Bilkent University

Department of Computer Engineering

Senior Design Project

reporTown

Analysis Report

Arda Akça Büyük, Elif Özer, Cemre Biltekin, Oğuz Kaan İmamoğlu, Mustafa Yaşar

Supervisor: Dr. Ayşegül Dündar

Jury Members: Dr. Shervin Arashloo, Dr. Hamdi Dibeklioğlu

Innovation Expert: Mehmet Çakır

Nov 15, 2021

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Table of Contents

1. Introduction

Cities are home to many problems that can victimize and even sometimes endanger the citizens living in them. Road problems, garbage problems, transportation problems can be given as examples of those problems. When citizens encounter such problems, most of the time, they are unable to interfere in those problems individually. Therefore, these problems can only be resolved by institutions like the municipality, governorship, or non-governmental organizations. Citizens who encounter such situations can report these situations to the authorities through various channels. However, these tools cannot provide adequate solutions both during the reporting of the situation and during the follow-up of the situation. In addition, if the relevant problem does not create enough of an agenda, the authorities tend not to do what is necessary about the issue or take it slow. The fact that the addressee of the problem is not known about many problems also undermines the problem-solving process.

In some cases, instead of reaching out to the authorities, people may need help from other people and may want to find a volunteer to fix a problem. The feeding of stray animals can be given as an example of such cases.

It is difficult to report problems in cities, and following this process makes people desensitized about these problems. The lack of platforms where these problems can have a social impact also makes authorities insensitive. The fact that people do not have the opportunity to see the problems in the city and the locations of these problems also makes people's daily lives difficult and wastes their time. All this causes cities to become more and more neglected and more challenging to live in. The need for a project that can offer solutions to all of these problems seems obvious.

2. Current System

Although there are applications that promise solutions in this area, almost none of these applications focus on solving problems related to the city while having social media platform features [1]. Reporting apps designed by municipalities for citizens provide users with a one-sided communication opportunity between the municipality

and the citizen. Citizens cannot interact with each other or even find a volunteer for a problem. In addition, these applications are local and do not have the opportunity to be used globally. On the other hand, although social media platforms offer such an opportunity to users, the main focus of these platforms is not solving problems in people's environment.

3. Proposed System

3.1 Overview

The reporTown application aims to accelerate the process of resolving problems that require collaborative or authoritative cooperation while ensuring safety and security for its users. The proposed solutions will enable people who encounter a problem to report their problem via their mobile phones to the related authorities or request help from volunteers to reach the solution as fast as possible.

A feature that makes reporTown different from similar applications and innovative is that it enables institutions, authorities, aid agencies, and volunteers to work together to solve the problems of their cities. They can see what people need from the posts reported to their accounts or search for problems that require volunteers. When they observe such problems, they could find out the details of the problems such as what it is about and where it is, after that, they could contact the person who posted the problem to solve it efficiently. After they solve the problem, they notify the post owner, and if the owner confirms that the problem is resolved, the report will be classified as resolved. Authorities that resolve more issues will have more points. This point system enables authorities to race with their counterparts, increasing their motivations to resolve more issues. The point system not only applied to authorities but also citizens. Citizens can upvote posts of other citizens, and a citizen with many upvotes will have more points and become one step closer to being a model citizen. In addition, posts of citizens with high points are more prominent than other posts.

A person who encounters a problem takes a photograph of the problem, uploads it to the application by giving detailed information such as the description and location of the problem, and tags the institution or municipality that should solve the problem, or

as volunteering aid. Along with machine learning algorithms and computer vision, the application analyzes the problem to categorize it.

Users may not know which institute is responsible for the solution for some specific problems, and this situation decelerates the solution process. Again, by using machine learning, the application recommends interlocutors for problems to users.

Users and authorities will be able to see the problems on a map user interface. Depending on the frequency of problems reported in an area, nodes of different colors will appear on the map. Hot colors like red and orange mean that there are many problems reported in that area, and cold colors like blue and green mean that a few problems are reported in the area. By touching these nodes, users and authorities can view related posts. Thanks to this map, authorities can plan their actions wisely, and citizens can plan their days with these problems in their minds.

3.2 Functional Requirements

3.2.1 Initial Screen

- All users must be presented with the “Login” screen.
- A citizen or an institution that wants to register must be able to click the “Register” button to go to the “Register” screen in which there will be a form that includes some informational and credential fields.
- Any type of user that wants to log in must be able to enter their credentials (username, password) and click on the “Login” button.
- Any type of user that forgot their password must be able to click the “Forgot Password” link to go to the “Change Password” screen where they can change their password.

3.2.2 Register Screen

- A citizen must be able to click on the “As Citizen” on the Register screen to go to the “Citizen Registration” screen where they can enter their information and credentials.

- An institution must be able to click on the “As Institution” on the “Register” screen to go to the “Institution Registration” screen where they can enter their information and credentials and upload their registration document.

3.2.3 Citizen Registration

- A citizen must be able to register to reporTown by entering the credentials (name, surname, email, username, password) on the “Citizen Registration” screen.
- A citizen must be able to accept the terms and conditions through a checkbox.
- A citizen must be able to verify their email by entering the authentication code sent to their email, to be successfully registered.

3.2.4 Institution Registration

- An institution must be able to register to reporTown by entering the credentials (institution name, username, email, password) on the “Institution Registration” screen.
- An institution must be able to accept the terms and conditions through a checkbox.
- An institution must be able to verify itself by uploading a document that the admins will manually check. Otherwise, their registration will not proceed.
- An institution must verify their email by entering the authentication code sent to their email, to be successfully registered.

3.2.5 Institution Official Registration

- An institution must be able to register an official account by entering the account information and credentials (name, surname, email, username, password, position) that the official will have, on the “Add Employee” screen.
- An institution must be able to create a position for the official on the “Create Position” screen before registering them.

3.2.6 Login

- All user types must be able to log in with their accounts' credentials (username, password) if there are no currently logged-in users.

3.2.7 Changing Password

- All user types must be able to receive a change password link to their email by entering their email on the "Forgot Password" screen.
- All user types must be able to change their password on the screen that comes after clicking on the verification link.

3.2.8 Reporting a Problem

- A citizen must be able to go to the "Post Report" screen by clicking the "Post Report" button at the bottom of the screen.
- A citizen must be able to fill a form with the following fields: A problem's picture (*), description (*), category (*), responsible institution (*), and location information (*).
- After uploading the picture, a citizen must be able to select an institution that the application recommends.
- A citizen must be able to upload the report by clicking the "Post" button at the bottom of the form.

3.2.9 Citizen Profile Navigation

- A citizen must be able to go to their profile by clicking the "Profile" button at the right bottom of the screen.
- A citizen must be able to see their profile picture, username, reliability score, a short bio, and two sections as Resolved Problems and Unresolved Problems, including their reports.
- A citizen must be able to see their reports that have not been resolved yet by clicking the "Unresolved" button that appears on their profile.
- A citizen must be able to see their reports that have been resolved by clicking the "Resolved" button that appears on their profile.

3.2.10 Institution Profile Navigation

- An institution must be able to go to their profile by clicking the “Profile” button at the right bottom of the screen.
- An institution must be able to see their profile picture, username, reliability score, a short bio, two sections as Resolved Problems and Unresolved Problems, including their reports, and a section as Employees which contains the Officials’ accounts.
- An institution must be able to see their reports that have not been resolved yet by clicking the "Unresolved" button that appears on their profile.
- An institution must be able to see their reports that have been resolved by clicking the "Resolved" button that appears on their profile.

3.2.11 Notifications

- A user must be able to see their notifications by clicking the “Notifications” button on the top right of the screen.
- A user must be able to navigate to the related post that the notification is associated with by clicking on the notification item.

3.2.12 Report

- A user must be able to click a report and navigate to the “Report” screen, where there is a single report with all of its details (description, picture, category, responsible institution, location, upvotes, and comments).
- A user must be able to upvote the report.
- A user must be able to comment on the report.
- A user must be able to reply to a comment on the report.
- A user must be able to click on the “Upvotes” button under the report and see the citizens that upvoted the report.
- A user must be able to click on the “Comments” button under the report and see the comments of the citizens that commented on the report.
- A user must be able to see the replies to the comments by clicking on the “Replies” button on a comment.

- A user must be able to see the solution that the institution uploaded on the “Report” screen if the report is solved.
- A user must be able to report the post as spam.

3.2.13 Posting a Solution

- An official of the institution must be able to post a solution to a report by clicking on the “Post Solution” button on the “Report” screen and filling out a solution form with the following sections to inform the user: an explanation, a picture showing the problem is resolved.
- A report owner citizen must receive the "Problem Resolved" notification, and they must be able to click on this notification to go to the “Solution Approval” screen.
- A report owner citizen must be able to approve or reject a solution by clicking the “Approve” or “Reject” button on the “Solution Approval” screen.
- A problem must be visible at the “Resolved Problems” of both the user’s account and institution account if the user approves the solution.
- When the approval process is completed, both the scores of the user and the institution must be incremented by a certain amount according to the number of interactions that the report got.

3.2.14 Map

- A user must be able to click on the “Map” button on the bottom left side of the screen to navigate to the Map.
- A user must be presented with a map with a heatmap structure where there will be intense colors on the regions with high report density.
- A user must be able to navigate on the map by swiping the screen.
- A user must be able to zoom out and zoom in on the map.
- A user must be able to click on a particular region on the map to see the reports from that particular region.

3.2.15 Feed

- A user must be able to see a feed that will consist of local reports from that other users posted, just after logging in or after clicking the “Home” button at the bottom of the screen.
- A user must be able to filter reports by keyword, category, or location from the “Filter Reports” button at the top of the feed if they want to see the reports at that location.
- A user must be able to navigate to the “Report” screen by clicking on the reports on the feed.
- A user must be able to upvote the reports on the feed.
- A user must be able to report a post as spam.

3.2.16 Search

- A user must be able to search for a specific topic about the reports by clicking the “Search” button at the bottom of the screen.
- A user must be presented with the trending topics, which they can click and see the related posts about that topic, at the top of the screen under the search bar.
- A user must be able to search with a keyword for user profiles, institution profiles, and posts, on the search page.
- After the search operation is done, a user must be presented with all results that contain the entered keywords.
- A user must be able to navigate the search results by clicking on the “Citizens”, “Institutions”, or “Posts” button after the search operation.

3.2.17 Volunteering

- A user must be able to post a report with the category “Volunteering”.
- If a user or institution wants to participate in volunteering, they must be able to hit the “Joining” button to indicate that they are joining, which corresponds to the “Upvote” functionality in the other report categories.

3.2.18 Settings

- A user must be able to go to the “Settings” screen by clicking on the settings icon at the top left of the screen.
- A user must be able to go to the “Edit Profile”, “Change Language” screens, or log out from the application from that screen.

3.2.19 Profile Editing

- All user types must be able to change their profile picture and bio by clicking the “Edit Profile” button.
- An institution must be able to pin some resolved problems it would like to be seen to the top of its profile.

3.2.20 Changing Language

- A user must be able to change the language of the application by selecting a language from the language list on the screen.

3.2.21 Logging Out

- All types of users must be able to log out by clicking the “Logout” button on their profiles.

3.3 Nonfunctional Requirements

3.3.1 Usability

- Providing citizens with two different report tracking systems, a map and a feed, gives citizens the chance to choose the feature they want to use more, thus increasing the application's usability.
- Problems that receive high upvotes will appear at the beginning of the feed. In this way, citizens will be able to see more common problems in the upper ranks, and the traceability of the problems will increase.
- In the map feature, the problems will be displayed on a map, and citizens will be able to access the reports in the city by going to the desired location on the

map, which increases the usability by increasing the intensity of colors on the map where report frequency is high.

- The feature of uploading photos to the reports in the application gives the citizens the opportunity to express the problems better and enables the institutions and other citizens to understand the problem visually. In this way, usability is increased by providing clarity of the problem and ease of use.
- While creating the report, there will be a category recommendation with the help of location information and computer vision so that citizens can create reports conveniently.
- A citizen whose English and/or Turkish proficiency level is A1 or A2 (beginner-level) should be able to understand menu headings, read report titles, categories and descriptions, read institution, institution officials' and users' pages, and post their own reports. They might need a dictionary for complex words for the reporting process.
- All functional screens/pages can be reached from the main (home) page within four clicks.

3.3.2 Scalability

- reporTown is an application that a large number of people will use. Therefore, there will be a massive load in the system. To avoid system failures, a NoSQL database should be used to obtain scalability [3].
- For a more scalable environment, load balancers on a cloud environment should be utilized.

3.3.3 Security

- The user should receive a confirmation mail while registering to the system to prevent creating fake accounts by using other people's email accounts.
- Institutions have to provide an official document approved by the institution so that the system admins deny the malicious and fake institution accounts.

3.3.4 Reliability

- The application should be 99% reliable while creating a single report and should not crash.

- If the application crashes or the user quits abruptly, the worst-case data loss in the application is losing the unfinished report, which is not yet published and was in the process of being created.
- When users forget their passwords, they will be able to create a new password using the verification code sent to their email so that they don't use their accounts.
- Users will be able to change their passwords at any time from the settings section.

3.3.5 Performance

- The response time of each request should be less than 2 seconds.
- Verification emails should be sent within 60 seconds so that users will complete their registration process in 180 seconds.
- The application should load in less than 30 seconds when the number of users is greater than 2500.
- The photo upload should be done in less than 5 seconds while creating a report.

3.4 Pseudo Requirements

3.4.1 Implementation

- The application must run on Android and IOS mobile platforms.
- Git will be used as a version control system, and GitHub will be used for managing Git repositories and collaborative work for project management.
- React Native will be used as a mobile application framework to build the application as a cross-platform native app (for both Android and IOS).
- The application will be written in Javascript programming language for the frontend and Java for the backend.
- Spring Boot will be used to build the backend services of the application (API), providing a RESTful service and easing the object-oriented programming (OOP) process.
- MongoDB as a NoSQL database management system will be used for storing user information (including credentials, scores, and reports (including components like photographs, location).

- Google Maps API will be used for location services and displaying the map.
- Python programming language and PyTorch, Tensorflow, and Keras libraries will be used to provide the machine learning functionality.
- i18next internationalization-framework will be used for internationalization and localization to offer smart language support in English and Turkish.
- Lombok Java library increases maintainability without the need to write getter and setter methods.
- YOLO algorithm, COCO dataset and OpenCV library will be used for object detection.

3.5 System Models

3.5.1 Use Case Model

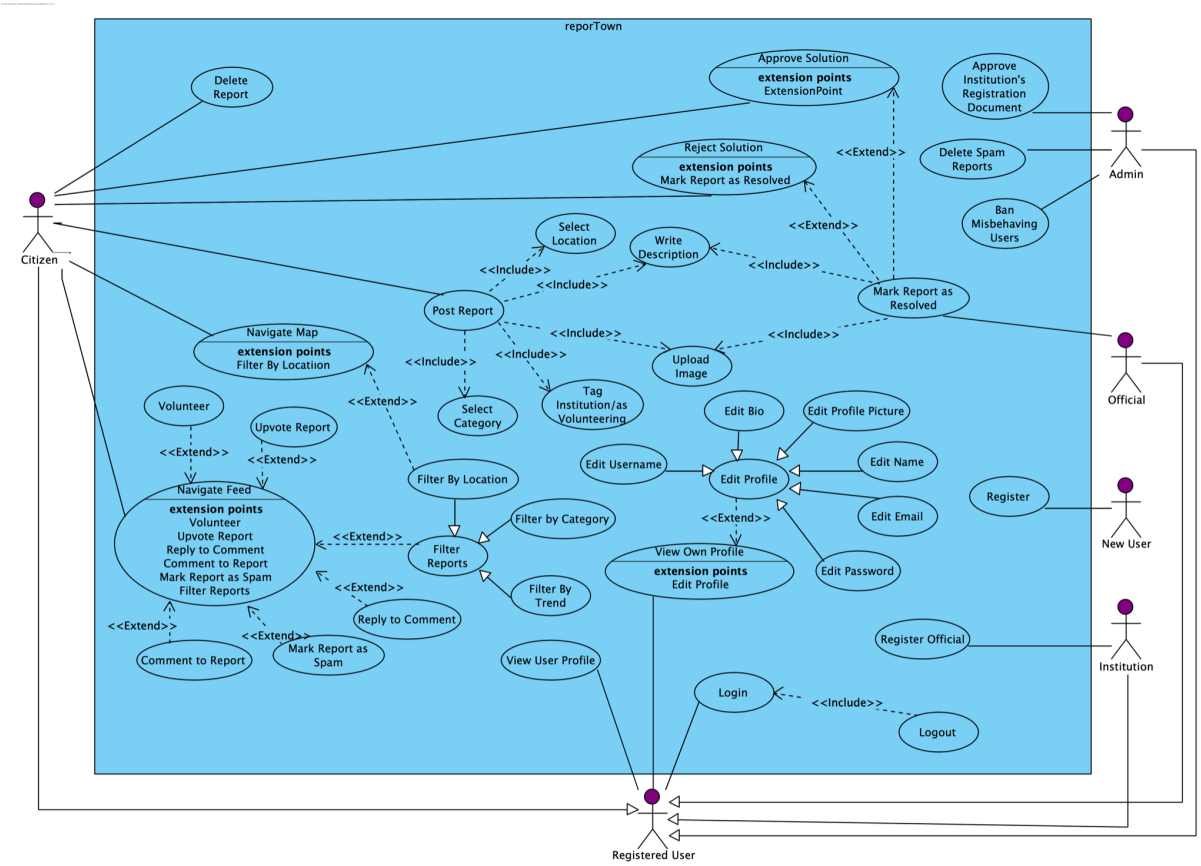


Figure 1. Use case model

3.5.2 Scenarios

3.5.2.1 Delete Report

a. Participating Actor

Citizen

b. Entry Condition

Citizen opens My Reports page.

