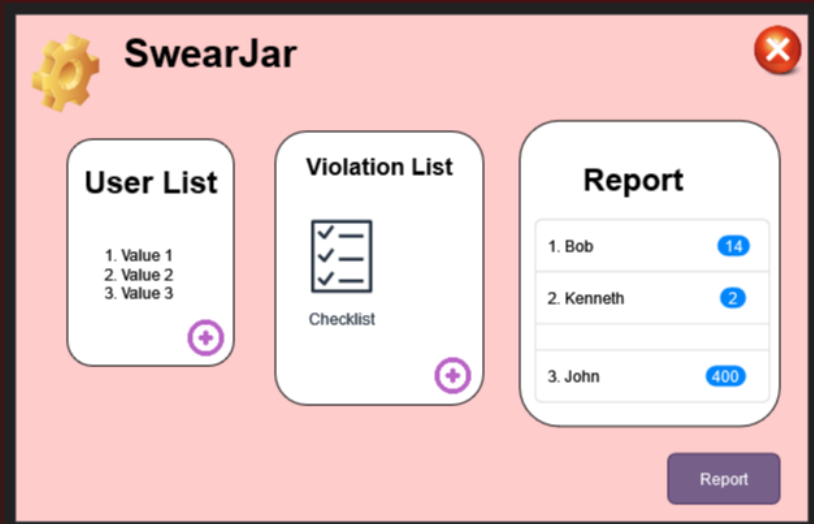
## Database Design UML



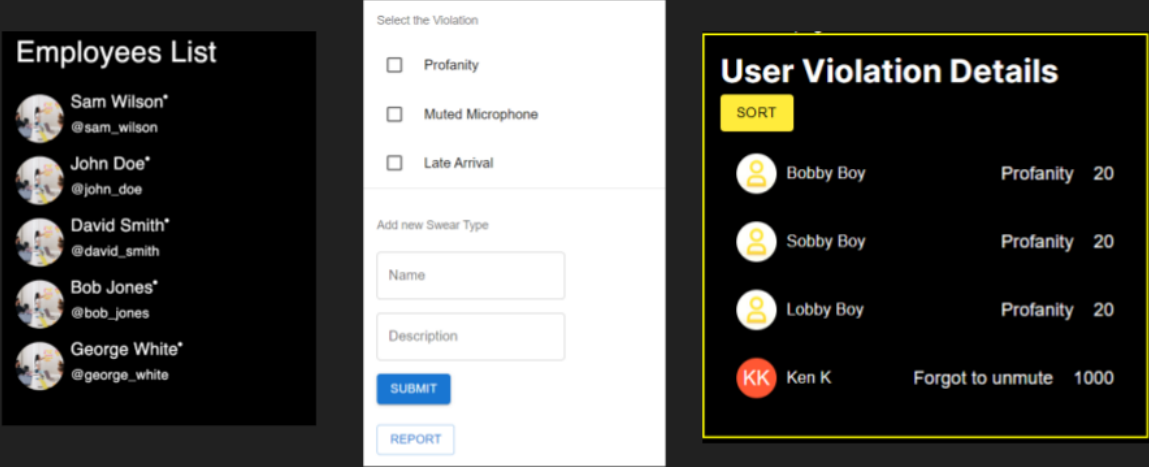
## Database



# UI Wireframe



# User Interface



Questions?



# GROUP 9

## Assignment 4 - Midterm Review

Kenneth Kouadio (T00633691)  
Iskander Sermanizov (T00645991)  
Mashruf Ahmed Zafri (T00657643)

### Company

#### SuperFly Aerial

- SuperFly Aerial: Drone service since 2015, expanded to engineering and industry.
- Custom Thermographic Sensor: Developing with specialized software.
- Functionality: Integrates real-time thermographic imagery with IMU data for 3D temperature location.
- Target Market: Focus on efficient thermographic inspections for engineering and industry.
- Comprehensive Solution: End-to-end services, showcasing innovation in the drone industry.

ISK

## Problem Description

- Identifying Office Challenges: SuperFly Aerial recognizes a need for addressing issues like profanity, late arrivals, and forgetting to unmute during meetings.
- Project Objective: Develop an integrated app or chatbot to track and manage these common office fouls efficiently.
- Public Domain Access: The goal is to create a freely available tool accessible to everyone, promoting a collaborative and positive work environment.
- Streamlining Accountability: The project aims to enhance workplace etiquette by providing a user-friendly solution for tracking and improving behavior in meetings.
- Innovation for All: SuperFly Aerial's commitment to fostering a productive and respectful workplace through accessible technology.

ISK

## Project - SuperFly SwearJar

A standalone program/chrome extension with user-friendly UI that will keep track of harmless workplace fouls from employees. A database will store these information and monthly/weekly send our reports. Mostly tracking reports will be done manually by admins by a few mouse clicks.

