2021

POLITECNICO
MILANO 1863

REQUIREMENTS ANALYSIS AND SPECIFICATION DOCUMENT

SE4G – SOFTWARE ENGINEERING FOR GEOINFORMATICS

AMIEVA, AUSTONI, SONDOQAH

POLITECNICO DI MILANO – MSC. GEOINFORMATICS ENGINEERING

Content

Overview

This document has as a main objective give a technical explanation about the requirements and specifications detected by the team in terms of the development of the new web APP "Milano Road Network Monitor".

The mentioned app will detect anomalies in the state of the urban roads of Milano by giving the opportunity to the citizens to contribute when any of them detects a problem in the urban network. At the same time the authorities in charge of the maintenance of the road network will acquire those anomalies reported and process the priorities of intervention of them. Finally, the platform will provide feedback to the citizens about the status of their requests and will show analysis performed about the maintenance processes in general (requests attended, kinds of problems, requests to attend, among others).

Project Goals and Scope

Goals

The project aims to develop a web app which could attend the following issues of the urban road network of Milano:

- Improve the level of communication between authorities in charge of road maintenance and citizens
  - Citizens may upload requests of maintenance once they detect an anomaly in the urban road
  - Authorities should answer users (citizens) requests with a planning intervention date
  - Users could rate the performance of the interventions
- Measure and display information about the Key Performance Indicators [KPIs] of the maintenance process of the urban road network
  - Average service time
  - Satisfaction level after intervention
  - Quantity of interventions solved
  - Quantity of requests received in last week/month
  - Quantity of requests not considered as relevant to repair

Scope

The application will be a "collaborative" management system which receives as inputs the requests of maintenance of citizens and transmits them to the authorities (who also will have a special user role in the platform). This system will also show and monitor KPIs about the maintenance process to the citizens.

Domain Analysis

Domain Entities

- Users
  - Citizens
  - Local Authorities (in charge of road maintenance)
- Streets anomalies database

Domain analysis table

Table 1: Domain Analysis

Phenomena	**Location**	**Controlled by**
Data is collected	Shared	World
The user opens the web explorer	World	World
The user searches in the web explorer the url of the web platform	Shared	World
The user selects the right web section to identify itself like citizen or authority	Shared	World
The system redirects the request to the right web section for citizens or authorities	Machine	Machine
The user registers in the app	Shared	World
Passwords are encrypted by the system before being stored	Machine	Machine

User's credentials are registered in the Database server of the app	Machine	Machine
The authority users are required to complete their registration by filling the related form	Shared	World
The system informs the official moderator by Email about the request of a new authority user	Shared	Machine
The official moderator opens the email and analyzes the request of a new authority user.	World	World
After the official moderator approves the authority user registration, the system queries to the DB to be updated	Machine	Machine
Once the new authority user account is created, the system informs the related user with approval of its account by email	Shared	Machine
Users sign in	Shared	World
Username and password are compared to already existing one in the app database	Machine	Machine
If login fails, an error message is displayed to the user	Shared	Machine

The citizen user selects the desired section in the web page depending on whether he wants to report a distress or visualize the map	Shared	World
The system redirects to the right section of the web page	Machine	Machine
The citizen detects a distress in the urban road network	World	World
The system provides a link to the Epicollect5 project to the citizen	Shared	Machine
The system redirects the request to the Epicollect5 project form	Machine	Machine
The citizen completes in Epicollect5's forms the details about the distress detected	Shared	World
The citizen user returns to the web app and confirm the successful uploading in Epicollect5 by pressing a confirmation button	Shared	World
A get request to update the database is sent once the user press the mentioned button	Machine	Machine
The map is re-rendered in order to show new information's	Shared	Machine
The system sends an email to the user in order to assert the success of the	Shared	Machine

operation and gives a track ID (of the request)		
The system gives a notification to the authority users about the added request in the notification section	Shared	Machine
In the authority web section, the system displays a map with the different status of the citizen requests and a notification section where only new requests of intervention are reported	Shared	Machine
Authority users access to the new request	Shared	World
Distresses information's are displayed to authority users	Shared	Machine
The new request is removed from the notification section by the system	Shared	Machine
Authorities users plan an intervention and assign a level of priority	World	World
Authorities users update the related request with the estimated date of intervention and setting the status to "in charge"	Shared	World
The system updates the citizen user's request of intervention in the database with the new status	Machine	Machine