Citizen has a report to be deleted.

c. Exit Condition

Citizen's report is deleted.

d. Flow of Events

Citizen opens My Reports page.

Citizen selects the report he/she wants to delete.

Citizen clicks the "Delete" button to delete the report.

Report is deleted.

e. Special/quality Requirements

The process should take no longer than 3 seconds.

3.5.2.2 Post Report

a. Participating Actor

Citizen

b. Entry Condition

Citizen should login to reporTown

c. Exit Condition

Report is created and posted to the feed and map.

d. Flow of Events

Citizen clicks **+** button visible below the feed to post a report.

Citizen uploads image(s) of the problem.

Citizen writes a description about the problem.

Citizen chooses a category for the problem domain.

Citizen enter the location information about where the problem occurs.

Citizen tags institution or volunteering.

- Citizen tags related institution to the problem if the report should be resolved by institution.

- Citizen choose Volunteering from the search bar of the tag if the report should be resolved by volunteers.

e. Special/quality Requirements

The process should take no longer than 10 seconds.

3.5.2.3 Navigate Map

a. Participating Actor

Citizen

b. Entry Condition

Citizen should be on the Map page.

c. Exit Condition

Citizen clicks the Feed button visible below the map to close the map.

d. Flow of Events

Citizen navigates on map through scrolling the map.

e. Special/quality Requirements

Navigation process should take no longer than 0.5 seconds.

3.5.2.4 Navigate Feed

a. Participating Actor

Citizen

b. Entry Condition

Citizen should be on the feed page.

c. Exit Condition

Citizen logout from the reporTown.

Citizen navigates map page.

d. Flow of Events

Citizen navigates on the feed through scrolling posts.

e. Special/quality Requirements

Navigation process should take no longer than 0.1 seconds.

3.5.2.5 Reject Solution

a. Participating Actor

Citizen, Official

b. Entry Condition

Official marks the report as resolved.

c. Exit Condition

Citizen clicks the “Decline Solution” button and solution is rejected.

d. Flow of Events

Report owner citizen receives the "Problem Resolved" notification.

Citizen opens the notification and controls the images indicating the problem is resolved.

Citizen clicks the “Decline Solution” button.

e. Special/quality Requirements

The process should take no longer than 5 seconds.

3.5.2.6 Approve Solution

a. Participating Actor

Citizen, Official

b. Entry Condition

Official marks the report as resolved.

c. Exit Condition

Citizen clicks the “Accept Solution” button and solution is approved.

d. Flow of Events

Report owner citizen receives the "Problem Resolved" notification.

Citizen opens the notification and controls the images indicating the problem is resolved.

Citizen clicks the “Accept Solution” button.

e. Special/quality Requirements

The process should take no longer than 5 seconds.

3.5.2.7 Select Location

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Post Report page.

c. Exit Condition

Citizen completes the location information.

d. Flow of Events

Citizen selects location in 2 ways:

- Citizen enters location information by clicking “Select Location” button.
- Citizen enter location information by “Use My Location” button if he/she wants to use his/her current location.

e. Special/quality Requirements

The process should take no longer than 0.1 seconds.

3.5.2.8 Write Description

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Post Report page.

c. Exit Condition

Citizen writes the description.

d. Flow of Events

Citizen explain the problem and its domain in the description text area.

e. Special/quality Requirements

The process should take no longer than 10 seconds.

3.5.2.9 Upload Image

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Post Report page.

c. Exit Condition

Image is uploaded to the system.

d. Flow of Events

Citizen clicks the Upload Image button.

Citizen can upload images in 2 ways.

- Citizen can take photos of the problem by clicking the “Camera” button.

- Citizen can upload images from the device by clicking the “Use Gallery” button.

e. Special/quality Requirements

The process should take no longer than 3 seconds.

3.5.2.10 Tag Institution/as Volunteering

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Post Report page.

c. Exit Condition

Citizen.

d. Flow of Events

Citizen searches for the institution's name by using search bar.

Citizen chooses the institutions from the results visible in the screen.

If the problem is resolved by volunteers, citizen searches for Volunteering.

e. Special/quality Requirements

The process should take no longer than 0.1 seconds.

3.5.2.11 Select Category

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Post Report page.

c. Exit Condition

Category is selected.

d. Flow of Events

Citizen chooses related category from the dropdown under the Category title.

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.5.2.12 Filter By Location

a. Participating Actor

Citizen

b. Entry Condition

A citizen is on the map page.

c. Exit Condition

Reports are filtered by their location.

d. Flow of Events

A citizen clicks the “Filter By Location” button located on top of the map.

A citizen selects the location from the dropdown.

A citizen clicks the “Apply” button.

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.5.2.13 Filter Reports

a. Participating Actor

Citizen

b. Entry Condition

A citizen is on the feed.

c. Exit Condition

Reports are filtered by desired option.

d. Flow of Events

A citizen clicks the “Filter Reports” button located on top of the feed.

A citizen selects the location from dropdown if she/he wants to filter reports by location.

A citizen selects the category from the dropdown if she/he wants to filter reports by category.

A citizen selects the trend if she/he wants to filter reports by trend.

A citizen selects the trend,category and location if she/he wants to filter reports for all three.

A citizen clicks the “Apply” button.

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.5.2.14 Filter By Category

a. Participating Actor

Citizen

b. Entry Condition

A citizen is on the feed.

c. Exit Condition

Reports are filtered by category.

d. Flow of Events

A citizen clicks the “Filter Reports” button located on top of the feed.

A citizen selects the category from dropdown.

A citizen clicks the “Apply” button.

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.5.2.15 Filter By Trend

a. Participating Actor

Citizen

b. Entry Condition

A citizen is on the feed.

c. Exit Condition

Reports are filtered by trend.

d. Flow of Events

A citizen clicks the “Filter Reports” button located on top of the feed.

A citizen selects the trend.

A citizen clicks the “Apply” button.

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.5.2.16 Volunteer

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Feed or Report page.

c. Exit Condition

Number of volunteers is incremented by 1.

d. Flow of Events

The citizen clicks the volunteer button visible under the post.

e. Special/quality Requirements

The process should take no longer than 0.2 seconds.

3.5.2.17 Upvote Report

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Feed or Report page.

c. Exit Condition

Number of upvotes is incremented by 1.

d. Flow of Events

A Citizen clicks the upvote button visible under the post.

Number of upvotes is incremented by 1.

e. Special/quality Requirements

The process should take no longer than 0.2 seconds.

3.5.2.18 Reply to Comment

a. Participating Actor

Citizen

b. Entry Condition

A Citizen is on the Comments page.

c. Exit Condition

Comment is replied.

d. Flow of Events

A Citizen clicks the “Reply” button visible below the comments.

A Citizen writes his/her reply to the comment.

A Citizen clicks “Send Reply” button to post his/her reply.

e. Special/quality Requirements

The process should take no longer than 5 seconds.

3.5.2.19 Mark Report as Spam

a. Participating Actor

Citizen

b. Entry Condition

A Citizen is on the Feed.

c. Exit Condition

Report is marked as spam by citizen.

d. Flow of Events

A Citizen clicks “...” button to mark a report as spam.

e. Special/quality Requirements

The process should take no longer than 0.2 seconds.

3.5.2.20 Comment to Report

a. Participating Actor

Citizen

b. Entry Condition

Citizen is on the Feed.

c. Exit Condition

Comment is visible under the commented report.

d. Flow of Events

A Citizen clicks the comment icon under the report.

A Citizen writes the comment and clicks the “Comment” button.

Comment is sent.

e. Special/quality Requirements

The process should take no longer than 5 seconds.

3.5.2.21 Mark as Resolved

a. Participating Actor

Official

b. Entry Condition

Report is resolved by the institution.

c. Exit Condition

Report is marked as resolved.

d. Flow of Events

An Official selects the report to be marked as resolved.

An Official navigates to “Post Solution” page.

An Official upload images indicating the problem is resolved.

An Official write description about the solution.

An Official clicks the “Post” button to notify the user.

e. Special/quality Requirements

The process should take no longer than 5 seconds.

3.5.2.22 Approve Institution’s Official Document

a. Participating Actor

Admin

b. Entry Condition

An Institution uploads an official document to register to reporTown.

c. Exit Condition

An Institution’s official document is approved.

d. Flow of Events

An admin navigates to the pending institution registrations page.

An admin selects the institution from the pendings by clicking the institution name and opens the official document.

An admin clicks the “Approve Register” button.

e. Special/quality Requirements

This process should take no longer than 5 seconds.

3.5.2.23 Delete Spam Reports

a. Participating Actor

Admin

b. Entry Condition

A Citizen marks a report as spam.

c. Exit Condition

Spam report is removed from the reporTown**.**

d. Flow of Events

An admin navigates to the spam reports page.

An admin selects the marked as spam reports from the list by clicking the report.

An admin clicks the “Remove Report” button to remove spam report from the reportTown.

e. Special/quality Requirements

This process should take no longer than 5 seconds.

3.5.2.24 Ban Misbehaving Users

a. Participating Actor

Admin

b. Entry Condition

An admin identifies a misbehave of a user.

c. Exit Condition

The user is removed from the platform.

d. Flow of Events

An admin decides the misbehave as intolerable.

An admin removes the user and all of their activity from the database.

e. Special/quality Requirements

This process should take no longer than 15 seconds.

3.5.2.25 Register

a. Participating Actor

New User

b. Entry Condition

A New User clicks the register button

c. Exit Condition

A New User is registered to the system.

d. Flow of Events

A New User clicks “As Citizen” button if she/he wants to register as a citizen.

- A New User fills the name, surname, username, email address, and passwords fields.
- A New User clicks the Register button.

A New User clicks “As Institution” button if it wants to register as an institution.

- A New User fills the institution name,city, country, username, email address, and passwords fields.
- A New User uploads a document showing that they are an official institution.
- A New User clicks the Register button.

e. Special/quality Requirements

This process should take no longer than 30 seconds.

3.2.5.26 Register Official

a. Participating Actor

Institution

b. Entry Condition

The institution is on the Add Employee page.

c. Exit Condition

A new official is registered in the system.

d. Flow of Events

An institution fills in the information about the official and enters the credentials of the official.

An institution clicks on the “Add Employee” button.

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.2.5.27 View User Profile

a. Participating Actor

Registered User

b. Entry Condition

A Registered User logged into the system

c. Exit Condition

A Registered User views the user profile.

d. Flow of Events

A Registered User searches user by entering name to the search bar located top of the feed.

A Registered User clicks the user that he/she searches from the results coming from the search bar.

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.7.5.28 View Own Profile

a. Participating Actor

A Registered User

b. Entry Condition

A Registered User logs in to the reporTown

c. Exit Condition

A Registered User navigates to its profile.

d. Flow of Events

A Registered User clicks profile icon located below the feed

e. Special/quality Requirements

The process should take no longer than 0.5 seconds.

3.7.5.29 Edit Profile

a. Participating Actor

Registered User

b. Entry Condition

A Registered User is on the profile page.

c. Exit Condition

A Registered User's profile is updated.

d. Flow of Events

A Registered User clicks the "Settings" button located top-left corner.

A Registered User clicks the "Edit Profile" button.

A Registered User can edit the username, bio, name, email and password.

A Registered User clicks "Save Changes"

e. Special/quality Requirements

The process should take no longer than 10 seconds.

3.7.5.30 Edit Username

a. Participating Actor

Registered User

b. Entry Condition

A Registered User is on the Edit Profile page.

c. Exit Condition

A Registered User's username is updated.

d. Flow of Events

A Registered User enters the new username to the username field.

A Registered User clicks "Save Changes"

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.7.5.31 Edit Bio

a. Participating Actor

Registered User

b. Entry Condition

A Registered User is on the Edit Profile page.

c. Exit Condition

A Registered User's bio is updated.

d. Flow of Events

A registered user writes the new bio to the bio field.

A Registered User clicks "Save Changes"

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.7.5.32 Edit Profile Picture

a. Participating Actor

Registered User

b. Entry Condition

A Registered User is on the Edit Profile page.

c. Exit Condition

A Registered User's profile picture is updated.

d. Flow of Events

A registered clicks the + button located near the profile picture.

A Registered User can upload new profile picture in 2 ways.

- Citizen can take photos by clicking the "Camera" button.
- Citizen can upload pictures from the device by clicking the "Use Gallery" button.

A Registered User clicks "Save Changes"

e. Special/quality Requirements

The process should take no longer than 3 seconds.

3.7.5.33 Edit Name

a. Participating Actor

Registered User

b. Entry Condition

A Registered User is on the Edit Profile page.

c. Exit Condition

A Registered User's name is updated.

d. Flow of Events

A registered user enters the new name to the name field.

A Registered User clicks "Save Changes"

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.7.5.34 Edit Email

a. Participating Actor

Registered User

b. Entry Condition

A Registered User is on the Edit Profile page.

c. Exit Condition

A Registered User's email is updated.

d. Flow of Events

A registered user enters the new email to the email field.

A Registered User clicks "Save Changes"

e. Special/quality Requirements

The process should take no longer than 2 seconds.

3.7.5.35 Edit Password

a. Participating Actor

Registered User

b. Entry Condition

A Registered User is on the Edit Profile page.

c. Exit Condition

A Registered User's password is updated.

d. Flow of Events

A Registered User clicks the "Change Password" button and navigates to "Changes Password" screen.

A Registered User enter the old password, new password and confirmation of new password to the related fields.

A Registered User clicks the “Add” button and navigates to the Edit Profile page again.

A Registered User clicks “Save Changes”

e. Special/quality Requirements

The process should take no longer than 5 seconds.

3.5.3 Object and Class Model

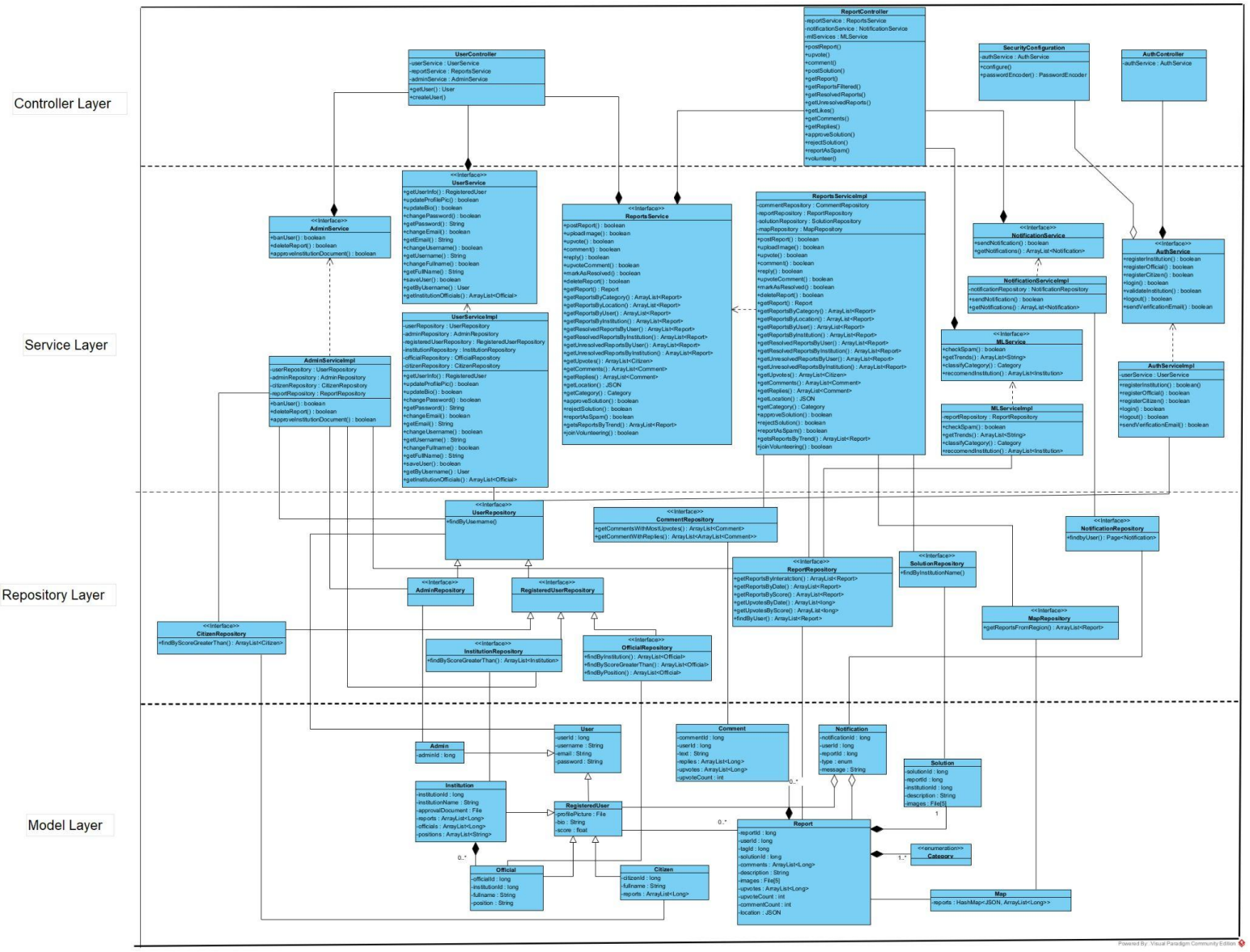


Figure 2. Object and class model

3.5.4 Dynamic Models

3.5.4.1 Sequence Diagrams

3.5.4.1.1 Sequence diagram of an institution registering to the system

Scenario: Institution wants to register to the system. Institution fills credentials and uploads an official verification document. Admin examines the registration request and either rejects or accepts it. Admin sends a registration verification email to the institution.