ISK

## Workflow / Architecture

Firebase Database (NoSQL) to store information. Firebase has a larger storage space over any other free services.

Next.js / React.js to maintain single code base which has both backend and frontend ( MUI)

Violations types - Chatgpt

Vercel / Netlify for deployment

Git + Github for collaborations

CI and CD - Github Actions + Unit Tests

Agile for software development process

KEN

## Requirements

MAZ

Be as configurable as reasonably possible to allow for application to various situations. The user should be able to create/edit/delete teams, users and violation categories.

Have rudimentary multi tenant functionality and administration. Multiple Teams should be supported by the app without information leaking between them. Recommend making it as self-serve/open as possible, while protecting the personal information of existing teams.

Have a dashboard showing team statistics over the past 30, 90, 365 days, with a simple interface for logging violations of the various categories for every user on the team.

Be designed with the intention of making the system a free service open to all users.



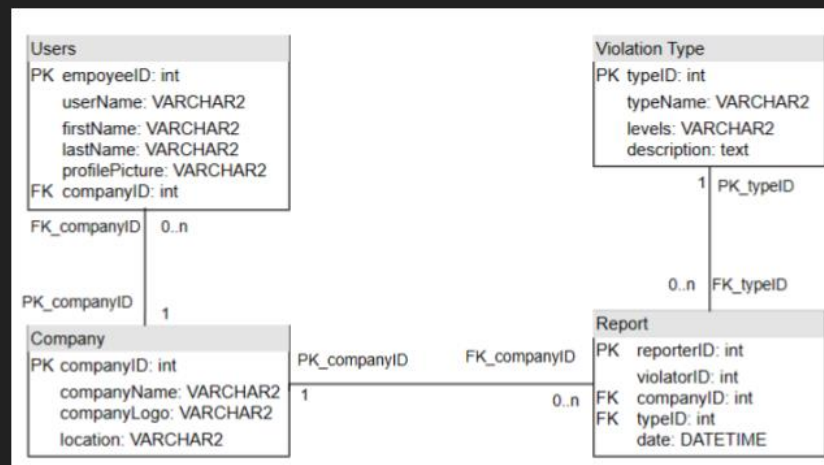
## Business Decisions

- Develop a standalone chrome extension to facilitate accessibility
- Whether to delete user data or not when they leave the team
- How CI/CD pipeline facilitate the deployment of new features pretty easily
- Firebase because it is cheap : 1Gb of DB storage, 5GB of Blob storage
- Chrome extension one time fee of about \$5

KEN

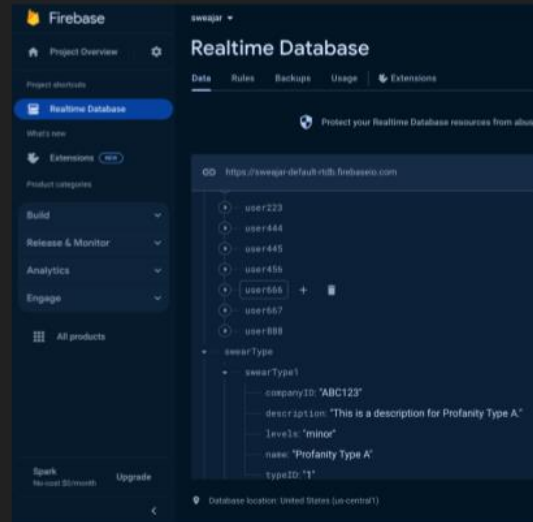
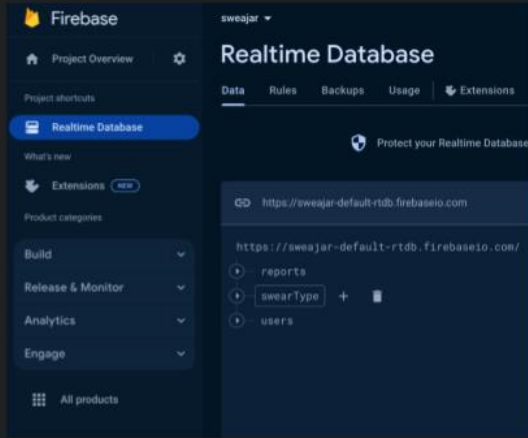
## Database Design UML

