

SW Engineering CSC648/848

Fall 2023 Section 2 M2

“ CalAwareNow - Digital Shield for California ”

Team Lead: Ameen Safi (asafi1@mail.sfsu.edu)

Github Administrator/Scrum Master: Dev Patel

Backend Lead / Front end: Himat Shrestha

Frontend Lead: Chih Lin Chien

Database Lead: Fasika Abera

Backend Engineer: Trina Haque

10/03/2023

Functional Requirements - prioritized – Brian

Priority 1 —must have

User Registration: Allow users to register with valid email addresses and confirm their emails.

User Profiles: Registered users can create and manage their profiles, including contact information and notification preferences.

County Data Entry: Enable county department directors to input COVID-19, wildfire, security, and weather metrics.

Role-Based Access Control: Assign roles to users (e.g., public, county director, administrator) to control access and permissions.

Alerting System: Provide a system for sending alerts to registered users based on state guidelines and user preferences.

Alert Preferences: Allow users to set their notification preferences for different types of alerts.

Search Functionality: Implement a search feature for users to find county-specific information.

Mapping Interfaces: Display events like wildfires on interactive maps.

Social Sharing: Allow users to share alerts on social media platforms to increase public awareness.

Priority 2 — desired

Data Validation: Validate data entries to ensure accuracy and consistency.

User Authentication: Implement secure login functionality for registered users.

Responsive Design: The web application is responsive and adapts to various screen sizes, including desktops, laptops, and mobile devices.

Media-Rich Content: The platform displays relevant information using media-rich content like images, videos, and interactive graphics.

System Performance: Ensure the application's performance is optimized to handle a large volume of users and data

User Feedback: Allow users to provide feedback or report issues related to the platform or data accuracy.

Priority 3 — opportunistic

Real-Time Collaboration: Enable real-time collaboration and chat features for county department directors.

Privacy Controls: Users can manage their privacy settings and control the type and frequency of alerts they receive.

Security Compliance: Ensure compliance with data privacy regulations, especially for health data.

User Help and Support: Provide user-friendly help and support resources, including FAQs and a knowledge base.

Security Testing: Perform security testing to identify and address vulnerabilities.

Modularity and Scalability: Develop the system with a modular architecture to facilitate future enhancements and customization for other customers.

UI Mockups and Storyboards (high level only) –brian



DB Organization(high level)

1. Users - represents registered user accounts in the system

- ◆ user_id: unique identifier for each user (key) (e.g., 1, 2, 3)
 - ◆ full_name (e.g., John Doe)
 - ◆ username (e.g., johndoe1)
 - ◆ email (e.g., johndoe@gmail.com)
 - ◆ password (encrypted)
 - ◆ user_type (e.g., county director, public)
 - ◆ county_id (foreign key)
 - ◆ registration_date (timestamp)
 - ◆ last_login (timestamp)
 - ◆ department_id (foreign key)

2. Roles

- ◆ role_id: unique identifier for each role (key) (e.g., 1, 2, 3)
- ◆ role_name (e.g., public, administrator, county director)

3. CA_Counties - represents counties in California

- ◆ county_id: unique identifier for each county (key) (e.g., 1, 2, 3)
- ◆ county_name (e.g., Orange county, Merced county)
- ◆ state (defaults to california)
- ◆ population (e.g., 815,201)
- ◆ county_status (boolean)(e.g., true, false depending on whether there is any public safety concern)
- ◆ county_longitude: double (e.g., [San francisco → 37.773972](#))
- ◆ county_latitude: double (e.g., [San francisco → -122.431297](#))

4. Departments - represents the different county departments responsible for public protection

- ◆ department_id: unique identifier for each department (key) (e.g., 1, 2, 3)
- ◆ department_name (e.g., Health department, Security department)
- ◆ department_location (e.g., 11 abc street, san francisco, CA 90000)

5. Covid_Metrics - stores covid metric for each county

- ◆ health_metric_id: unique identifier for each health metric (key) (e.g., 1, 2, 3)
- ◆ county_id (foreign key)
- ◆ cases_per_100k (e.g., 62.1)
- ◆ deaths_per_100k (e.g., 2.8)
- ◆ recored_at (timestamp)

6. Weather_Metrics - stores weather metric for each county

- ◆ weather_metric_id: unique identifier for each weather metric (key) (e.g., 1, 2, 3)
- ◆ county_id (foreign key)
- ◆ extreme_condition_type (e.g., Thunderstorms, Heat waves)
- ◆ recommended_actions (e.g. Road A blocked due to the weather condition, use Route B)
- ◆ recorded_at (timestamp)