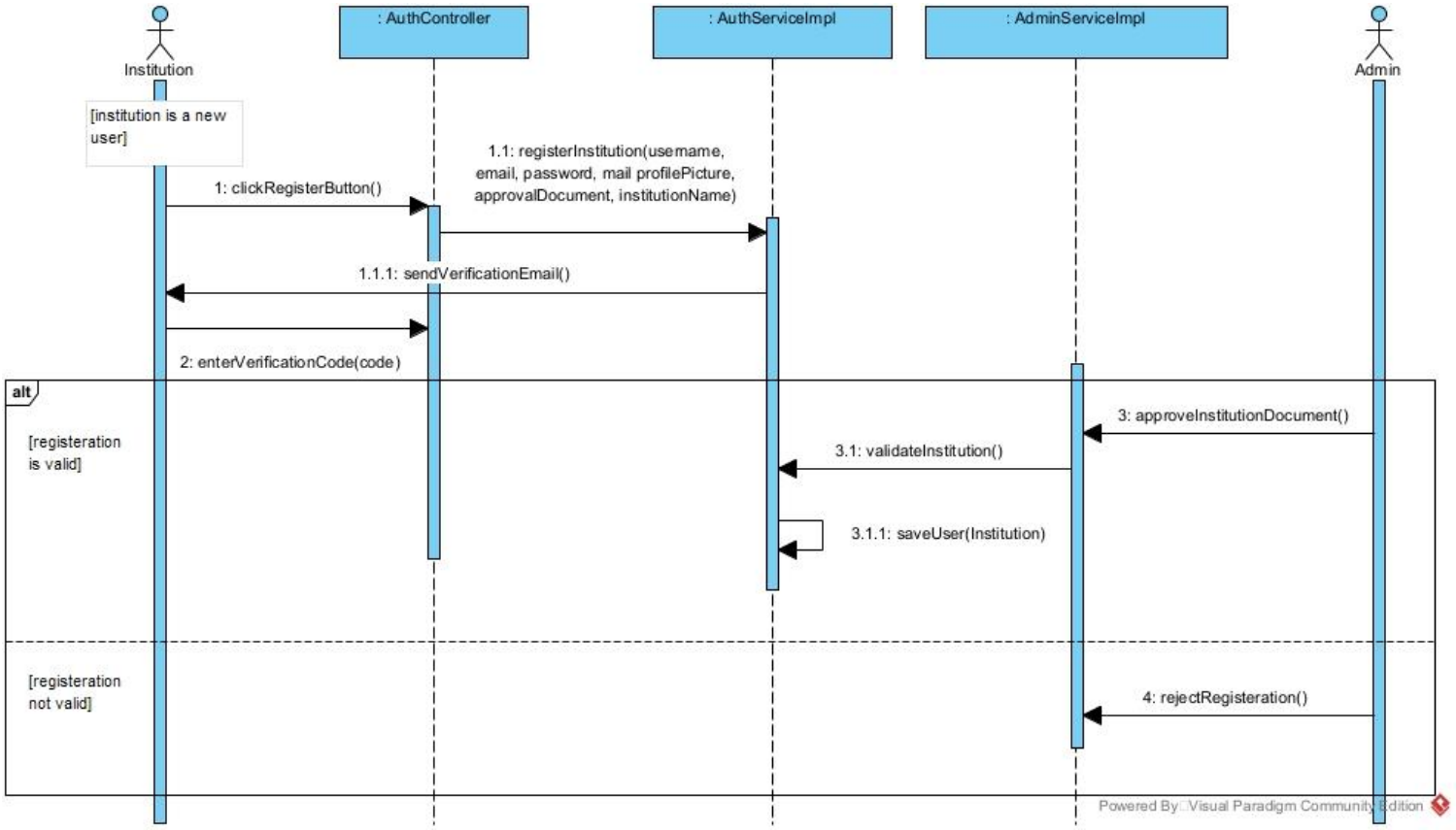


Figure 3. Sequence diagram of an institution registering to the system

3.5.4.1.2 Sequence diagram of a user logging in to the system

Scenario: Registered User enters username and password for logging in. If the credentials are valid, the user successfully logs in. Otherwise, the system throws exceptions. Should the user want to choose Forgot My Password option, the user

receives a password reset email which directs the user to the Reset Password Screen.

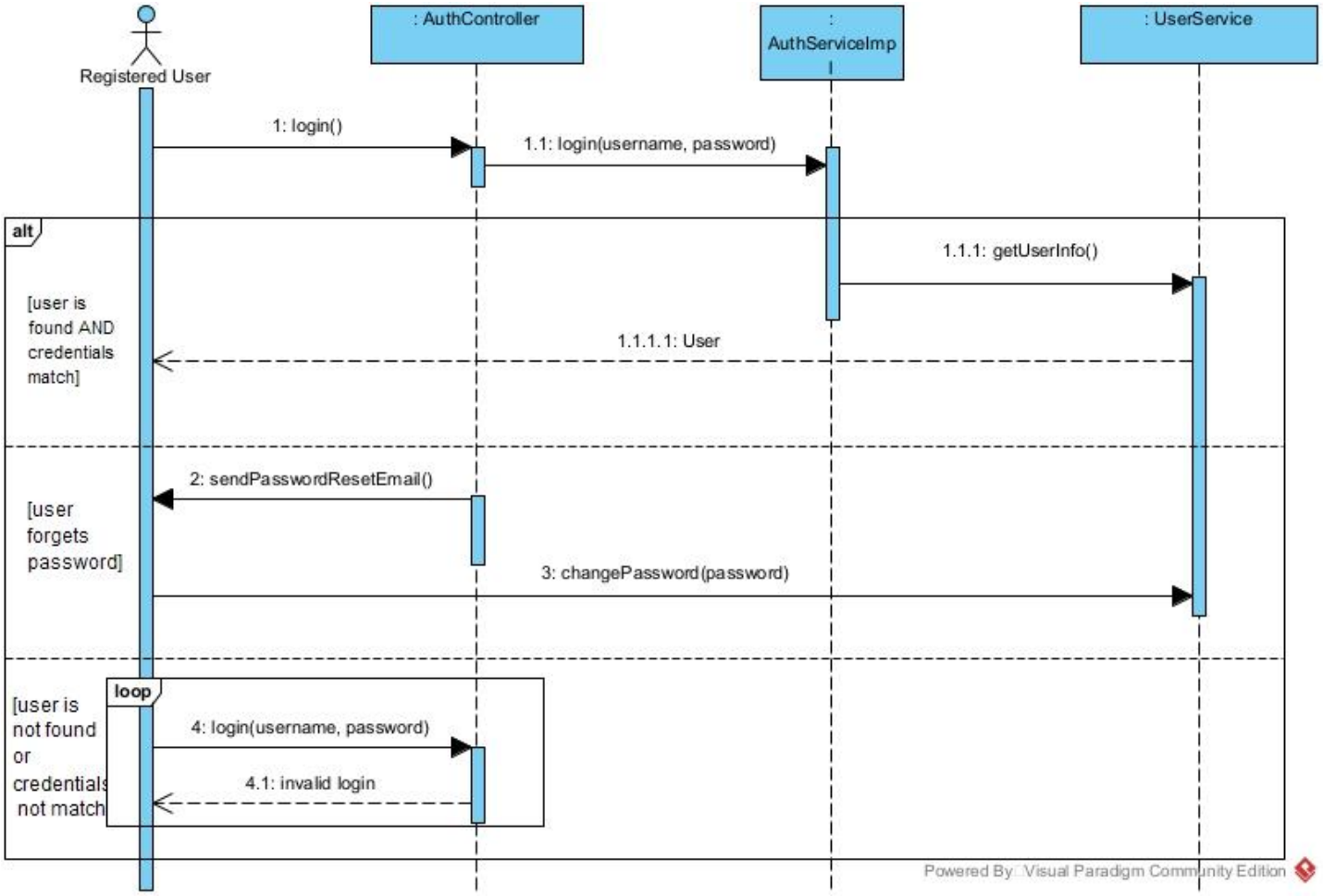


Figure 4. Sequence diagram of a user logging in to the system

3.5.4.2 State Machine Diagrams

3.5.4.2.1 States of the posted report

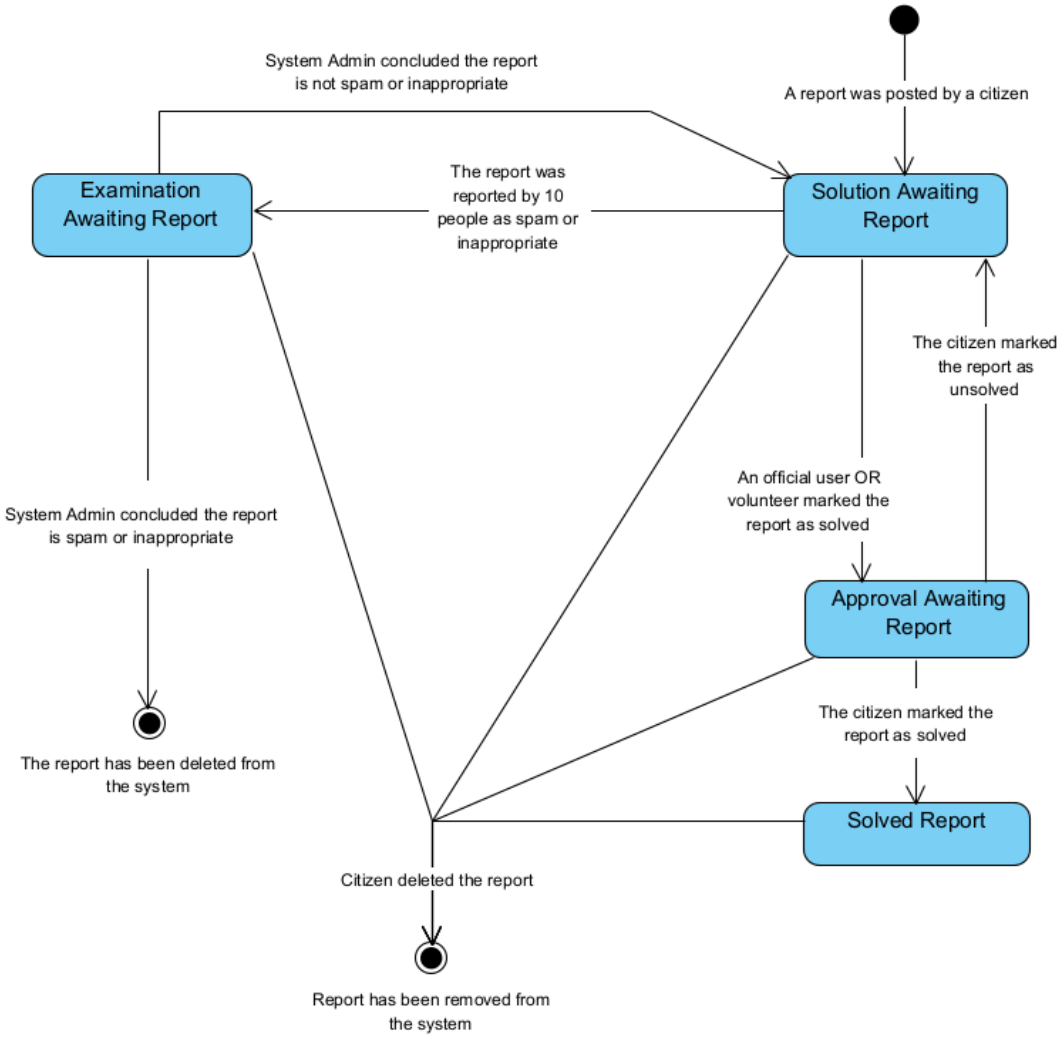


Figure 5. States of the posted report

The diagram above shows the different major states of a report that was posted by a citizen. The initial state is the “Solution Awaiting Report” state that indicates the report is not marked as solved, can be commented, and can be reported as spam or inappropriate. The report could be marked as solved by volunteers or official users which moves the report into “Approval Awaiting Report”. The citizen who posted the report could approve the solution which makes the report’s state “Solved Report”, or reject the solution which makes the report’s state “Solution Awaiting Report” again. If a report is found to be inappropriate or spam by 10 people and reported, the report moves into the “Examination Awaiting Report” in which the System Admin’s decision

is needed. For instance, if a report's description contains inappropriate sentences, hate speech, or image, it can be reported. The System Admin is able to delete the report from the system if found inappropriate or spam, and reject the reports which makes the report's state go into the "Solution Awaiting Report".

3.5.4.2.2 States of the organizations

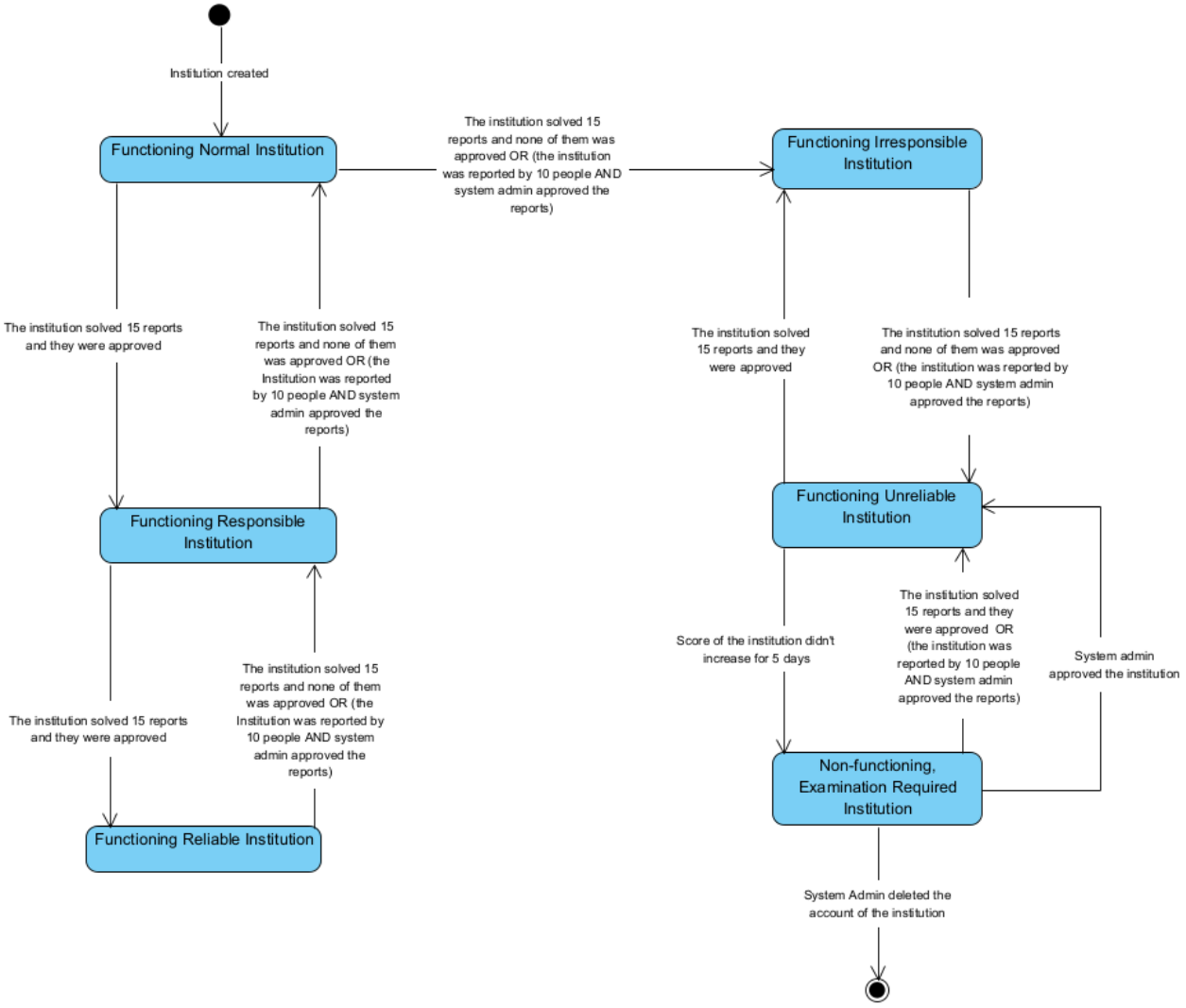


Figure 6. States of the organizations

The diagram above gives information on possible states of an organization that was registered into the system. The current state of the organization is determined by scores gained or lost by the organization. Organizations can gain scores when they solve reports and these solutions are approved by the citizens who posted the

reports. On the other hand, they can lose scores if their proposed solutions are rejected by the citizens or their accounts were reported by people and those reports were approved by admins. By solving 15 reports successfully, the state of the institution transitions into Responsible and Reliable respectively. Conversely, it can become Irresponsible and Unreliable if the score of the institution decreases. If the state of the institution is Unreliable and the score of the institution did not increase for 5 days, the state of the institution transitions into Non-functioning, Examination Required. The system freezes the account of the institution and a System Admin decides whether or not the institution will be functioning. The admin is able to delete the account of the institution from the system and approve the institution so that it can continue functioning again.