ISK



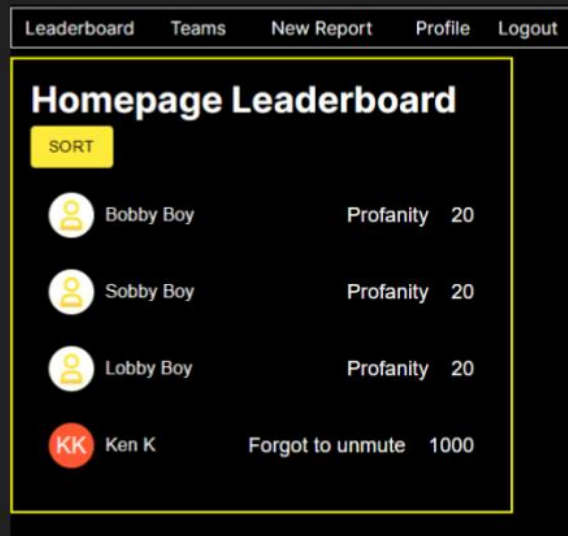
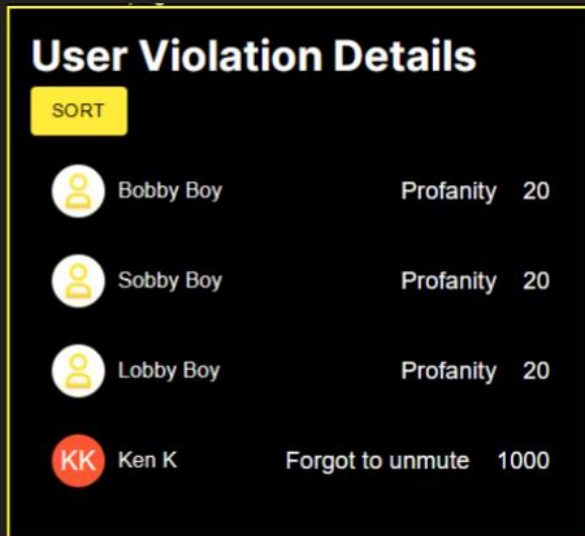
# Database

KEN

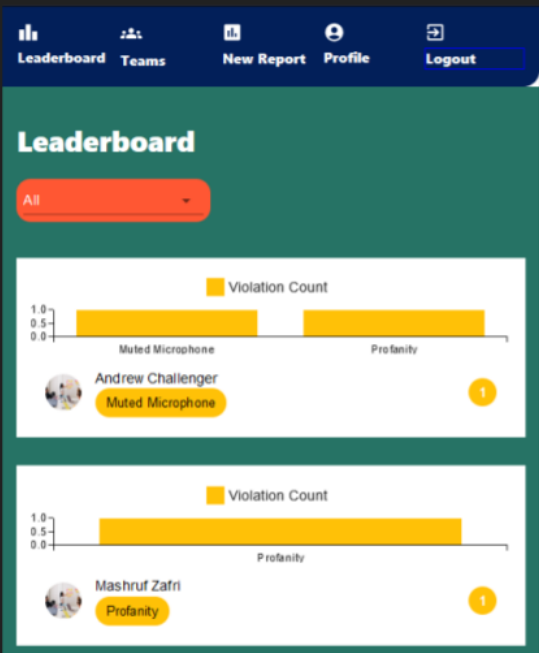
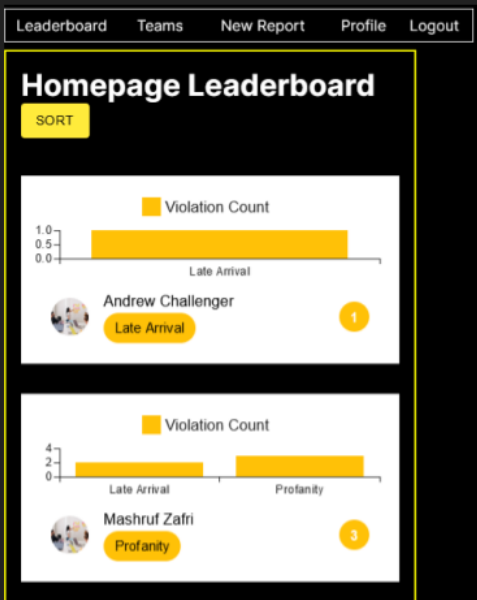


# UI Evolution - Leaderboard

MAZ



# Leaderboard Continued



# Team Viewer

## Team Viewer

☒ sweajarX

Create a New Team

Team Name

**CREATE**

Add Team Member

Team Member Email

**ADD**

## Team Viewer

☒ sweajarX

Create a New Team

Team Name

**CREATE**

Add Team Member

Team Member Email

**ADD**

## Team Viewer

ISK

Your Teams:

☒ sweajarX

**VIEW DETAILS**

Create a New Team

Team Name

**CREATE**

# New Report

ISK

Select the Violation

☐ Profanity

☐ Muted Microphone

☐ Late Arrival

Add new Swear Type

Name

Description

**SUBMIT**

**REPORT**

## New Report

Select the Violation

- ☐ Profanity
- ☐ Muted Microphone
- ☐ Late Arrival

**Next**

Add new Swear Type

Name

Description

Select Level

**SUBMIT**

## New Report

Select the Violation

- ☐ Profanity  
Choose this option if the user used inappropriate language
- ☐ Muted Microphone  
Choose this option if the user's microphone was muted
- ☐ Late Arrival  
Choose this option if the user joined the meeting late
- ☐ Goofy 🤡🤡  
Choose this option if the user is goofy on meeting