7. Wildfire_Metrics - stores wildfire metric for each county

- ◆ wildfire_metric_id: unique identifier for each wildfire metric (key) (e.g., 1, 2, 3)
- ◆ county_id (foreign key)
- ◆ incident_description (e.g., Fire on areas A, B, and C. People in these areas may be asked to evacuate if the wildfire worsens. The fire is currently not contained and cover 2 acres of Area B)
- ◆ incident_cause (e.g., under investigation)

- ◆ acres (e.g., 2100)
- ◆ incident_status (e.g., true or false depending on containment percentage)
- ◆ incident_containment (e.g., 25 %)
- ◆ recorded_at (timestamp)
- ◆ last_updated_at (timestamp)

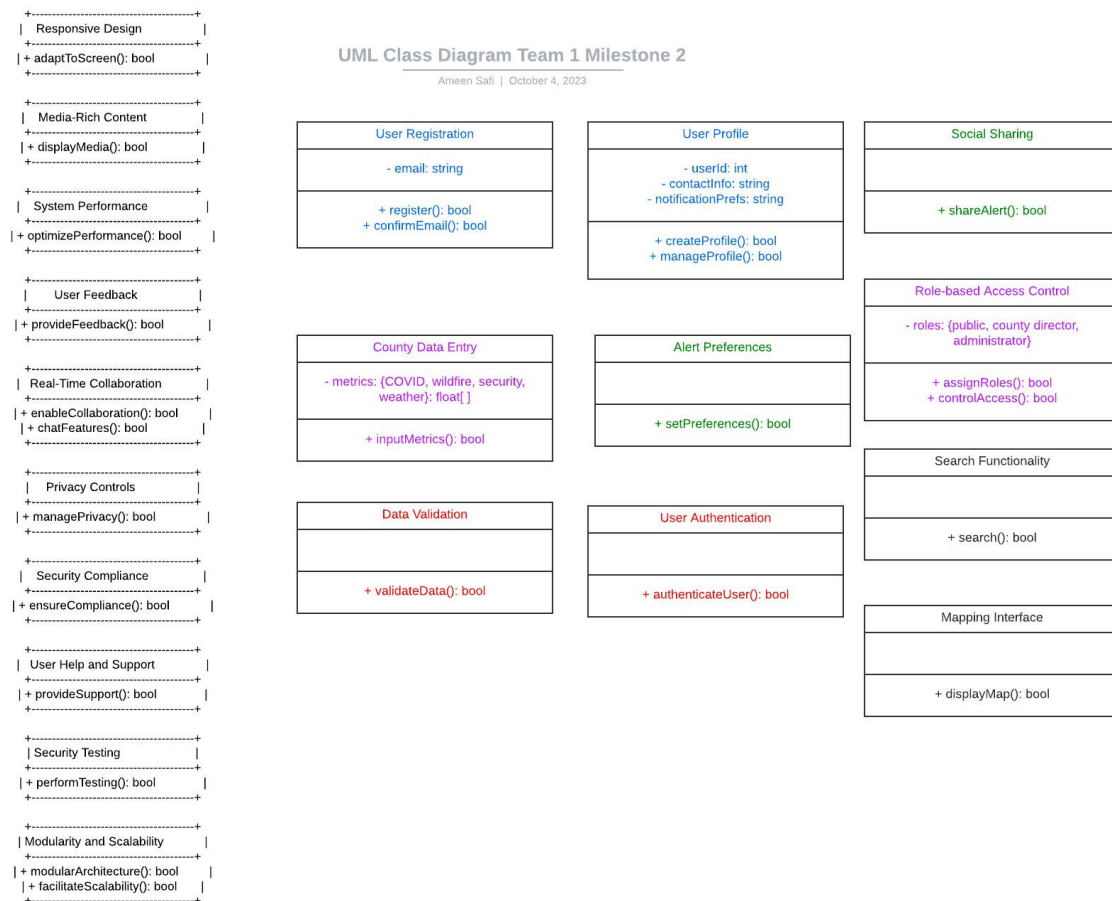
8. **Security_Metrics** - stores security metric for each county

- ◆ security_metric_id: unique identifier for each wildfire metric (key) (e.g., 1, 2, 3)
- ◆ county_id (foreign key)
- ◆ security_concern_type (e.g., protest, vandalism)
- ◆ security_action_description (e.g., concern abc reported at areas A and B, avoid areas A and B until cleared)
- ◆ recorded_at (timestamp)
- ◆ incident_status (e.g., true or false depending on whether area is cleared
Note: should be NN)

9. **Alerts** - stores all emergency alerts triggered by administrators

- ◆ alert_id: unique identifier for each alert (key) (e.g., 1, 2, 3)
- ◆ county_id (foreign key)
- ◆ triggered_by (Users foreign key)
- ◆ alert_message

UML Class Diagram (high level) - Ameen



Above is the high level UML class diagram for our product. The vertically listed left-hand portion of the image lists concepts for classes which aren't necessarily essential to the application, but would be useful to consider for optimization purposes. The right-hand side which contains actual UML style classes are high-level groupings which go hand-in-hand with the functional requirements detailed in section 1. Classes which would be grouped under various interfaces are colored to display relationships for what they fall under.

