CITRA

Public **C**ivic **I**ssue **Tra**cking and Mapping Platform

About CITRA

Background and Purpose:

The lack of a transparent communication system between the residents of a particular locality and their corresponding local authorities causes immense trouble to the general public and is a major problem in metropolitan cities of developing countries like India, where the response to reported issues is very slow, and the problems themselves go unnoticed for a long time.

CITRA aims to fill in the communication gap between the residents and their respective authorities by providing a resilient and fast communication system, through which people can view existing issues, report their issues with ease and also track the progress. Such a system where existing issues can be viewed and their progress being monitored by all the residents nearby, puts more weight on the resolution of the issues, and might help in faster action by the authorities.

In addition to this, CITRA also brings an ingenious SOS feature, where a user can call for emergency services , as well as call for help from nearby CITRA users, who can lend a helping hand before the emergency services arrive at the scene, all at the click of a button.

Click here to see UML Use Case Diagram.
Click here to see the mock flow and wireframe of the application.

Assumptions and Constraints:

Since the application is primarily powered by the public, we expect the public to be proactive in terms of reporting their issues and grievances on the app.

Secondly, the authority end of the application requires some form of a desktop browser. Hence, we assume that a desktop computer capable of web browsing is available at the local authority.

Interfaces to external systems:

The application would use one of the openly available map APIs such as Google Maps,OpenStreetMap etc. The exact implementation details would be finalized once further exploration is done in this domain.

Functional Requirements:

Data Requirements:

1. Users (Public/Residents) :
    a. Name
    b. Email
    c. Phone number
    d. Location
    e. Photo
    f. Reward credits
2. Authorities (Municipal/NGOs/Govt Services)
    a. Name
    b. Type of Authority (Category under which the authority is classified as: eg. Water Works Dept, Municipal Corporation etc.)
    c. Location
    d. Address
    e. Phone Number
    f. Email
    g. Types of issues addressed (The type of issues which the authority is responsible for.)
    h. Incorporation Details(PAN Card Details)
    i. Photo

3. Issues (Public grievances)

a. Title

b. Description

c. Type of issue (Category under which the issue is classified: eg. Garbage related issues, road works issues, water supply issues etc)

d. Location

e. Photo

f. Tag

g. Added Timestamp

h. Upvotes (Indicative display of people’s self marked concern over the issue. More number of upvotes indicate more users are concerned about the issue. A user can upvote a certain issue only once)

i. Assigned Authority (The authority to whom the issue has been assigned for resolution)

j. Comments(username:Comment text: timestamp) (Made by the people, to convey their words to the authority)

k. Completion Status (Descriptive Status of the resolution of the issue. Marked by the assigned authority)

l. Added By (User details of the user who has added the issue)

m. Positive Verifiers (users who have verified the issue to be true through the rewards program.)

n. Negative verifiers (users who have verified the issue to be false through the rewards program.)

Process Requirements:

1. User Processes:

a. Profile Related Processes

  i. Sign Up: Used to create a new account as a resident. Email, Password, Phone Number, Location, Display Picture etc information is taken

1. Verify Email/OTP: Once user signs up, Email and Mobile Number verification is mandatory to access the application (Verification link is sent to the specified email and phone number)

ii. Login: To validate and enable an existing user to use the functionality of the application. Email and Password are required.

iii. View and Edit Profile: All the details of the user(except password) are displayed to themselves to view/edit

b. Issue/Complaint Related Processes:

i. View Nearby Issues: Shows the existing issues in the vicinity of the user based on their location information. (Takes current location of user as input and searches for issues within a specified radius)

ii. View Nearby Authorities: Shows the existing authorities in the vicinity of the user based on their location information. (Takes current location of user as input and searches for authorities within a specified radius)

iii. View Complaint: Displays the details of a specific complaint from the set of existing complaints (All the attributes mentioned in the data requirements for issue will be displayed)

iv. View Authority: Displays the details of a specific authorities from the set of existing authorities (All the attributes mentioned in the data requirements for authority will be displayed.)

v. Add Complaint: Enables user to add a new complaint. (All the attributes specified for issue need to be provided by the user)

vi. Edit Complaint: Enables the user who had originally added the complaint to edit any mistakes

vii. Verify complaint: Verification of the issue is done through the app's reward programme, where the user can go to an issue location and verify if the issue actually exists or not. User must add a photo to support his claims.

viii. View rewards: Enables the user to see the rewards he has won so far.

ix. Upvote Complaint : Enables the user to “Upvote” a Complaint (“Upvote” - as defined in the data requirements section).

x. Add Comment: Enables the user to add a comment on the Complaint

c. Additional Processes:

i. Search: Takes a search term from the user and returns relevant results across various items such as issues, authorities, locations etc.

ii. SOS Button: At the Click of the SOS button, an alert is sent to nearby Emergency services(Based on the emergency type chosen) . Also, nearby users are alerted. Both are sent the details of the person and their GPS coordinates.

iii. Receiving SOS alerts : User may receive an SOS alert if a person nearby presses SOS Button on their phone.

2. Authority Processes:

a. Profile Related Processes:

i. Sign Up: Used to create a new account as an authority. Authority Name , Type of Authority, Type of Issues dealt with , Contact details, as well as proof of incorporation , Location details need to be provided.

ii. Login: Existing Authorities can login and access the application by entering their credentials.

iii. View and Edit Profile: All the details of the authority are displayed to themselves to view/edit.

b. Complaint Related Processes:

i. View Nearby Issues: Shows the existing issues in the vicinity of the authority based on their location information. (Take location of authority as input and searches for issues within a specified radius)

ii. View Complaint: Displays the details of a specific complaint from the set of existing complaints (All the attributes mentioned in the data requirements for issue will be displayed.)

iii. Status Update: Authority can add an official status update about the progress on a specific complaint , on the complaint's "view complaint" page

iv. Add comment: Authority can add a comment on a specific complaint

v. Mark as complete: Authority can mark a specific issue as completed/resolved.

c. Additional Processes:

i. Receiving SOS alerts: Emergency Authorities(based on type) may receive SOS alerts if a user presses the SOS button on their phone. The user's details and location will be sent to the authority

3. Operational Requirements

3.1 Security

Following are the security and access management measures related to User account privileges:

1. The owner of an account can edit their profile, posts, comments.
2. Users can only view limited information on other users' profile.
3. Users can view and track any reported issue posted by other peers.
4. Users can neither update the status of a reported issue nor can they delete any reported issue posted by peers
5. Only respective authorities can update the status of a reported issue.
6. Only people close to an issue can validate the updates on the issue.

3.2 Audit trail

1. History of completed issues will be tracked
2. Updates on current issues will be tracked

3.3 Data Currency

Data should be as current as possible both from user and authority ends.

3.4 Reliability

Working of SOS is the most critical part of the application. Its reliability depends on the telecommunication network and internet access at a particular location.
Other features such as issue updation or issue posting are also susceptible to internet connection. However, caching state of the application on the devices will make it work in a bad internet network scenario.

3.4 System Availability

System Availability: 24-7 service
Peak time: Around 5 pm IST every day

3.4 Performance

Response time:

For SOS : < 2 sec
For other queries/actions: <30 secs

Expected volume of data:

Issue details(as listed above) for 1 issue per user per day.

3.5 Capacity

Data Capacity:

1. Approx 40 users per square km
2. 60 issues per month per square km

Data Retention:

1. Issue data: until issue is resolved
2. User data: forever