# Some of the violation categories

KEN

- ☐ Office Jester  
Choose this option if the user could make even the grumpiest colleague crack a smile with their quick wit 🤡🤡
- ☐ Lunchtime General  
Choose this option if the user plans lunch outings with military precision and tactical precision 🍴🕒
- ☐ Happy Hour Maestro  
Choose this option if the user is the grand conductor of post-work shenanigans and merriment 🍷🎵
- ☐ Office Plant Whisperer  
Choose this option if the user confides in office plants as their personal confidants and leafy therapists 🌿🗨️
- ☐ Spreadsheet Sorcerer  
Choose this option if the user can turn mundane data into a magical masterpiece in Excel 📊🪄

- ☐ Office Bard  
Choose this option if the user expresses thoughts and requests in poetic form, turning daily interactions into lyrical verses 🎭🎵
- ☐ Office Jester  
Choose this option if the user keeps a rubber chicken handy for impromptu comedy moments, spreading laughter throughout the office 🐔🤡
- ☐ Conference Call Ninja  
Choose this option if the user can juggle multiple calls without missing a beat, like a stealthy master of virtual meetings 🥷📞
- ☐ Office Socrates  
Choose this option if the user engages colleagues in deep discussions about life and work, pondering the profound in the midst of office-hustle and bustle 🧠💡
- ☐ Culinary Innovator  
Choose this option if the user experiments with new recipes and treats the office to delightful samples, turning the break room into a gourmet laboratory 🍳🔬

- ☐ Office Politician  
Choose this option if the user could win an Oscar for their performance in 'The Manipulative Monologues' 🎭🗣️
- ☐ Professional Irresponsible  
Choose this option if the user's motto is 'I'll do it tomorrow...or never' 📅🙅
- ☐ Coffee Connoisseur  
Choose this option if the user sees coffee as the elixir of life ☕🧘
- ☐ Meme Maestro  
Choose this option if the user's superpower is turning frowns upside down with the perfect meme 🤡👤
- ☐ DJ of the Office  
Choose this option if the user's playlist could win a Grammy for 'Best Office Soundtrack' 🎵🎧
- ☐ Pun Prince/Princess  
Choose this option if the user's pun game is so strong it should be illegal 🗨️🤡

## Employees List





KEN

### Employees List

-  Sam Wilson\*  
@sam\_wilson
-  John Doe\*  
@john\_doe
-  David Smith\*  
@david\_smith
-  Bob Jones\*  
@bob\_jones
-  George White\*  
@george\_white