3.5.4.3 Activity Diagrams

3.5.4.3.1 Posting a report by citizen activity diagram

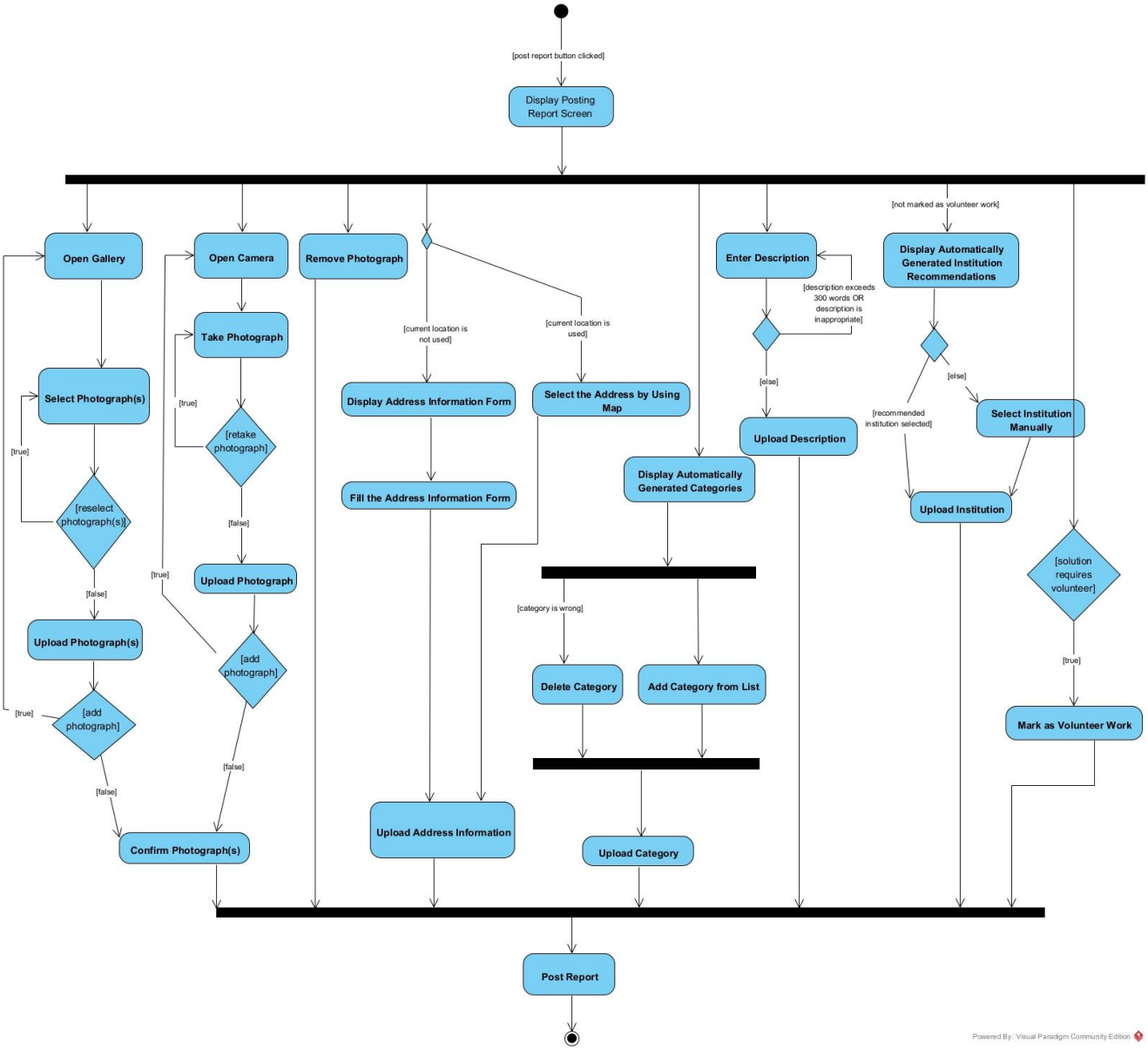


Figure 7. Posting a report by citizen activity diagram

When the citizen clicks to the “post report” button, the application displays the posting report screen in which the citizen must provide details about the issue. The citizen can freely fill the required fields (photograph, address, category, description, and institution that should solve the report).

To upload a photograph, the citizen can select photographs from their gallery as well as they can use their cameras to take a photograph. The citizen can at most upload

5 photographs to the system for their report. In order to select a photograph from a gallery, the citizen must enable the application to access their gallery. Similarly, camera access by the application must be enabled as well.

In order to fill the location information that will be used by the official users or volunteers who decide to solve the report, the citizen can either fill the address information form or they can use the location service of their cellphones. To be able to use the location service, the access permission must be given to the application by the citizen.

When the citizen uploads photograph(s) and description, the system automatically generates related categories that the citizen can choose to add to their report. They can delete a category that was automatically generated, or add a category by themselves from the list to the report.

The description explains the problem in detail but it is not mandatory to add a description to the report. However, adding details to the description is encouraged since it may accelerate the solution attempts. Unless the description exceeds 300 words, the citizen can upload the description.

When the citizen uploads photograph(s), description, and categories, the system automatically generates institution recommendations that can solve the problem. Citizens can choose the institution that should solve the problem from the generated list of institutions as well as they can choose an institution manually.

If the solution requires a volunteer, the report can be marked as volunteer work so that volunteers can see the report and try to solve the issue.

3.5.4.3.2 Institution registration

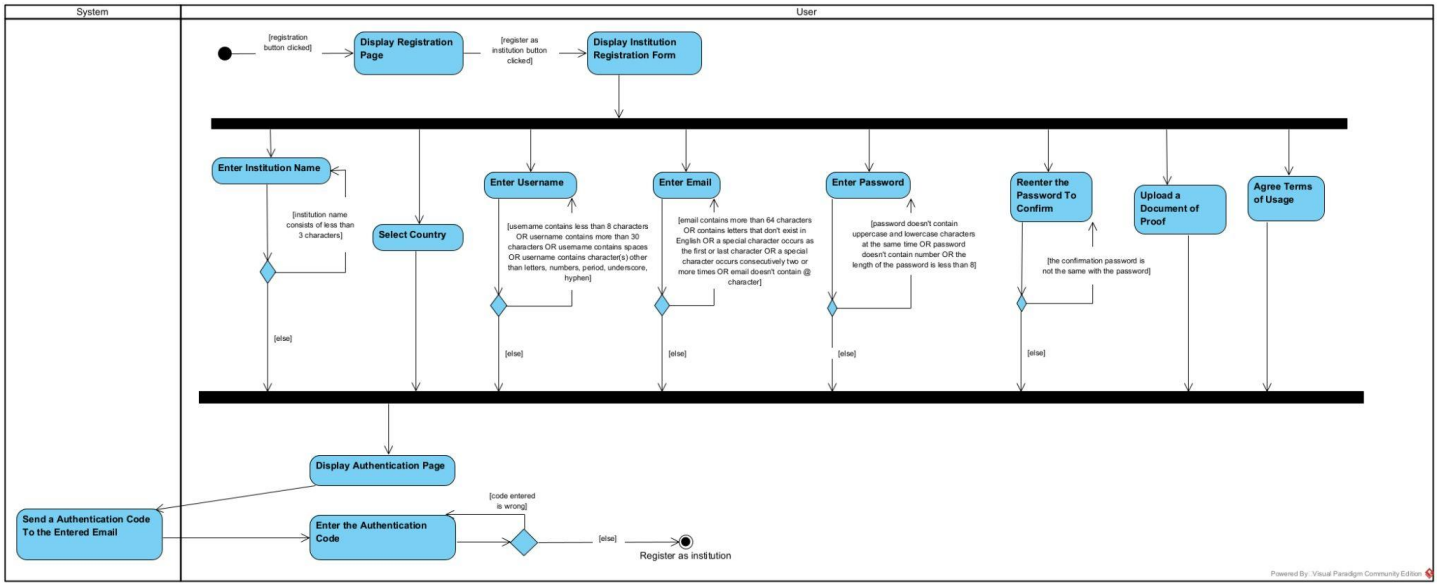


Figure 8. Institution registration activity diagram

In order to register as an institution, one must click the “register as institution“ button in the application. When they click, the institution registration form is displayed.

To be able to register, every field must be filled with proper information such as institution name, country, username, email, password, and document of proof.

The institution name will be displayed by everyone using the application and it must contain at least 3 characters.

The country must be selected from the list that the system proposes.

The username will be used to login to the institution account and it must contain at least 8 characters, at most 30 characters. The username can’t contain spaces and characters other than letters, numbers, period, underscore, and hyphen.

The email can also be used while logging in and it must contain at most 64 characters. The email can’t contain letters that don’t exist in English. A special

character can’t be the first or last character of the email, a special character can’t occur consecutively two or more times.

The password must contain uppercase and lowercase characters as well as numbers. The length of the password must be at least 8.

While registering, the institution must upload a document that can prove the person registering into the application belongs to the institution. As an example, this document can be an official document containing a stamp of the institution.

The person registering to the application must read and comprehend the terms of usage and approve it by checking the checkbox of terms of usage.

After satisfying the requirements for information fields, when the register button is clicked, the system generates an authentication code and sends it to the email that was written in the email field. This code must be entered properly to request registration.

After entering the authentication code correctly, the admin’s approval is needed to conclude the registration. This activity has been explained in the following section (3.5.4.3.3).

3.5.4.3.3 Institution registration approval by Admin

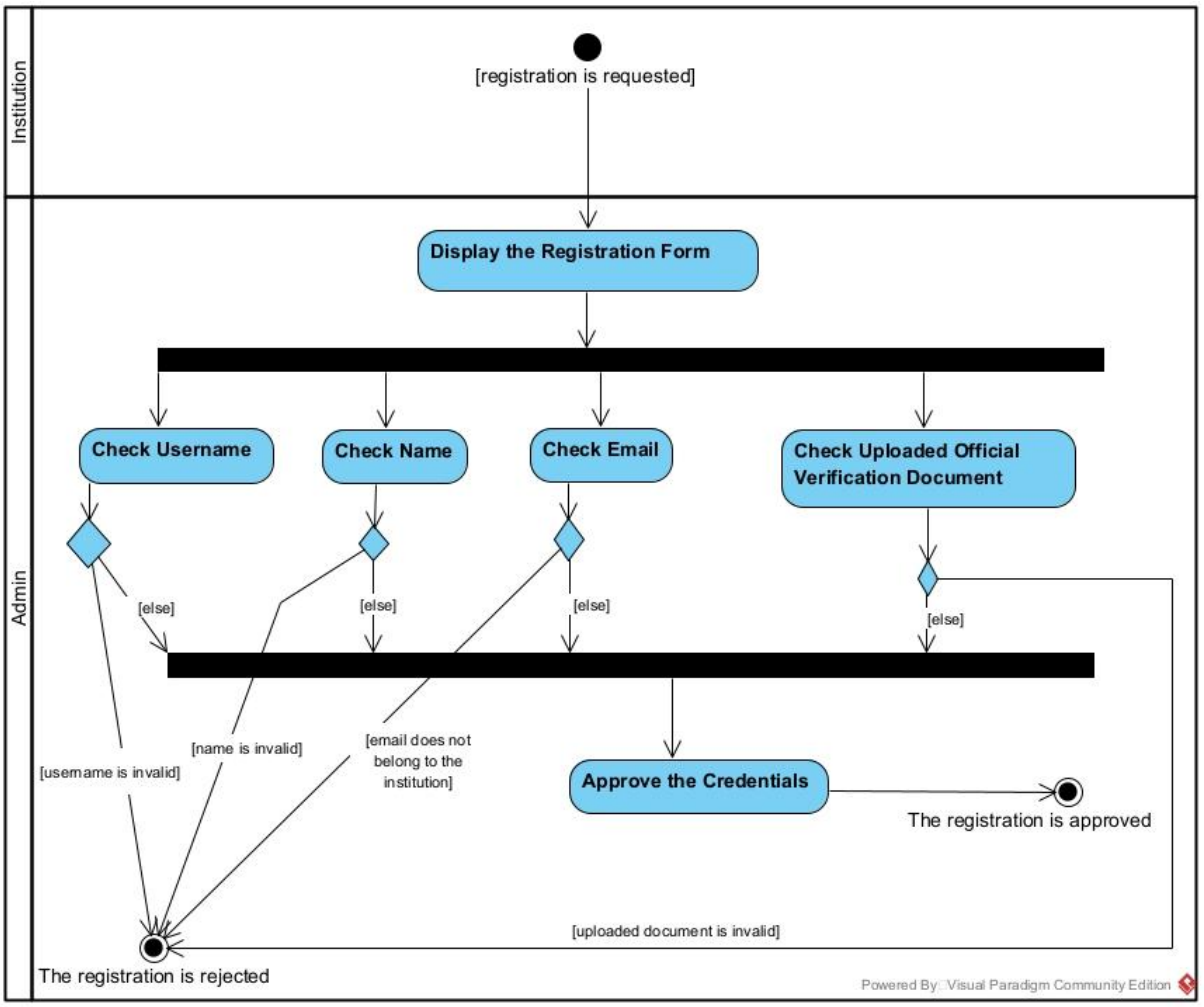


Figure 9. Institution registration approval by Admin activity diagram

When an institution fills every information in the registration page and verifies their account by entering the code that was sent to their email, a request is generated and sent to the system admin. The admin reviews the information that was entered such as username, name, email, and official verification document uploaded by the institution. The admin is able to reject the registration which prevents the institution from using the application, and approve the registration so that the institution can use the application.

3.5.4.3.4 Official user creation by Institution

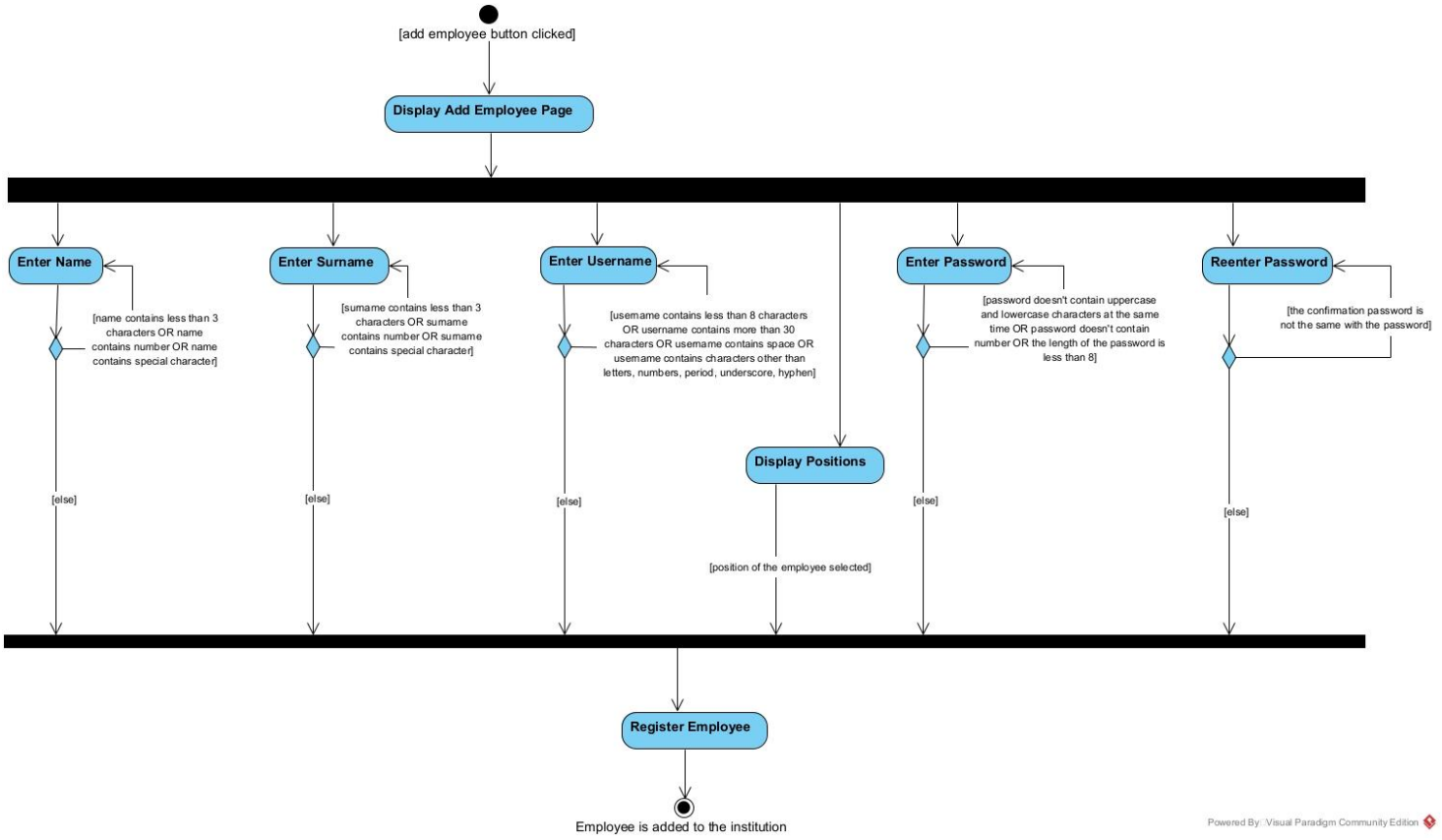


Figure 10. Official user creation by Institution

In order to register into the application as an official user, the institution that the official user will belong to must create his/her account.

To register properly, every information of the official user must be entered such as name, surname, username, password, and position.

The name mustn't contain any numbers or special characters and must contain at least 3 characters.

The surname mustn't contain any numbers or special characters and must contain at least 3 characters.

The username must contain at least 8 characters and at most 30 characters. It can't contain space or characters other than letters, numbers, period, underscore, or hyphen.

Position of the official user must be selected from the positions list that was created by the institution.

Password must contain uppercase characters, lowercase characters, number, and the length of it must be at least 8.