Blue text describes class information which would fall under some sort of interface which describes the user.

Purple coordinates with an interface describing the officials who add data to the platform.

Red will go under some form of interface which facilitates general authentication and validation.

Green coordinates with an alert interface.

Black coordinates (until further consideration) with standalone classes.

Media Storage - Decide if images and video/audio will be kept in the file systems or in DB BLOBs.

Describe any other special data format requirement like for video/audio/GPS

There are fields that store geographical information, specifically longitude and latitude coordinates for counties. These fields are used to represent locations on a map.

Search/filter architecture and implementation: what will be the alg/SW for search; how will you organize search items for the user; what DB terms will be searched, and how will be coded and organized in the DB

Search Algorithm or Software - team will utilize a text-based search engine/API(TBD)

Organization for search items - team will create a user-friendly search bar/ picklist where users can input search/select criteria and apply filters by alert type, county, and so on.

Database Terms to Search

- Alert Type (Covid, Weather, etc.)
- County
- Timestamp (to search for recently triggered alerts)

Your own APIs (if any): Describe and define at a high level the major APIs that you will create other than standard ones provided by tools and frameworks you use

Google map API (TBD)

Authenticate API (TBD)

API for County Data (TBD)

Search API (<https://developers.google.com/webmaster-tools>)

Describe any significant non-trivial algorithm or process if any (like rating, ranking, automatic prioritizing of items, etc.)

None

If you have changed SW tools and frameworks or added any new ones please describe it. Any new SW or framework you will be using has to be approved by the CTO in writing by this time.

None

5 Identification of Key Risks & Mitigation Strategies

1. Skills Risks: Our team might lack expertise in implementing specific functionalities such as real-time collaboration or advanced encryption techniques.

Resolution Strategy: To resolve the above issue, we will conduct Knowledge-sharing sessions, enabling collaborative problem-solving and enhancing skills among all members. Where necessary, we'll seek insights from knowledgeable team members or utilize on resources like Google.

2. Schedule Risks: To complete the project with all requirements on time will be challenging due to availability issues of the team members and some technical issues and some features which need to be implemented in the app will take more time.

Resolution Strategy: To resolve the above issue, we will be adopting agile SE process Regularly review project, prioritize some important asks and allocate extra hours or resources if necessary. We will be also doing weekly meeting other than class and also we will be in touch via Slack to keep everyone updated and on track.

3. Technical Risks: The main technical risk the team faces currently is connecting to the database through ssh. So far the team is only able to access the database using TCP/IP.

Resolution Strategy: Reviewing the logistics of the ssh instance. Making sure credentials

align and that the DB is trying to connect to the proper ssh. Also ensuring that the instance is active when trying to connect to the DB.

4. Teamwork Risks: Miscommunication or Differences in understanding among team members might lead to delays or incorrect implementations.

Resolution Strategy: To avoid this, we will Set up regular check-ins and updates and employ clear documentation and ensure open channels of communication.

5. Legal/content risk: using certain software or frameworks or necessary content with proper licenses and copyrights, leading to legal repercussions if not adhered to.

Resolution Strategy: To avoid this, we will ensure that all content and software integrated into the project is either created in-house, sourced from open-source platforms, or licensed correctly.

6 Project Management

For the efficient management and success of our "CalAwareNow - Digital Shield for California" project, we have integrated a systematic project management strategy anchored in collaboration, transparency, and adaptability. Our team operates with a clear hierarchy, ensuring each member understands their roles and responsibilities. The Team Lead is the main member of this structure, overseeing the project's evolution, keeping tasks in alignment with our objectives, and guaranteeing everyone remains informed. This is enhanced by regular meetings where we discuss progress, address issues, and share updates.

task management system, Trello will be used by us moving forward because it can provide a clear snapshot of all tasks, their status, and their priority. We will rank our work as "Must have," "Should have," and "Could have," allowing us to focus on essential features first. Our methodology also emphasizes feedback. Post any implementation, feedback is sourced and necessary changes made, reinforcing our commitment to a user-centric, iterative development model. As we progress, GitHub aids us in version control and issue tracking. In conclusion, by maintaining regular communication, and adhering to our prioritization strategy, we aim to ensure smooth progress and timely delivery of our project, "CalAwareNow - Digital Shield for California." We are committed to building an app that is not only up to the mark in terms of specifications but also delights our users.