### Employees List

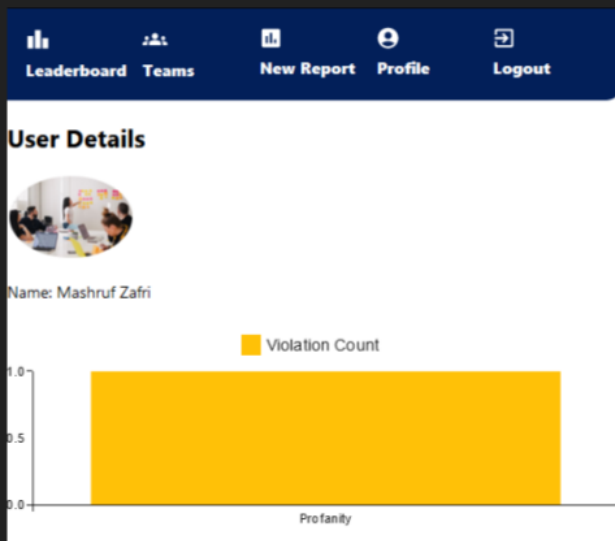
REPORT

-  Andrew Challenger
-  Mashruf Zafri
-  name lname
-  Kenneth Kouadio



## Profile

MAZ



## File Structure Overview

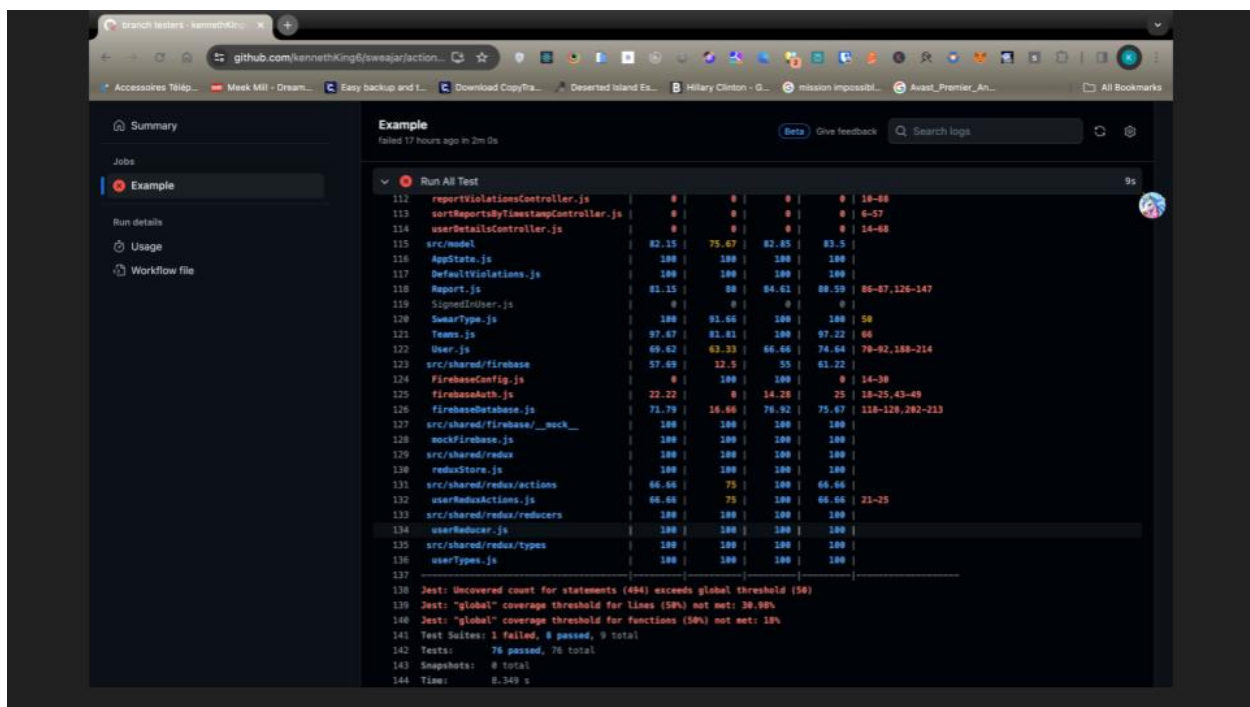
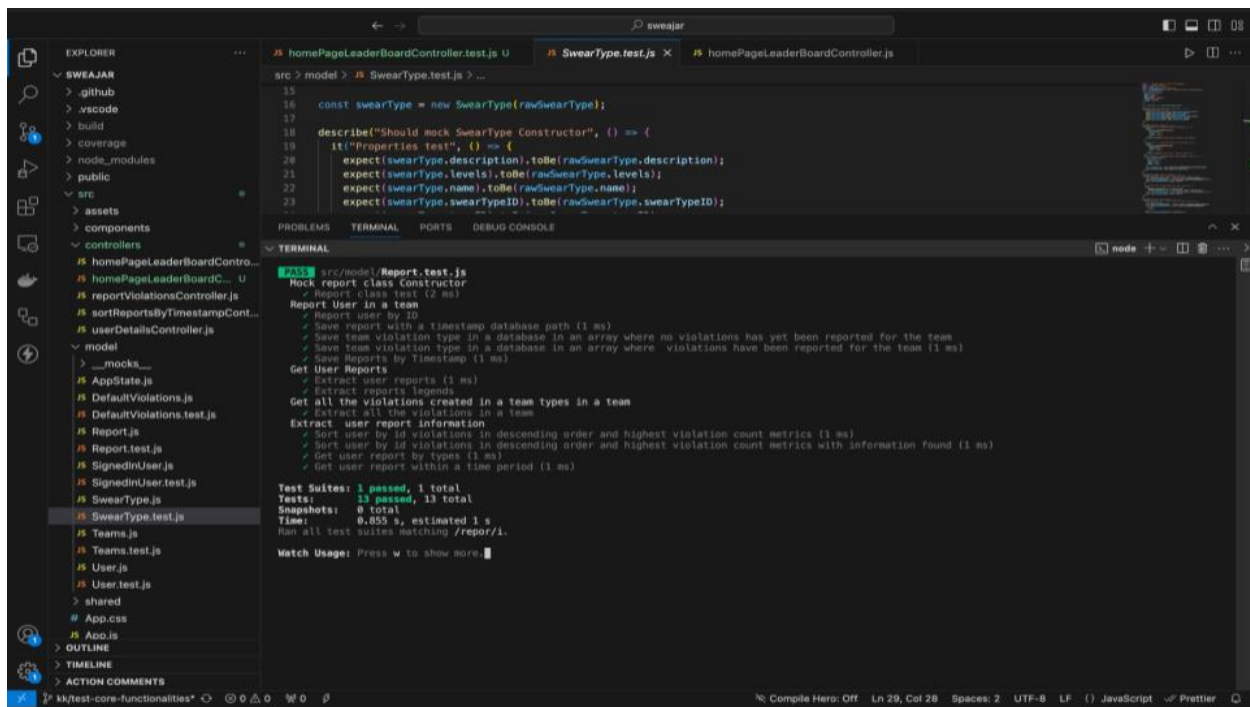
ISK