After satisfying requirements, the account of the official user can be created by the institution.

3.5.4.3.5 Feed actions

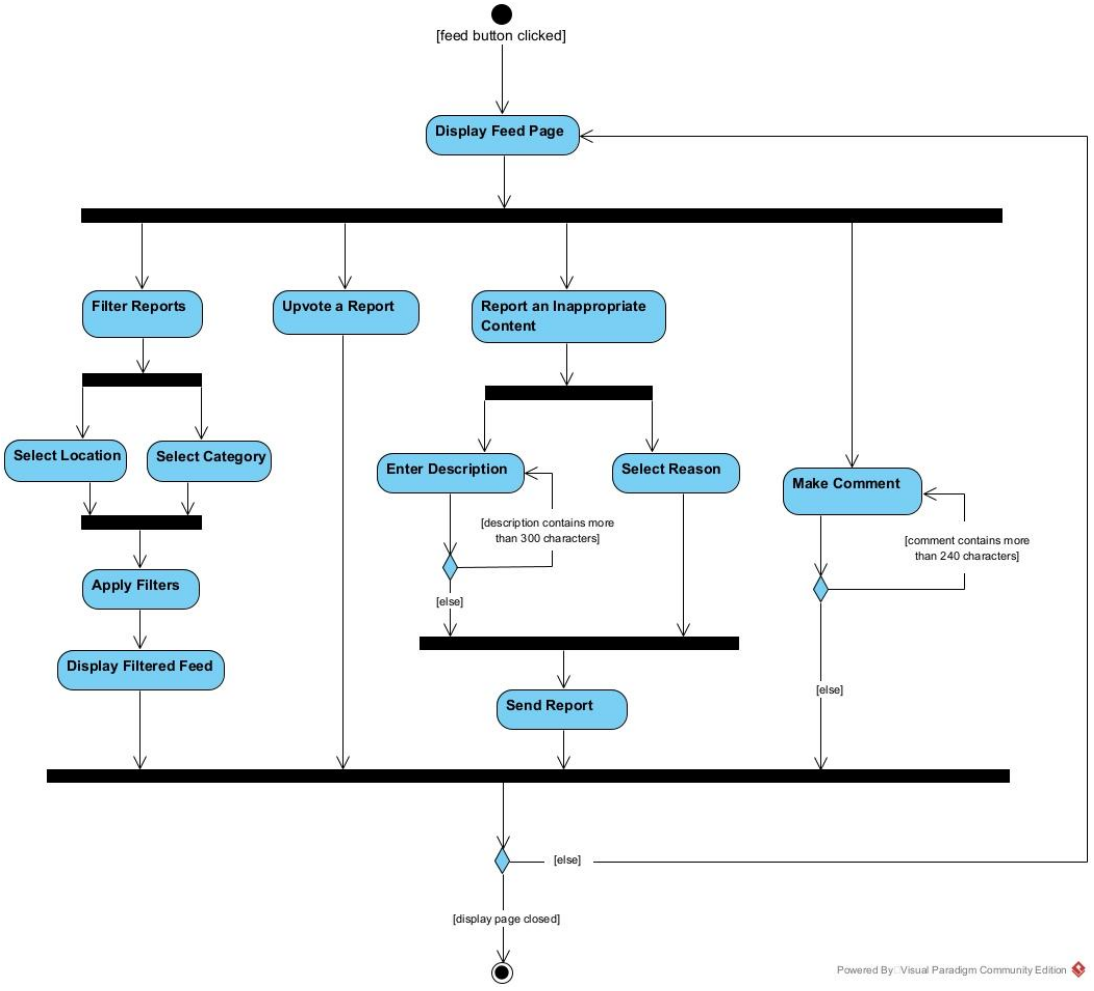


Figure 11. Official user creation by Institution

Every user of the application (official users, citizens, and institutions) has a feed containing reports posted and displayed according to their locations. The activity diagram above gives information about the activities that can be done while displaying the feed.

When the users click the feed button in the navigation bar at the bottom of the screen, the feed page is displayed.

In order to filter the reports displayed in the feed page, the users can use filtering by clicking the apply filter button. They can filter the reports according to their location and categories. The system displays the locations and categories that can be selected to filter and user selects as they desire from the choices. After choosing, they can apply filters and the feed is displayed according to the filters.

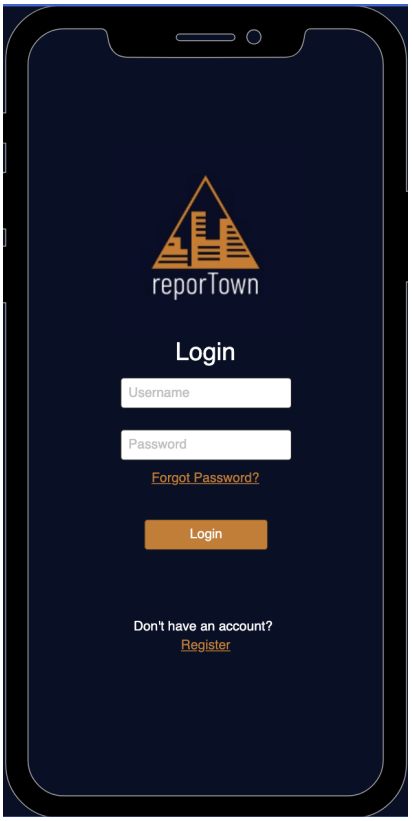
Users can upvote reports so that these reports will be seen as more important and noticed by more people using the application. Users can achieve this by clicking the upvote button in the report.

In case the users encounters an inappropriate or a spam report, they can easily report this report by clicking the mark as inappropriate/spam button at the top right corner of the report. They must select one of the reasons the system presents and they can enter a description which will facilitate the judgment made by system admins. The entered description must be less than 300 characters.

Users can make comments to the reports by clicking the text area and writing their comments, and clicking the send button. The comment made must contain at most 240 characters.

3.5.5 User Interface - Navigational Paths and Screen Mock-ups

3.5.5.1 Login

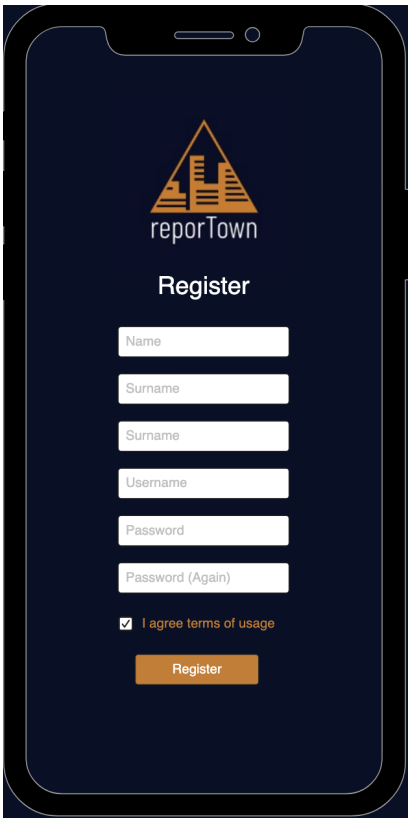


Users can log in by entering their usernames and passwords. If they do not have accounts, they can navigate to the registration page. Also, they can navigate to the password renewal page if they forget their passwords.

3.5.5.2 Register

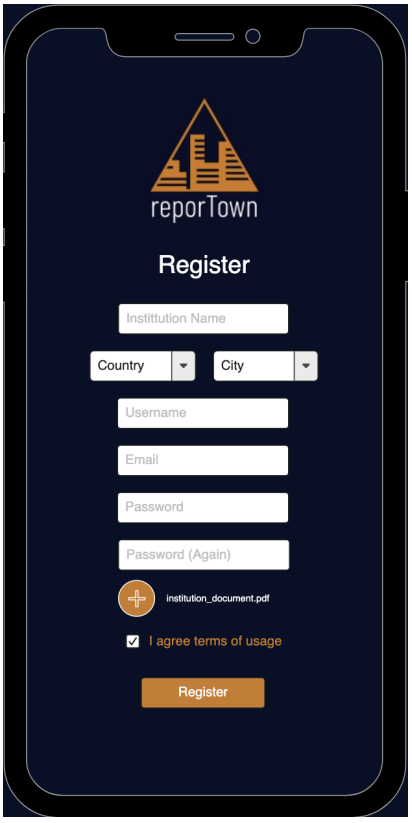
Users must choose whether they will register as a citizen or as an institution.

3.5.5.3 Citizen Register



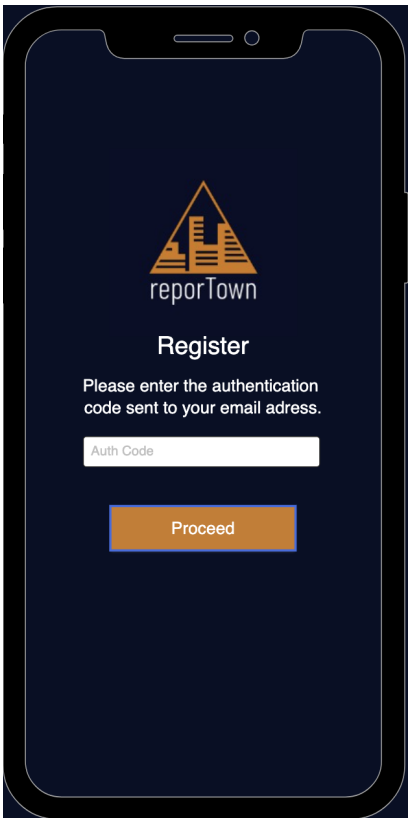
Citizens must enter their names, surnames, usernames, email addresses, and passwords. Also, they should agree to terms of usage.

3.5.5.4 Institution Register



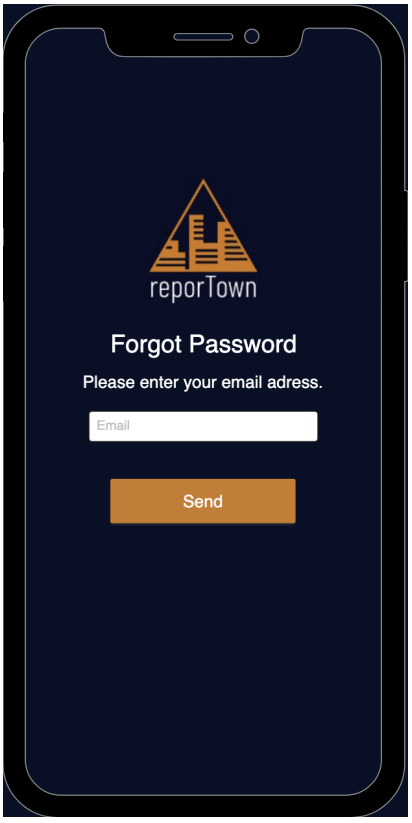
In addition to citizen registration, institutions also must choose their location and authenticate themselves by uploading an official document.

3.5.5.5 Email code verification

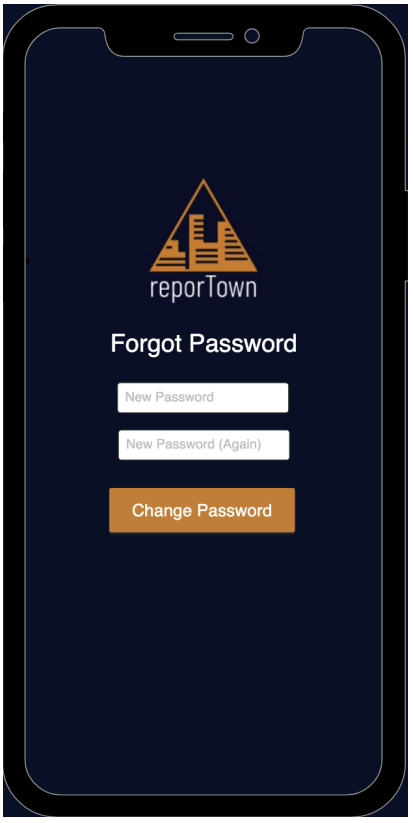


Before completing the registration process, users must verify their email addresses.

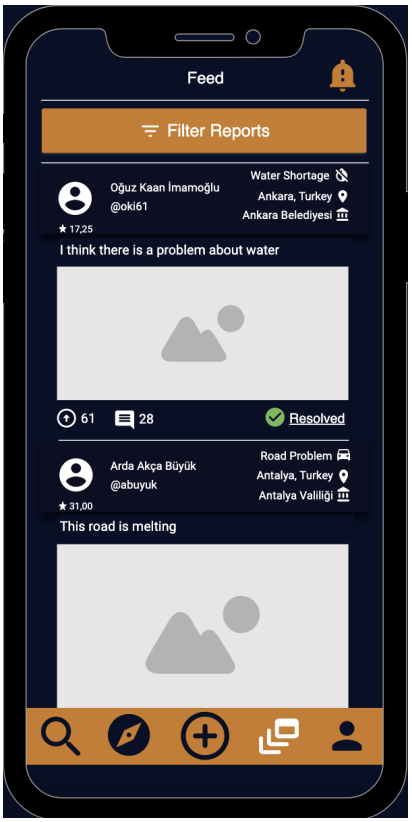
3.5.5.6 Forgot Password



In the first forgot password screen, users enter their mail address and after clicking the verification link sent to their mail address, they are navigated to the second forgot password screen and they can choose a new password here.



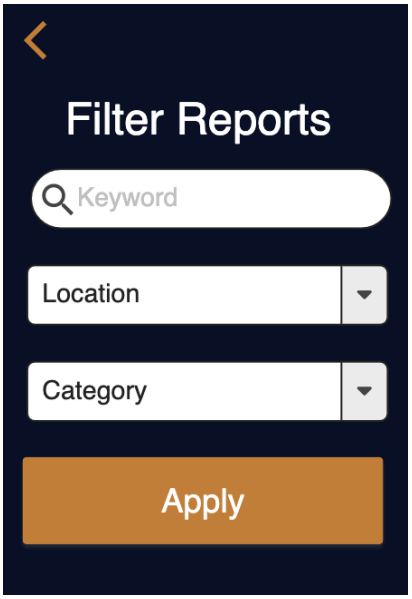
3.5.5.7 Feed Screen



The main screen is the feed screen. Citizens can navigate to other screens by using the navigation bar residing at the bottom of the screen. The active screen is indicated by a different color. The navigation bar includes 5 screens. These are feed screen, post report screen, profile screen, search/trends screen, and map screen. Also, every screen has a notifications button.

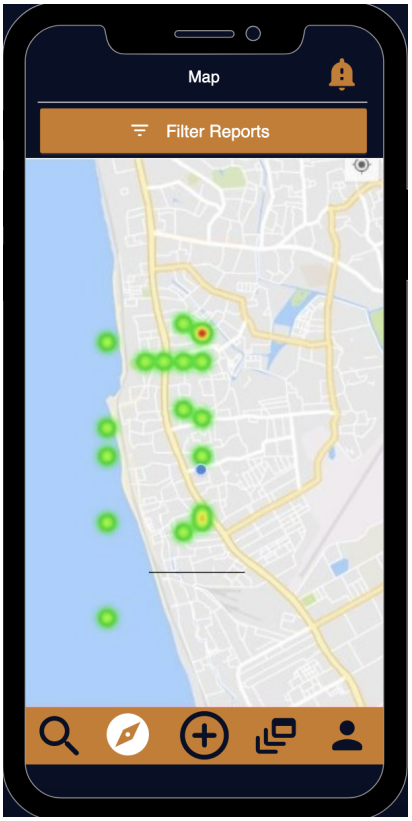
In the feed screen, citizens can see reports from other citizens, see the details of posts by touching the posts, or filter reports by touching the filter button.

3.5.5.8 Filter Mini Screen



In this mini screen, citizens can filter the posts by keywords, location, or category. After the Apply button is tapped, be returned to the previous screen and filtering is applied.

3.5.5.9 Map Screen



In this screen, citizens can view the reports based on the locations of the reports. Reports are represented as nodes on the map and by touching these nodes, related posts can be viewed. Colors of the nodes depend on the intensity of the reports of an area.

3.5.5.10 Search - Trend Screen

In this screen, citizens can search a citizen or an institution. In the middle of the screen, citizens can see trends. In the Trends section, it is seen which types of reports are popular, or if there is too much posting about a problem, that problem is seen, and how many reports there are for each element. Citizens can access a feed view where reports related to that topic are seen by touching the topic titles.

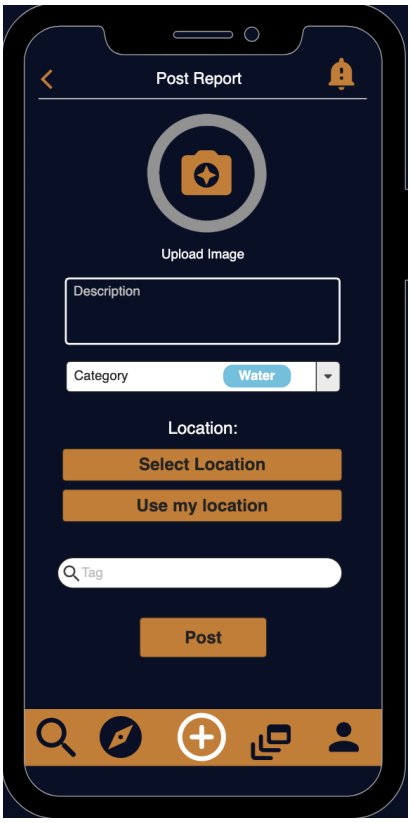
3.5.5.11 Searching

Citizens can search a keyword then they should choose whether they are looking for an institution, a citizen or a report. At the end of this selection, the relevant results are shown to the citizen on the screen.