- Assets
- Components
- Controllers
- Model
- Shared
- Manifest - Chrome extension configuration
- Package.json - CI/CD, Jest, Jest Watch, Prettier, Build
- .github - github actions workflows
- README

## Tests - Jest + Github Actions

MAZ

- Increase confidence
- Quickly catch bugs
- Main and feature(dev) branches



## Security

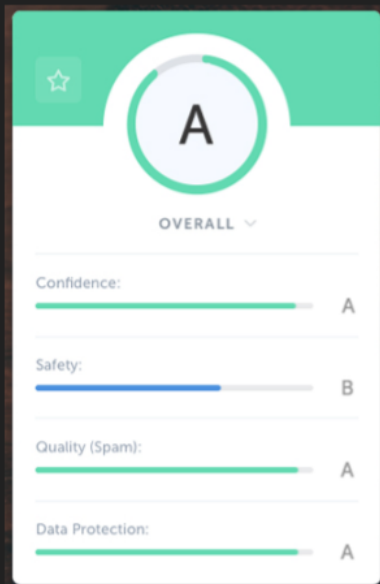
ISK

- Not Exposing credentials: Database credentials are stored in an .env and never pushed to github
- Each Team is separated from each other and managed by the admin.
- No Data leak between teams and users

## Next Steps

- UI razzle-dazzle
- Unit Tests - 25% left to meet the 50% threshold
- Add Graphs for Team details
- Delete a team
- Add Legends and Bar Chart series in User Details (Profile)
- Beta Testing: <https://swearjar-extension.netlify.app/>
- Upload/Scraping User Profile Pictures





User Profile Inspiration



Look and feel of the extension.  
Will include yellow

Questions?



# GROUP 9

## Show Case

Kenneth Kouadio (T00633691)  
Iskander Sermanizov (T00645991)  
Mashruf Ahmed Zafri (T00657643)

### Company - SuperFly Aerial

SuperFly Aerial has been a drone service company since 2015, and later expanded to engineering and industry. Key highlights:

- Custom Thermographic Sensor: Developed with specialized software, this sensor integrates real-time thermographic imagery with IMU data for precise 3D temperature location.
- Target Market: Focus on efficient thermographic inspections for the engineering and industry sectors.
- Comprehensive Solution: Provides end-to-end services, showcasing innovation in the drone industry.

## Problem Description

- Identifying Office Challenges: SuperFly Aerial recognizes a need for addressing issues like profanity, late arrivals, and forgetting to unmute during meetings.
- Project Objective: Develop an integrated app or chatbot to track and manage these common office fouls efficiently.
- Public Domain Access: The goal is to create a freely available tool accessible to everyone, promoting a collaborative and positive work environment.
- Streamlining Accountability: The project aims to enhance workplace etiquette by providing a user-friendly solution for tracking and improving behavior in meetings.
- Innovation for All: SuperFly Aerial's commitment to fostering a productive and respectful workplace through accessible technology.

## Project - SuperFly SwearJar

A standalone program/chrome extension with user-friendly UI that will keep track of harmless workplace fouls from employees. A database will store this information and monthly/weekly send our reports. Mostly tracking reports will be done manually by admins by a few mouse clicks.

## Workflow / Architecture

Firebase Database (NoSQL) to store information. Firebase has a larger storage space over any other free services.

Next.js / React.js to maintain single code base which has both backend and frontend ( MUI)

Violations types - Chatgpt

Vercel / Netlify for deployment

Git + Github for collaborations

CI and CD - Github Actions + Unit Tests

Agile for software development process

## Requirements

Be as configurable as reasonably possible to allow for application to various situations. The user should be able to create/edit/delete teams, users and violation categories.

Have rudimentary multi tenant functionality and administration. Multiple Teams should be supported by the app without information leaking between them. Recommend making it as self-serve/open as possible, while protecting the personal information of existing teams.

Have a dashboard showing team statistics over the past 30, 90, 365 days, with a simple interface for logging violations of the various categories for every user on the team.

Be designed with the intention of making the system a free service open to all users.