3.5.5.12 Profile Screen

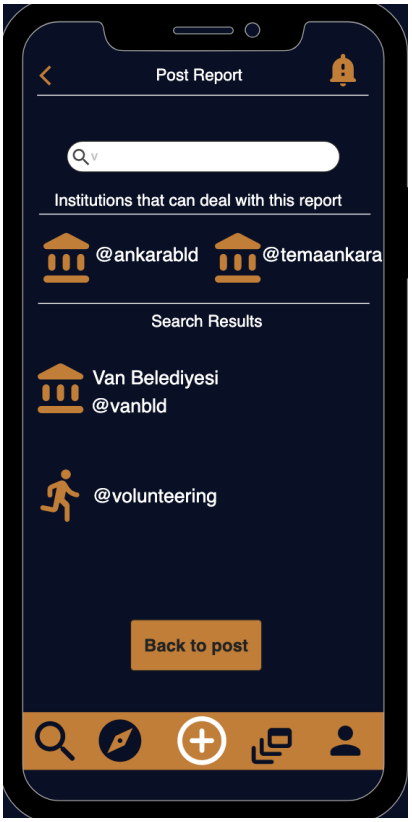
In the profile screen, users can see their bios, profile photos, scores and their unresolved and resolved reports. Also, the edit profile button is located in the top-left corner of this screen. Profile screens of other users are also similar, with one difference, there is no edit profile button.

3.5.5.13 Post Report Screen



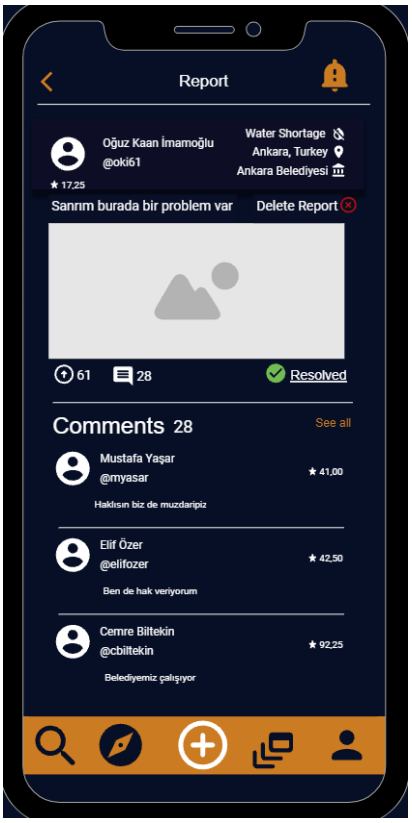
In this screen, citizens can post a report about a problem they saw. They should enter a descrşption about the problem, and upload a photograph of the problem. According to the photograph, the system recommends problem categories to users. Users can accept the recommendations or ignore them and choose manually. They can input the location of the problem manually, or they can just use their current location. Also they must add a tag to the report, in order to indicate who is the interlocutor of the problem.

3.5.5.14 Institution/Volunteering Tag - Recommendation



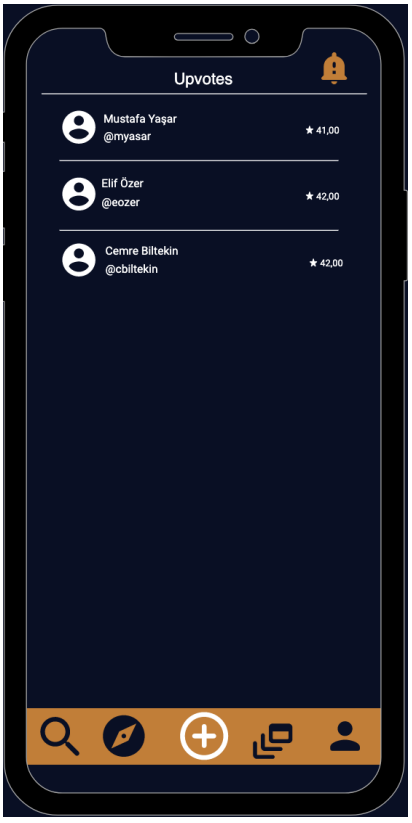
This screen will be opened when the tag bar in the previous screen is tapped. Users can search for an institution and can tag it, or they can add @volunteering tag in order to indicate the problem can be solved by volunteering. Also the system recommends proper institutions for the problem, these can be seen below the tag search bar.

3.5.5.15 Report Screen



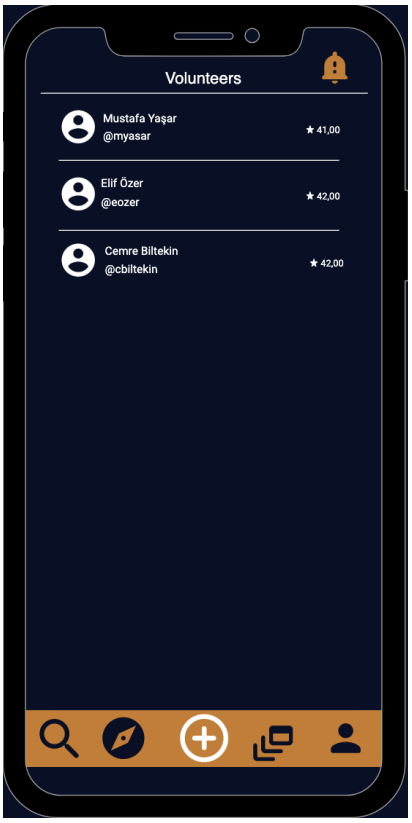
In this screen, citizens can see the report with details such as location, category, related institution, upvote count and whether they are resolved or unresolved. Also some of the comments can be seen on that screen.

3.5.5.16 Upvotes Screen



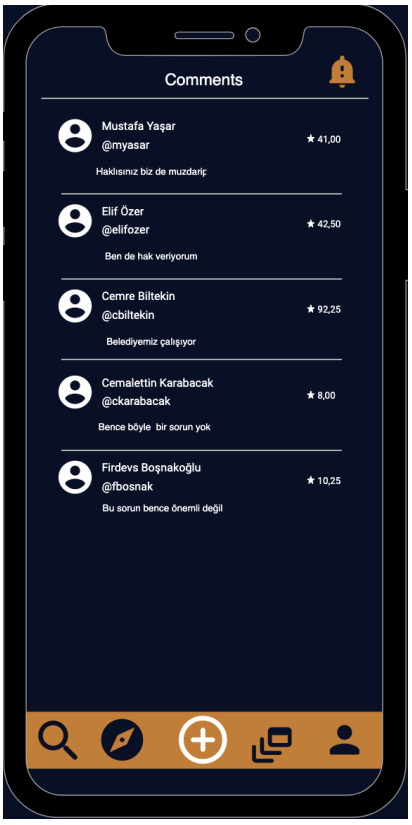
In this screen, citizens that upvoted the report can be seen.

3.5.5.17 Volunteers Screen



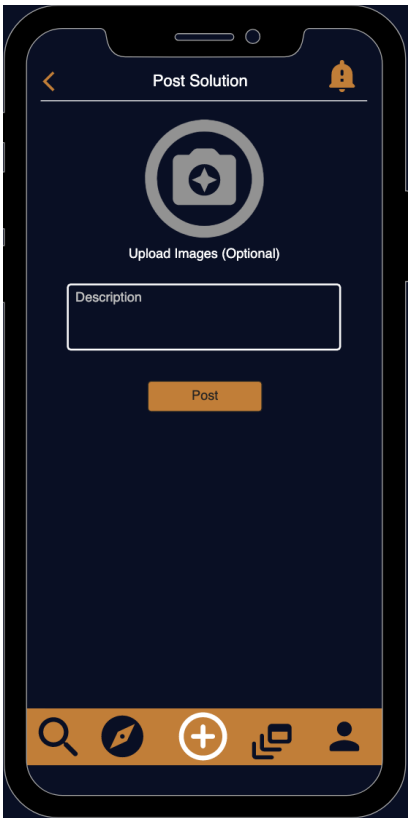
In this screen, volunteers of a report can be seen.

3.5.5.18 Comments Screen



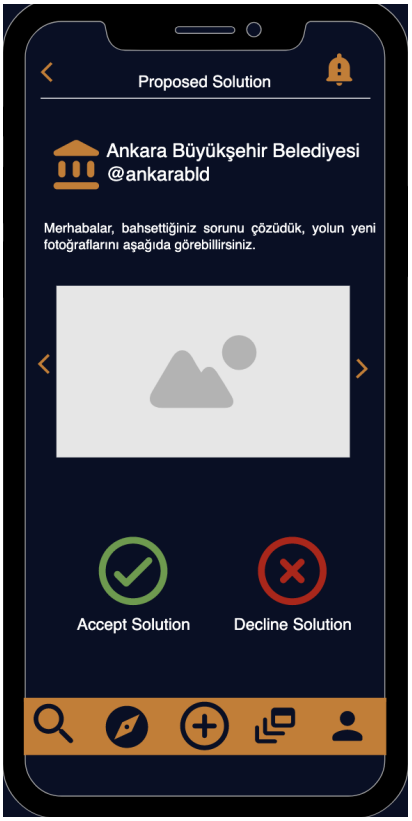
In this screen, citizens that commented about the report and their comments can be seen.

3.5.5.19 Post Solution Screen



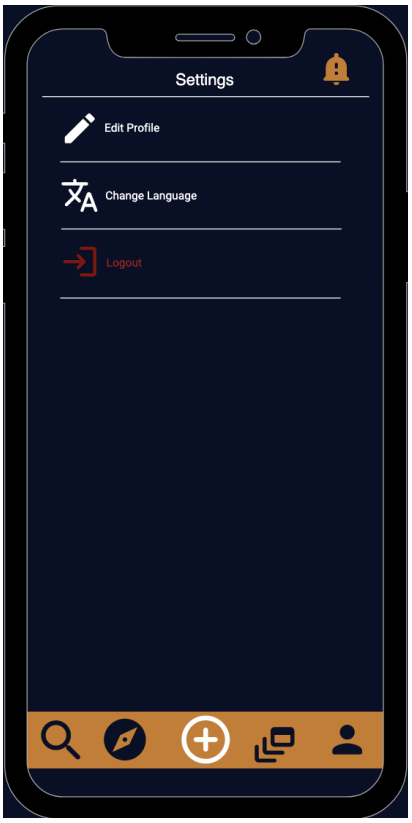
In this screen, institutions can post their solutions to a problem, they should upload a photo and enter a description about the solution they will post.

3.5.5.20 Solution Approval Screen



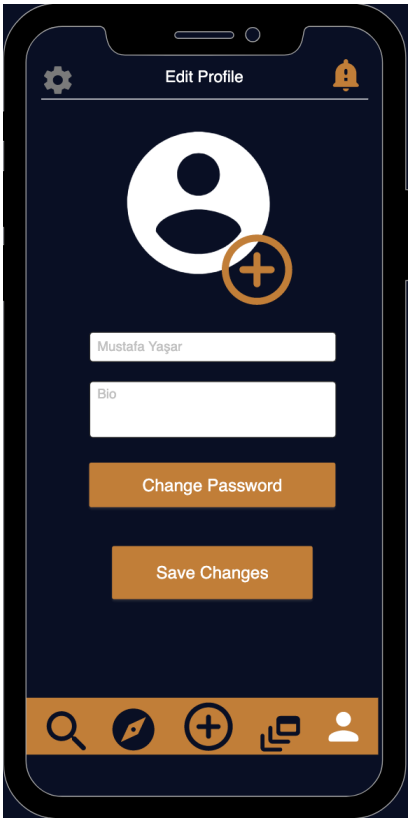
In this screen, citizens can approve or reject the solutions that are submitted to their reports.

3.5.5.21 Settings Screen



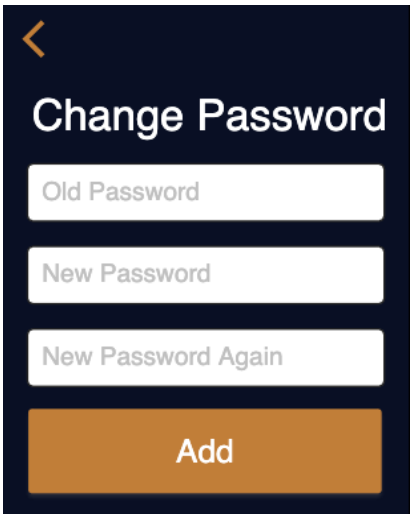
On this screen, users can log out, change the language of the application, or edit their profile by tapping the relevant item.

3.5.5.22 Edit Profile Screen



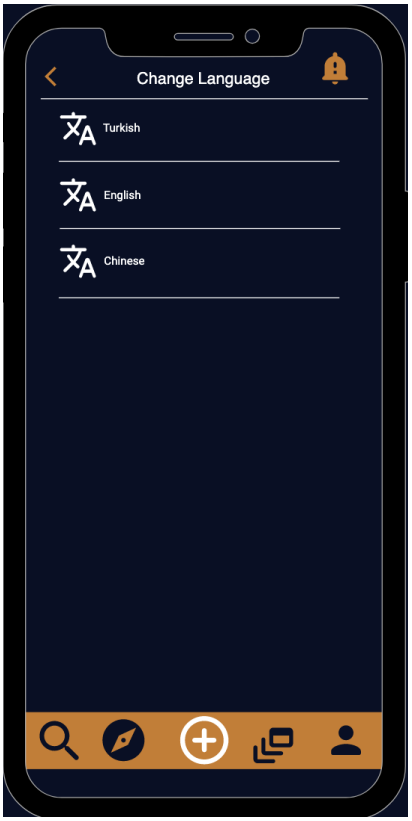
On this screen, one can edit their profile information such as profile picture, name, or bio. They also can change their passwords through the “Change Password” button.

3.5.5.23 Change Password Mini Screen



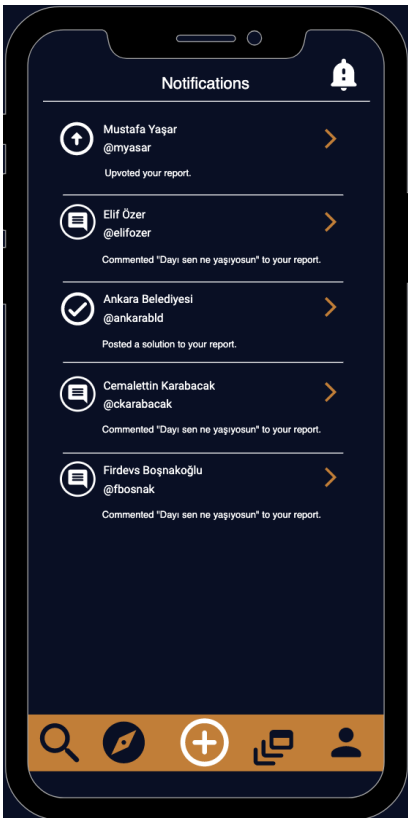
On this mini screen, one can change the password by entering the old password and the new password (twice).

3.5.5.24 Change Language Screen



On this screen, the language of the application can be changed.

3.5.5.25 Notifications Screen

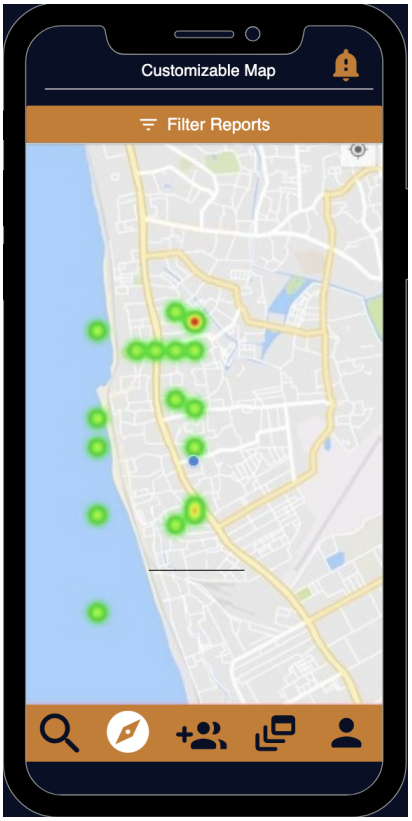


This is the screen where a citizen can see the notifications that they received.

3.5.5.26 Institution Profile

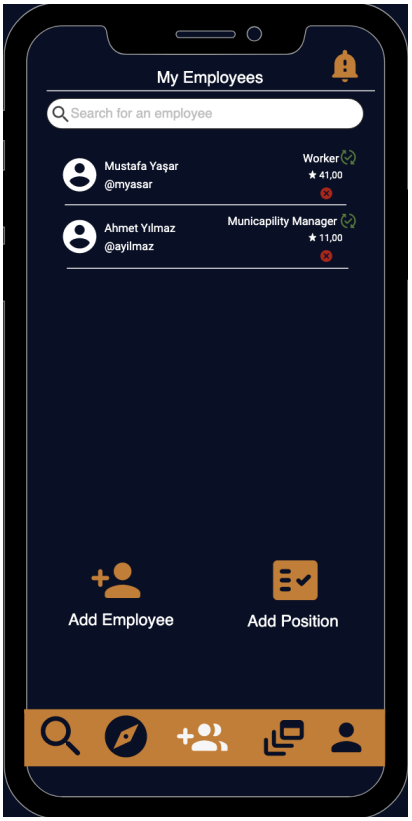
This screen is the institution profile. It contains information about the institution. The reports that the institution solved and unresolved can be seen. Also, one can see the officials of the institution, too.

3.5.5.27 Institution Map



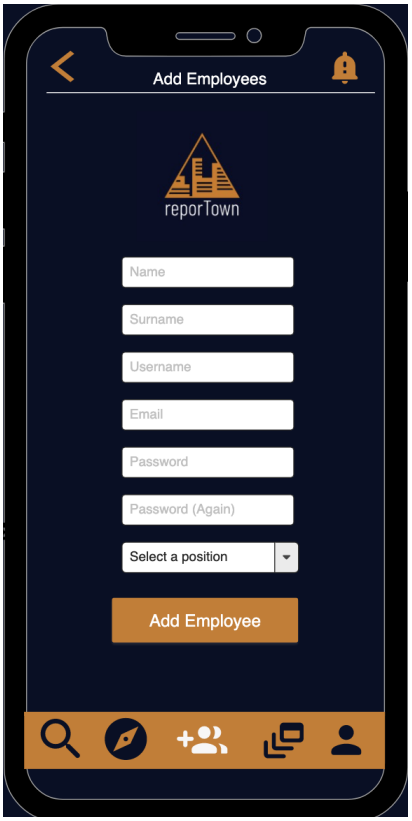
This screen is the map for an institution. It can see the reports associated with the institution itself. The “Filter Reports” option will be useful for institutions to change the view according to their needs for better planning.

3.5.5.28 Employees Screen



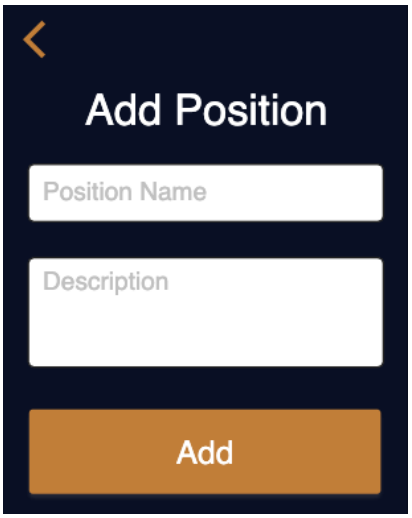
On this screen, an institution can see its officials (employees). It can add more employees through the “Add Employee” button, or it can add a position through the “Add Position” button.

3.5.5.29 Add Employee Screen



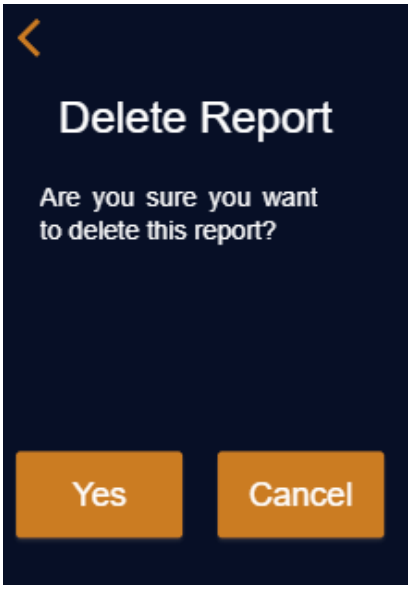
On this screen, an institution can register an official by entering the information and credentials of the official. After this form is submitted, an official can log in later on and post solutions to reports.

3.5.5.30 Add Position Mini Screen



On this mini screen, the institution can enter the information about the position and add it to their organizational structure.

3.5.5.31 Delete Report Mini Screen



This mini screen is where the user will be prompted an “are you sure” screen while deleting a report.

4. Other Analysis Elements

The following sections present how factors affect the project, project risks and alternative plan, project plan including goals and work packages, team collaboration methods, acknowledged responsibilities, and learning strategies for acquiring new knowledge.

4.1. Consideration of Various Factors in Engineering Design

Below are the factors that are taken into consideration in analysis which might have an effect on design and shape constraints/necessities for the project. Their degree of effect is given in Table 1.

Public Health

The project allows citizens to report city problems under eight significant categories, some of which pose risks for the public health, to encourage governmental institutions, private organizations and/or volunteers to eliminate the problem quickly. Consequently, protection of the public health is the main concern for reporTown. The most profound city problem that threatens public health and is one of the categories is improper garbage disposal on streets as it emits toxic gases and might create fire

hazards [2]. The citizens can report improper waste management to prevent being exposed to harmful gases. Other reportable problems are for ill stray animals and dangerous road conditions. The citizens can report stray animals in bad health condition via reporTown to eliminate the risk of spread of zoonotic diseases like rabies. They can also report detrimental road conditions like flooded/eroded/icy to forestall traffic accidents so that other reporTown users would be cautious about the reported road in their region as institutions would be acting to repair it. The project must efficiently call for the collective aid of institutions and volunteers to early eliminate public health threats mentioned above, so it must present a quick report generation with computer vision for problem categorization, machine learning for institution recommendation, and create a linked community for increasing the visibility of a report with upvotes from other users. Additionally, the citizens themselves should accept the proof of solution, so they have control over satisfaction of protection of their health.

Public Safety

Similar to ensuring public health, some of the eight categories for city reports like traffic accident reports are chosen to promote public safety by notifying appropriate institutions for solutions. Besides, other citizens in the region should see these reports on their feed which decreases the risk of hazard exposure by the public.

It is of utmost importance that only verified institutions and their verified officials must be able to register to the system since fake institution and official accounts would ruin the reputation of the imitated party as well as harm the purpose of the project which is providing communication to citizens with reliable problem solvers for city problems. For ensuring the safety of exchange between user groups, admins must verify each institution's registration on the system, and institutions must create accounts for their officials.

Moreover, reporTown’s posting a report process requires special permissions such as accessing the camera and/or camera roll for photographic evidence, current location information, and authentication information from the citizens. The application should ask for permission for each feature from the citizens and should promise the

safety of their data by stating the protection of their user data from third parties and encrypt the data.

Finally, inappropriate content in harmful reports can threaten user groups and frequent spams would shadow authentic reports and postpone the solution for serious city problems. Considering this, the project applies a spam filter on the institution and official reports feed to detect authentic reports. To protect user groups from inappropriate content in posted reports, a report button is added to mark them as inappropriate/irrelevant. These complaints are sent to the admin system to be reviewed. If the post violates community rules, the admins should hold the authority to delete the posted content from the system.

Public Welfare

The project makes the aiding communication from governmental and private organizations and volunteers to citizens easier by providing an easy to use free native application for the general age group. Also, reporTown is expected to include a wide range of institutions to keep the array of help available and diverse. In the long run, the project should improve the cities at a better rate than when only traditional or limited media platforms are used. However, if the city's welfare is low, the institutions and volunteers might not have the resources to solve city problems and citizens might not have access to a smartphone, so the download count of the application is affected negatively in that region and the application might not be preferred.

Global Factors

A world map is integrated into reporTown to support international users, which makes the application open for global use (any user type from any region will have its own region feed). Additionally, i18n internationalization framework provides easy implementation of language options and reporTown is initially expected to support English (the most spoken language) and Turkish. A legal constraint due to globalization for the mobile project is the General Data Protection Regulation (GDPR) which is imposed by the European Union (EU) to any organization that collects user data from the European countries [3]. Consequently, reporTown should ask for permission explicitly and collect only necessary user data, encrypt them, and designate data protection responsibilities for the project team.

Cultural Factors

Cultural factors do not have a considerable impact on reporTown as eight city problem categories are independent from cultural backgrounds. For instance, region A and region B can both have improper garbage disposal problems. Possibly, interest in volunteer work reports might change depending on how supportive a culture is for helping people in need.

Social Factors

reporTown supports a volunteering system to ease the resource utilization of institutions and introduce a cooperative aiding platform between citizens of the same region. Thus, citizens can adopt social responsibility if they choose to. They can communicate for volunteer work via comments on posted reports. All user types have scores based on feedback and report metrics for social credibility.

Environmental Factors

The project's motivation is to sustainably ameliorate cities with quick response from authorities and collaboration within communities. Fixing the reported city problems implies living in better environments for citizens from any social class with great accessibility. Following are environmental impacts of reporTown solutions:

- Proper waste management: Less greenhouse gas emission for the neighborhoods
- Proper road maintenance: Less frequent road reconstruction means less noise, dust and vibrations [4]
- Cooperative help for stray animals: Less disease spread means protection of local animal populations and citizens, more opportunities for adoption, decrease in overpopulation
- Reported water shortages: Better utilization of water resources (i.e. water tanks at homes) for the duration of shortage
- Reported traffic accidents: Preferring other routes with the acknowledgment of the accident decreases intensity of traffic on the lane of the accident and results in less traffic time and gas emission.

Economic Factors

The project functions better (better rate of problem solutions) as more users register from any type, so keeping the application free on both Google Play and Apple Store is essential. The more users the application attracts, more work is loaded to the database. Additional costs might be required to sustain the system in the future. This might be solved by requiring an affordable periodic membership fee from institutions in that case.

Table 1: Factors that can affect analysis and design and their degree of effect.

	Effect level	Effect
Public health	10	reporTown considers report categories to eliminate city problems threatening public health.
Public safety	9	reporTown considers problem categories, verification of institutions, access permissions, inappropriate content reporting system and spam filters with machine learning.
Public welfare	4	reporTown's active user rate in a region is affected by the region's welfare, and report solutions are equally provided to any social class citizen group.
Global factors	5	reporTown can be globally used on an interactive world map and language options are added to support internationalization.
Cultural factors	2	reporTown's addressed city problems are global problems independent of cultural backgrounds, volunteering tendency might change due to cultural value for altruism.
Social factors	3	Volunteering and commenting system in reporTown introduces social responsibility to its users.
Environmental factors	10	reporTown provides access to solutions which have a wide range of good environmental impacts such as less gas emission, noise pollution, disease spread.
Economic factors	2	reporTown is a free to register and use native mobile application to encourage more user rate, costs might increase with time.

4.2. Risks and Alternatives

Risks envisioned in reporTown project must be analyzed to address backup plans in case of the emergence of the risk to prevent an unexpected halt in the project plan and progress. The risks are given likelihood on the scale of low to high. 5 potential risks are addressed below:

Low accuracy in categorizing reports with computer vision: Computer vision and machine learning are two knowledge areas which the project team is still in the process of learning and gaining experience. Therefore, the categorization model for detecting city problems from photographs might not work as accurately as our expectation due to improper data engineering, labeling or modeling. False categorization would mean falsely directing citizens to choose the generated categories, but categories are editable before the report is submitted. Thus, even if the model had low categorization accuracy, the current interface design would permit users to ignore auto generated categories and edit to their preference. If the computer vision model proves to be unfixable, it is easy to remove it from the system to let citizens manually decide on categories fully.

Cyberattack on user data: There might be data breaches due to insufficient security analysis and design in the project in which the user data is stolen. This is a serious potential risk in privacy and security, and violates GDPR terms the project is expected to comply with. Such an incident would cause public uproar and result in active user loss and break trust in the application. Therefore, reporTown should use password hashing functions like Bcrypt to stop the possibility of attempts of decryption or reverse engineering of sensitive data. However, security configurations might not be enough due to our limited knowledge of cybersecurity to forestall a cyberattack incident, so in the case of a data breach a procedure could be followed [5]:

1. Find sources of data breach and vulnerabilities with the team and prevent additional data loss.
2. Immediately notify the users about the incident with its transparent details via another communication platform like SMS or email.
3. Disclose the content of stolen user data to users.

4. Instruct users to change their passwords.
5. Enhance the security of the current system.

Failure to accurately filter spam reports: Spam filters are essential if the application includes user generated content since the user is given the freedom to post to their liking. This is tolerable for entertainment mobile applications, but for service applications like reporTown it only means unsolved and unprovided service. Thus, the spam filter machine learning model is essential in keeping the report solving rate satisfactory as it saves time for institutions and their officials to only direct attention and resources to genuine reports. If the spam filter works incorrectly to mark authentic reports as spam and mistake spam as genuine, there would be serious unsolved city problems (the application would fail to fulfil its promised service) and the problem solving rate could drop (officials would be busy inquiring the validity of the report). Should the spam filtering model fail, another machine learning model could be implemented as a backup plan to increase the accuracy or filtering could entirely depend on admin actions and user complaints on spam reports/posts.

Table 2: Risks considered in reporTown project and fallback plans

	Likelihood	Effect on the project	B Plan Summary
Low accuracy in categorizing reports with computer vision	Low	False auto categorization, confusion of users in choosing categories for a report	If the model proves to be unfixable, remove the auto categorization system (categorization should rely wholly on manual category choosing by citizens)
Cyberattack on user data	Medium	Stolen user data, legal consequences, losing active users	Follow a certain procedure to stop further data loss, notify users with necessary details and transparency and enhance the security of the system in different ways
Failure to accurately filter spam reports	Medium	Authentic reports labeled as spam and never solved or spams	Change spam filter model and increase accuracy, filter reports

		pass as genuine reports and institutions' time and resource wasted	only when citizens mark reports as spam by admins

4.3. Project Plan

4.3.1 reporTown Gantt Chart

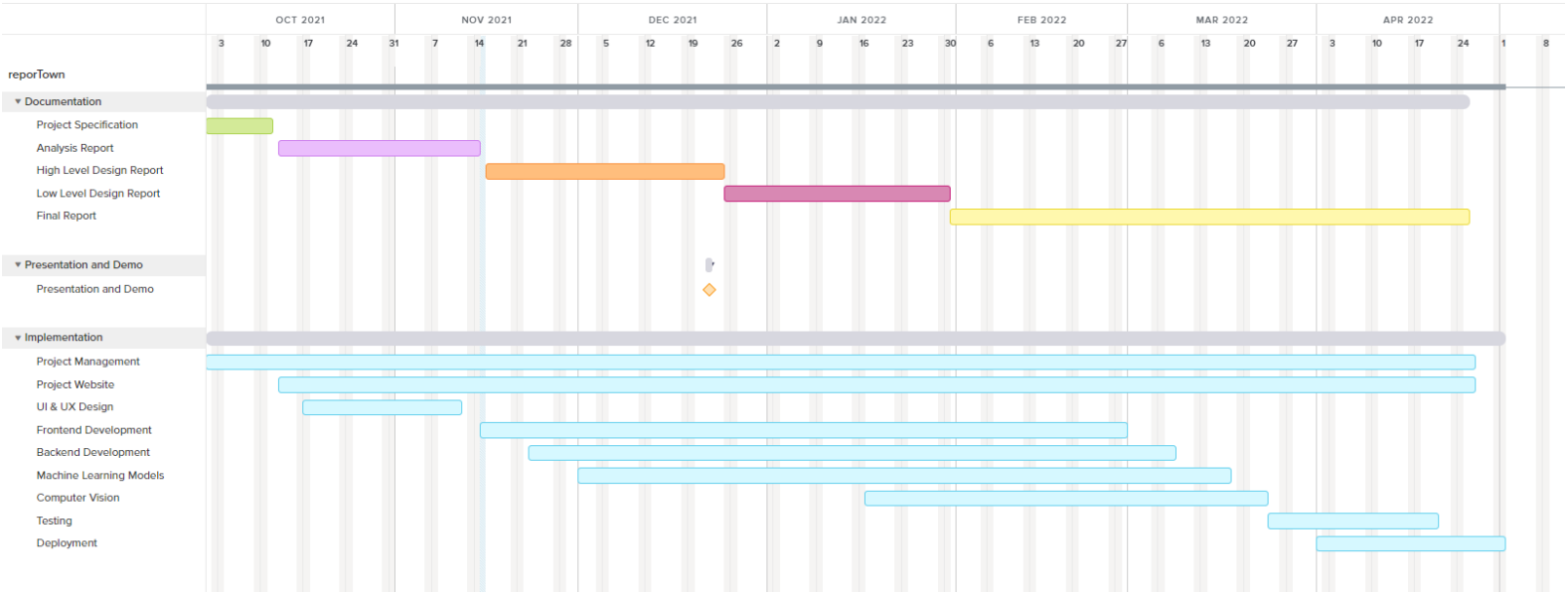


Figure 12. Gantt chart of the reporTown

4.3.2 Work Packages and Planning

Table 2: Work packages table for reporTown

WP	Work Package Title	Leader	Involved
WP1	Project Specification Report	Oğuz K. İmamoğlu	All team members
WP2	Analysis Report	Cemre Biltekin	All team members
WP3	High-level Design Report	Mustafa Yaşar	All team members
WP4	Low-level Design Report	Arda Akça Büyük	All team members
WP5	Final Report	Elif Özer	All team members
WP6	Presentations & Demos	Mustafa Yaşar	All team members
WP7	Project Website	Elif Özer	Oğuz K. İmamoğlu

WP8	Computer Vision	Cemre Biltekin	Mustafa Yaşar, Arda Akça Büyük
WP9	UI & UX Design	Arda Akça Büyük	Oğuz K. İmamoğlu, Mustafa Yaşar
WP10	Project Management	Cemre Biltekin	Arda Akça Büyük, Elif Özer
WP11	Machine Learning Models (Recommendations & Classification)	Oğuz K. İmamoglu	Cemre Biltekin, Arda Akça Büyük
WP12	Frontend Development	Mustafa Yaşar	Oğuz K. İmamoğlu, Cemre Biltekin
WP13	Backend Development	Elif Özer	Oğuz K. İmamoğlu, Arda Akça Büyük
WP14	Testing	Arda Akça Büyük	Elif Özer, Mustafa Yaşar, Cemre Biltekin
WP15	Deployment	Oğuz K. İmamoglu	Elif Özer, Mustafa Yaşar

Table 3: Project specifications wp table

WP1: Project Specifications Report
Start Date: Oct. 1, 2021 **End Date:** Oct. 11, 2021
Leader: Oğuz Kaan İmamoğlu **Members Involved:** All team members
Objectives: To give a title and a brief description of the proposed project, determine requirements and constraints of the project, evaluate the project in terms of different aspects such as sustainability.
Tasks: **Task 1.1:** Writing introduction **Task 1.2:** Writing requirements **Task 1.3:** Reviewing the report
Deliverables: Specifications Report

Table 4: Analysis report wp table

WP2: Analysis Report
Start Date: Oct. 13, 2021 **End Date:** Nov. 14, 2021
Leader: Cemre Biltekin

Members Involved: All team members
Objectives: Analysing the problem the project addresses and the system to be developed in an object-oriented way regarding its detailed requirements, application model, constraints, risks, project plan, learning strategies; and inscribe the system analysis on the analysis report as a deliverable.
Tasks: **Task 1.1:** Work division for analysis report writing **Task 1.2:** Preparation of the diagrams **Task 1.3:** Writing the analysis report according to responsibility of each member **Task 1.4:** Integration of written parts into final form of the analysis report **Task 1.5:** Analysis report review and delivery
Deliverables: Analysis report

Table 5: High-level design report wp table

WP3: High-Level Design Report
Start Date: Nov. 16, 2021 **End Date:** Dec. 24, 2021
Leader: Mustafa Yaşar **Members Involved:** All team members
Objectives: Creating a high-level design report in which the analysis model is transported into a system design model including design goals of the project, creating smaller subgroups to share among the project members, and considerations about public health, safety, global, cultural, social, environmental, and economic factors in engineering design process.
Tasks: **Task 1.1:** Researching about the considerations about the project, and inspecting those considerations in terms of the project. **Task 1.2:** Creating the high-level design report of the project. **Task 1.3:** Deciding about the communication of the smaller subgroups among each other.
Deliverables: High-level design report

Table 6: Low-level design report wp table

WP4: Low-Level Design Report
Start Date: Dec. 25, 2021 **End Date:** Jan. 30, 2022
Leader: Arda Akça Büyük **Members Involved:** All team members
Objectives: Refinement of the high-level design to low-level.