## Business Decisions

- Develop a standalone chrome extension to facilitate accessibility
- Whether to delete user data or not when they leave the team
- How CI/CD pipeline facilitate the deployment of new features pretty easily
- Firebase because it is cheap : 1Gb of DB storage, 5GB of Blob storage
- Chrome extension one time fee of about \$5

## Tests - Jest + Github Actions

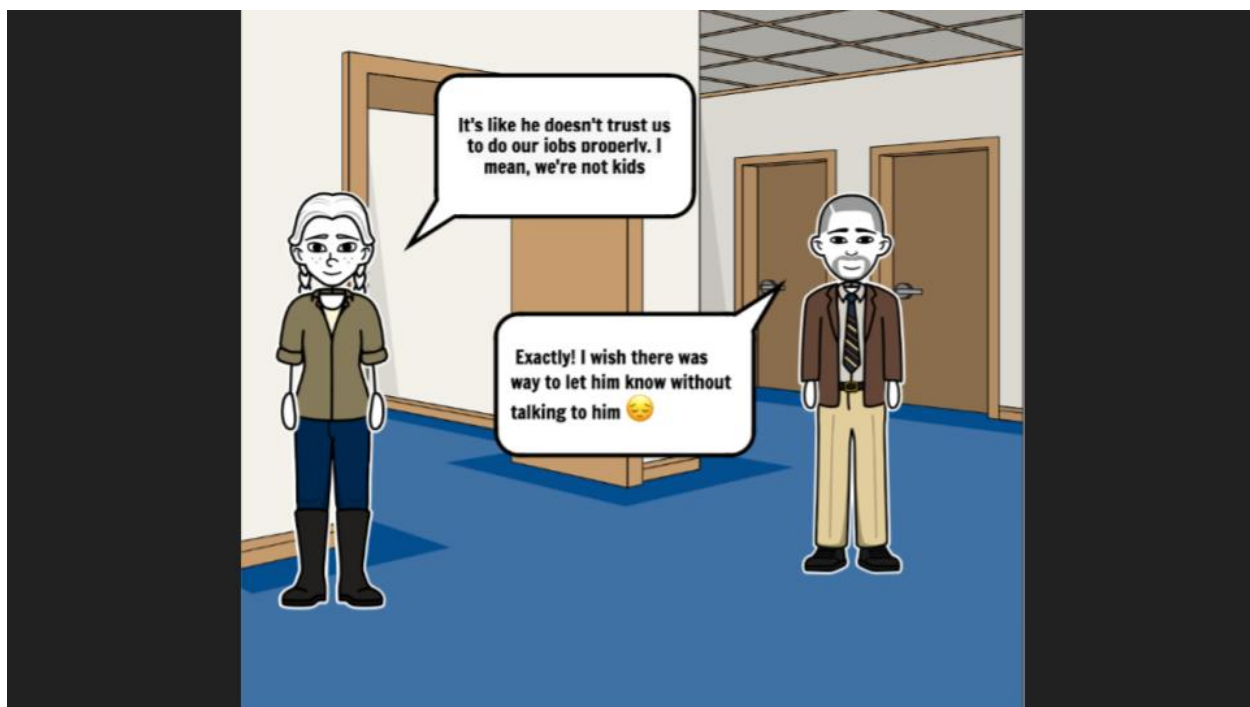
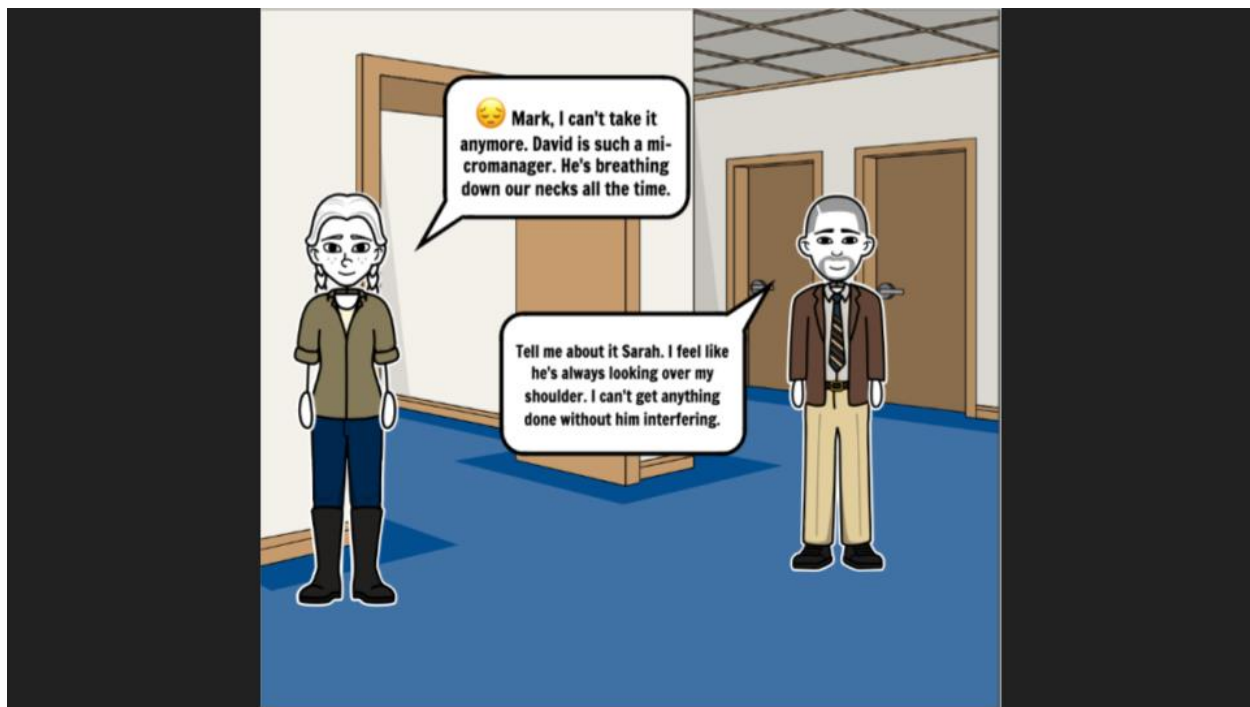
- Increase confidence
- Quickly catch bugs
- Main and feature(dev) branches

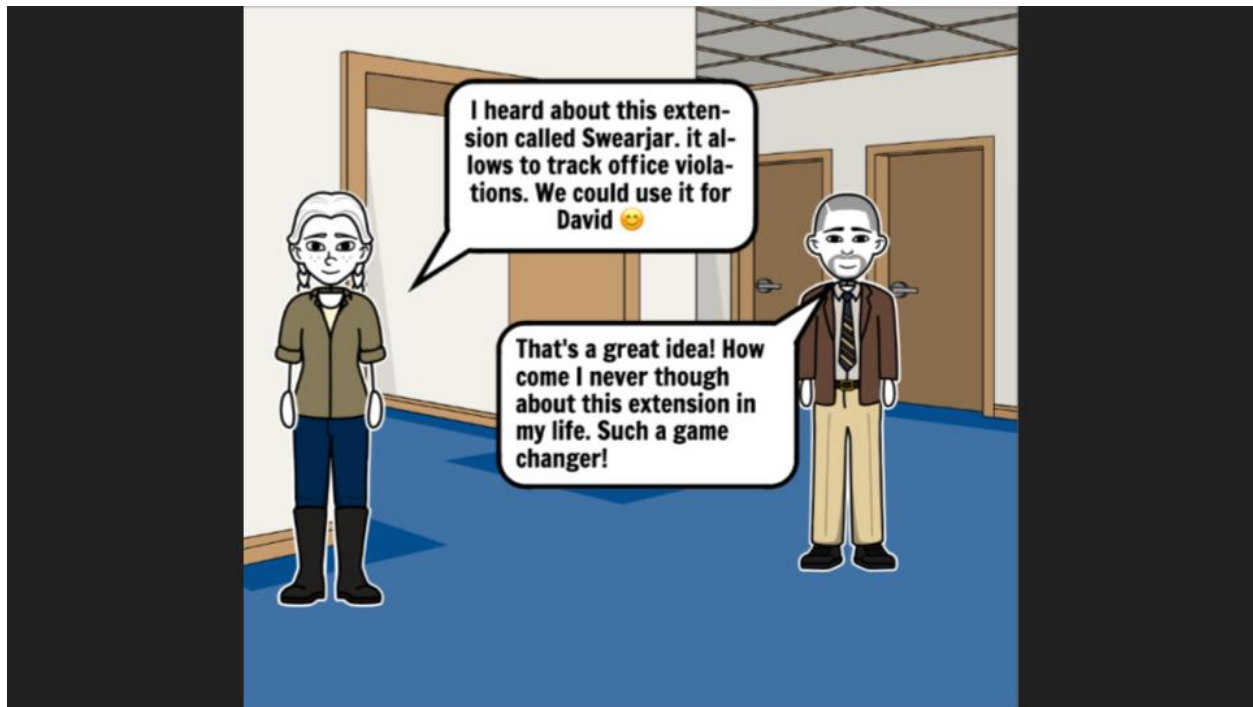
## Security

- Not Exposing credentials: Database credentials are stored in an .env and never pushed to github
- Each Team is separated from each other and managed by the admin.
- No Data leak between teams and users

## Mid Term Review Goals

- UI razzle-dazzle
- Unit Tests - meet the 50% threshold
- Add Graphs for Team details
- Delete a team
- Add Legends and Bar Chart series in User Details (Profile)
- Beta Testing: <https://swearjar-extension.netlify.app/>





## Demonstration

1. Sign In Sarah
2. Select the Team
3. Report David
4. Leaderboard - Click on a user to view Profile
5. Log in as David



Questions?