Tasks: **Task 1.1:** Explanation of the extent and validity of the design principles that were used to carry out the project. **Task 1.2:** Explanation of engineering standards used during design.
Deliverables: Low-Level Design Report

Table 7: Final report wp table

WP5: Final Report
Start Date: Jan. 31, 2022 **End Date:** Apr. 25, 2022
Leader: Elif Özer **Members Involved:** All team members
Objectives: Preparing a report stating final architecture, design and status of the project. Complete information about the system is provided and details about the test and maintenance plan will be given in detail with the other project elements such as ethics and professional responsibilities, judgements and impacts to various contexts and teamwork information.
Tasks: **Task 1.1:** Work division for final report writing. **Task 1.2:** Writing the details of requirements, final architecture and design, implementation and testing and maintenance. **Task 1.3:** Writing other project elements by explaining engineering/project standards used during design. **Task 1.4:** Final report review and delivery.
Deliverables: Final Report

Table 8: Presentations and demos wp table

WP6: Presentations & Demos
Start Date: Dec. 20, 2021 **End Date:** Dec. 22, 2021
Leader: Mustafa Yaşar **Members Involved:** All members
Objectives: Preparing the presentation and presenting the demo
Tasks: **Task 1.1:** Prepare the presentation to explain the project in detail to the audience **Task 1.2:** Conduct the demo
Deliverables: Presentation

Table 9: Project website wp table

WP7: Project Website
Start Date: Oct. 13, 2021 **End Date:** Apr. 26, 2022
Leader: Elif Özer **Members Involved:** Oğuz Kaan İmamoğlu
Objectives: Preparing web page for the reporTown describing the project. Each report will be available on this project website.
Tasks: **Task 1.1:** Preparing website **Task 1.2:** Make Project Specification available on the website. **Task 1.3:** Make Analysis Report available on the website. **Task 1.4:** Make High Level Design Report available on the website. **Task 1.5:** Make Low Level Design Report available on the website. **Task 1.6:** Make Final Report available on the website.
Deliverables: https://oguzkaanimamoglu.github.io/reporTown/

Table 10: Computer vision wp table

WP8: Computer Vision
Start Date: Jan. 17, 2022 **End Date:** Mar. 23, 2022
Leader: Cemre Biltekin **Members Involved:** Mustafa Yaşar, Arda Akça Büyük
Objectives: Implementing a computer vision algorithm for object identification and detection with utilization of YOLO algorithm and COCO dataset with OpenCV for city problem categorization on images in reporting process by a citizen
Tasks: **Task 1.1:** Training of YOLO object detector on COCO dataset inclusive of all 8 categories **Task 1.2:** Evaluation of the YOLO object detector for city problem categorization with test sets **Task 1.3:** Integration of the computer vision model into the system **Task 1.4:** Testing of the computer vision model in the system
Deliverables: Computer vision model for city problem categorization

Table 11: UI/UX Design wp table

WP9: UI/UX Design
Start Date: Oct. 17, 2021 **End Date:** Nov. 11, 2021
Leader: Arda Akça Büyük

Members Involved:
Objectives: Designing the UI with compliance to enhanced UX. Ease the job of the front-end people.
Tasks: **Task 1.1:** Defining navigational paths **Task 1.2:** Creating mock-ups **Task 1.3:** Communicating with front-end people during implementation
Deliverables: Mock-ups

Table 12: Project management wp table

WP10: Project Management
Start Date: Oct. 1, 2021 **End Date:** Apr. 26, 2022
Leader: Cemre Biltekin **Members Involved:** Arda Akça Büyük, Elif Özer
Objectives: Managing project management strategies, organizing efficient team meetings, setting up team environments for tracking team collaboration.
Tasks: **Task 1.1:** Choosing, setting up, and managing team collaboration tools which are GitHub, Jira, Google Docs. **Task 1.2:** Organizing sprints **Task 1.3:** Organizing team (weekly) meetings and supervisor meetings **Task 1.4:** Providing educational resources
Deliverables: Collaboration environments' pages

Table 13: Machine learning models wp table

WP11: Machine Learning Models (Recommendations and Classifications)
Start Date: Dec. 1, 2021 **End Date:** Mar. 27, 2022
Leader: Oğuz Kaan İmamoğlu **Members Involved:** Cemre Biltekin, Arda Akça Büyük
Objectives: Establishing machine learning models to be used for features such as

institution recommendation, problem classification, trend problems, obtaining the necessary datasets and training the models.
Tasks: **Task 1.1:** Getting or generating datasets **Task 1.2:** Building the models **Task 1.3:** Training the models **Task 1.4:** Testing the models **Task 1.5:** Adding necessary features to the application
Deliverables: Trained machine learning models

Table 14: Front-end development wp table

WP12: Front-end Development
Start Date: Nov. 15, 2021 **End Date:** Feb. 28, 2022
Leader: Mustafa Yaşar **Members Involved:** Cemre Biltekin, Oğuz K. İmamoğlu
Objectives: Implementing the user interface design of the application that aims to achieve maximum user friendliness and usability by using React Native
Tasks: **Task 1.1:** Starting implementation by creating registration, sign-in, sign-up, and profile screens **Task 1.2:** Implementing feed screen and integrating the front-end with the back-end **Task 1.3:** Implementing posting report, institution-volunteering recommendation, solutions, and institution profile screens **Task 1.4:** Implementing employee-related screens such as creating employees, adding employee screens **Task 1.5:** Implementing Map screens and integrating the machine learning algorithms to the front-end
Deliverables: All screens that can be reached while using the application

Table 15: Backend development wp table

WP13: Backend Development
Start Date: Nov. 23, 2021 **End Date:** Mar. 8, 2022
Leader: Elif Özer **Members Involved:** Oğuz K. İmamoğlu, Arda Akça Büyük

Objectives: Developing server logic and handling functionality of the application with the database connection.
Tasks: **Task 1.1:** Providing database connection **Task 1.2:** Implementing reusable code parts **Task 1.3:** Implementing logic of the application **Task 1.4:** Implementing APIs for get and post processes
Deliverables: APIs and Database connection

Table 16: Testing wp table

WP14: Testing
Start Date: Mar. 24, 2022 **End Date:** Apr. 20, 2022
Leader: Arda Akça Büyük **Members Involved:** Elif Özer, Mustafa Yaşar, Cemre Biltekin
Objectives: Automated/Manual Unit & Integration Testing of the application and reporting the bugs to developers.
Tasks: **Task 1.1:** Code Unit & Integration Tests **Task 1.2:** Assuring Continuous Integration **Task 1.3:** Reporting the bugs to developers
Deliverables: Unit & Integration Tests, Automated testing environment

Table 17: Deployment wp table

WP15: Deployment
Start Date: Apr. 1, 2022 **End Date:** May. 1, 2022
Leader: Oğuz Kaan İmamoğlu **Members Involved:**
Objectives: Making the application usable and accessible by building it, deploying new features, making it available in application stores
Tasks: **Task 1.1:** Building the application for Android **Task 1.2:** Building the application for IOS **Task 1.3:** Publishing the app in stores **Task 1.4:** Updating app and deploying new features if necessary
Deliverables: AAB file and IOS application

4.4. Ensuring Proper Teamwork

In order to be able to work efficiently in a software project that requires fast and continuous improvement as a team, proper planning is a crucial aspect. To achieve fast improvement and proper teamwork, agile methodologies that ensure the plans that are made to achieve success can be used. In this project, the Scrumban project management framework is being used by the developers. Scrumban merges two agile methodologies which are Scrum and Kanban.

In the Scrum methodology, the developer team decides what to complete during the next sprint which lasts one week for reporTown. Every member of the team decides on the work that they will be doing during the next sprint.

In the Kanban methodology, the developer team creates a board that contains the information about the team members and their responsibilities for the sprint. That way, every team member knows their responsibility as well as every other team member's.

The developer team of reporTown combines these two methodologies to achieve more durable, continuous, and fast improvement in their project. While creating and sharing the workload, it is important for the team members to select the work that they will be most efficient. Increasing the efficiency of a team member ultimately and significantly increases the efficiency of the project as they will be contributing to the project at their maximum potential.

The board has been created in Jira software that contains the issues with states such as To Do, In Progress, In Review, and Done. While working on an issue, the team members set the state of their issue so that other team members can know exactly the other member's work.

Documents for the project such as Project Specifications, Analysis Report, and Design Report are prepared by using Google Docs to enable every team member to contribute to the document simultaneously. While preparing documents, workload

has been arranged properly so that every team member collaborates on the documents equivalently.

The reporTown team uses whatsapp, discord, and zoom applications to communicate with each other quickly. That way, team members having difficulty on an issue, or team members working on a similar issue can help each other so that the issues can be solved easily and quickly.

Additionally, the developer team of reporTown uses GitHub which helps them to store and manage their code easily because they can check others' contributions to the code and changes in the code.

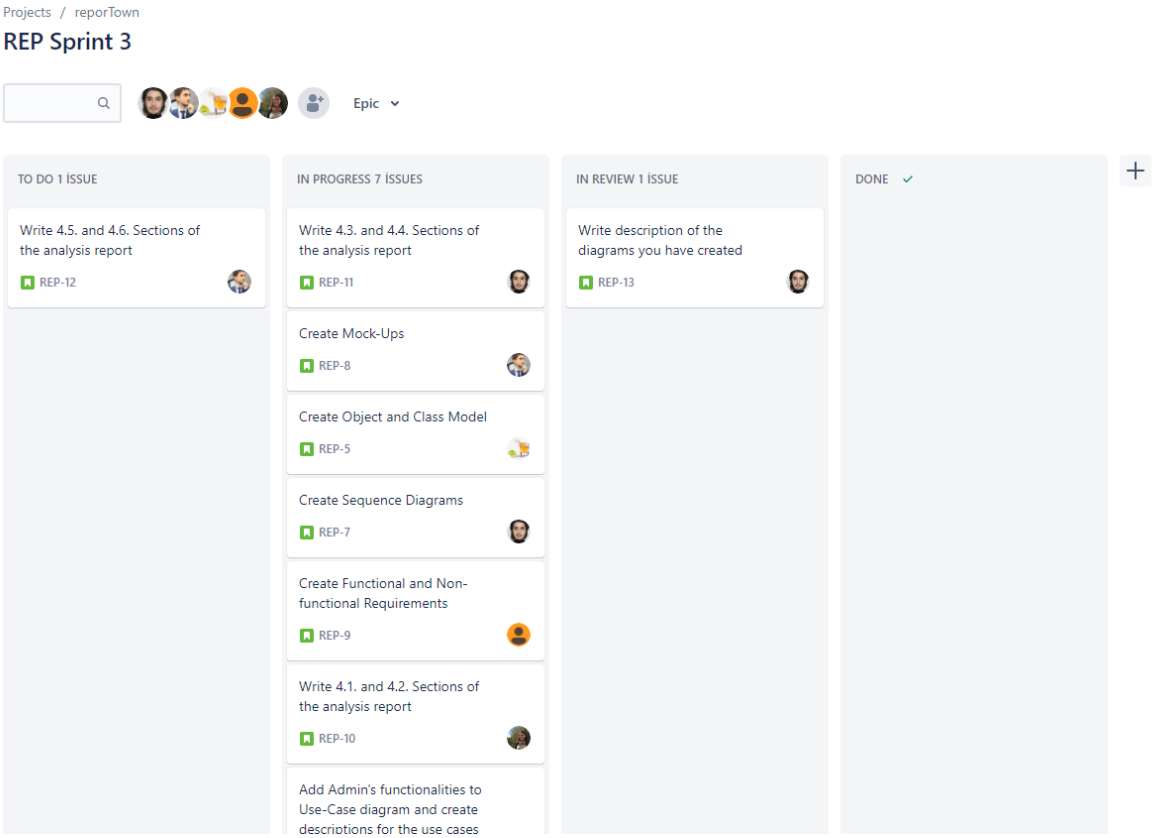


Figure 13. Jira board the reporTown team uses

The Jira board that gives information on the issues and their assignees can be seen above.

4.5. Ethics and Professional Responsibilities

ReportTown is an application that has an impact on the social, global, economic and environmental aspects, and this imposes many ethical and professional responsibilities on its developers and itself.

The most prominent aspect is the environmental aspect, because the goal of reportTown is to make cities more livable places and involve their residents in this process. Thanks to reportTown, it is possible to achieve a more beautiful and livable environment with the help of people.
ReportTown is not just an application designed for local use, on the contrary, it can be used anywhere in the world; It is open to the use of every person, regardless of race, country. With the worldwide use of the application, hundreds of different cities in the world will become more livable, and the application will contribute to the increase in the satisfaction of the city residents and even the people who come to the city as tourists.

ReportTown, which has effects on the economic aspect, will also contribute positively to the economies of the cities as it will help the development of the cities.
Another aspect taken into consideration while developing ReportTown is the social aspect. The application has a social media feature. Users' personal data will be used within the application and therefore this information is stored. Keeping this data safe is an important and huge responsibility. Also, along with its volunteering feature, reportTown aims to increase social cohesion, which is an important effect in the social aspect.

While developing an application with such effects, developers should meticulously fulfill the responsibilities brought by the project and be aware of the contribution value of their work.

4.6. Planning for New Knowledge and Learning Strategies

Since the project will be a large-scale project, many different technologies will be used. Some of these are technologies that we are all familiar with, some

technologies that only a few people on the team are familiar with, and some will be a new experience for all of us. Different learning strategies will be used for all of these.

For example, no one from the team has experience with the Google Maps API to be used to implement the map part, and it will need to be learned from scratch. For this, resources such as online resources, blogs, API documentation will be used.

React Native, Expo will be used for mobile development, MongoDB will be used as a database, Yolo will be used for computer vision, Spring will be used for backend development. These are technologies that one or two people on the team have experience with. In addition to online resources to learn technologies such as these, experience transfer between peers will also be applied.

5. Glossary

Reports: The content that the citizens posted to the application which includes the problem they encountered.
Citizens: People who are using the application by posting reports and requesting solutions to the problems they encountered.
Officials: Employees who are working for an institution. They can solve problems by analyzing reports and trying to solve the problems by taking necessary steps.
Institutions: Institutions like the municipality, governorship, or non-governmental organizations that are using the application in which. Officials work as employees in institutions.
Volunteers: People who are using the application in order to help other people by solving the reports they posted.
Admins: The team that is controlling the application by preventing any inappropriate content.

6. References

[1] “Mavi Masa - Ankara Büyükşehir belediyesi,” *Mavi Masa - Ankara Büyükşehir Belediyesi*. [Online]. Available: https://mavimasa.ankara.bel.tr/#about-mavimasa. [Accessed: 10-Oct-2021].

[2] L. Osborne, “Garbage governance: Poor waste management causes environmental crises: DW: 27.08.2015,” *DW.COM*, 27-Aug-2015. [Online]. Available: https://www.dw.com/en/garbage-governance-poor-waste-management-causes-environmental-crises/a-18677430. [Accessed: 15-Nov-2021].

[3] B. Wolford, “What is GDPR, the EU's new Data Protection Law?,” *GDPR.eu*, 13-Feb-2019. [Online]. Available: https://gdpr.eu/what-is-gdpr/. [Accessed: 15-Nov-2021].

[4] “4. environmental issues related to road management,” *ROADEX Network*, 23-Apr-2020. [Online]. Available: https://www.roadex.org/e-learning/lessons/environmental-considerations-for-low-volume-roads/environmental-issues-related-to-road-management/#:~:text=The%20three%20most%20damaging%20effects,roads%20and%20unbound%20aggregate%20layers. [Accessed: 15-Nov-2021].

[5] R. Lishchuk, “What you should do after a data breach,” *Clario*, 22-Feb-2021. [Online]. Available: https://clario.co/blog/what-to-do-after-data-breach/. [Accessed: 15-Nov-2021].