The system updates the map in order to include the new information about the citizen request and the status	Shared	Machine
Authorities fix the road damage after planning the intervention	World	World
After finishing the work authorities change the status of the corresponding request to “fixed” in the web app	Shared	World
A request to update the database with the new status is sent by the system	Machine	Machine
The new map is rendered after updating the database	Shared	Machine
The citizen user selects the section about general information in the web page	Shared	World
The request is redirected to the section with general information and KPIs of the maintenance process of road urban network of Milano	Machine	Machine
The system queries the DB for some KPIs predetermined from the maintenance process: total quantity of maintenance requests, Total quantity of maintenance requests solved, Total quantity of maintenance requests uploaded by citizen users depending to kind of distress, Mean attention time, Max attention time (among others)	Machine	Machine

The system renders the mentioned KPIs in predetermined charts associated to them	Shared	Machine
The citizen selects a category of distress in the web page (if he wants to)	Shared	World
The system provides different chart and map configurations to the citizen, accordingly to his category selection	Shared	Machine
The citizen user selects the desired distress in the chart	Shared	World
Information about the distress is provided by the system to the citizen	Shared	Machine

Relevant Phenomena

World Phenomena

1. The user opens the web browser
2. The official moderator opens the email received where is reported the new request of registration and analyzes it.
3. The citizen detects a road distress in the urban road network
4. Accordingly, to the request of intervention authorities' users plan an operation and assign a level of priority to the request in order to classify all the requests and serve first the ones with higher priority
5. Authorities fix the road damage after planning the intervention

Machine Phenomena

1. The system redirects the request (of accessing the right page) to the web section for citizens or authorities
2. Passwords are encrypted by the system before being stored in the database
3. Username and password are saved in the Database server of the web app
4. After the official moderator approves the authority user registration, the system queries the DB asking for an update

5. Username and password are compared to already existing one in the app database at the moment of log-in
6. The system redirects the request to the right web page for reporting an anomaly or visualizing the anomaly map
7. After clicking to the link to Epicollect5 app in the web page the system redirects the request to the Epicollect5 project form for inserting new data
8. A get request to update the database is sent once the user press the “get ID” button
9. The system updates the status of the citizen user’s request of intervention in the database with the new one (if the authority takes it in charge the status will change to “in charge”)
10. A new request to update the database with the status is sent by the system after the authorities close the citizen claim of intervention
11. After clicking the button of general informations the request is redirected to the section with general info and KPIs of the maintenance process of road urban network of Milano
12. The system queries the DB for some KPIs predetermined from the maintenance process: total quantity of maintenance requests, Total quantity of maintenance requests solved, Total quantity of maintenance requests uploaded by citizen users depending to kind of distress, Mean attention time, Max attention time (among others)

Shared Phenomena

Phenomena controlled by the **world** and observed by the machine:

1. Citizen user collects data
2. The user searches in the web explorer the url of the web application
3. The user selects the right web section to identify itself like citizen or authority
4. The user registers in the app
5. The authority users are required to complete their registration by filling the related form in order to be evaluated for the registration
6. Users makes the sign in operation
7. The citizen user selects the desired section in the web page depending on whether he wants to report a distress or visualize the map
8. The citizen completes in Epicollect5’s forms the details about the distress detected

9. The citizen user returns to the web app and confirm the successful uploading in Epicollect5 by pressing a confirmation button (“get ID”)
10. Authority users opens the new request made by the citizen user
11. Authorities users (after an intervention plan has been planned) update the citizen request with the estimated date of intervention
12. After finishing the work authorities change the status of the corresponding request to “fixed” in the web app
13. The citizen user selects the section about general information in the web page
14. The citizen selects a category of distress in the web page (if he wants to)
15. The citizen user selects the desired distress in the chart displayed by the system accordingly to the user category selection

Phenomena controlled by the **machine** and observed by the world:

1. The system informs the official moderator/admin by Email about the request of registration of a new authority user
2. Once the new authority user account is created, the system informs by email the related user that his registration has been approved
3. If login fails, an error message is displayed to the user
4. The system provides a link to the Epicollect5 project to the citizen
5. The map is re-rendered in order to show informations about the new request of intervention made by the citizen and if the citizen user is registered also the status of his request is shown
6. The system sends an email to the user in order to assert the success of the retrieval of the data uploaded in Epicollect5 and gives a track ID (of the request)
7. The system gives a notification to the authority users about the new request made by the citizen in the notification section
8. In the authority web section, the system displays a map with the different status of the citizen requests and a notification section where only new requests of intervention are reported
9. Distress’s informations are displayed to authority users (after they open the desired request of intervention)
10. The new request of intervention is removed from the notification section by the system after the authority has opened it

11. The system updates the map in order to include the new information about the status of citizen's request that has been taken in charge
12. The new map is rendered after updating the database with the new "fixed" status (once authorities set the new status after finishing the work)
13. The system renders the KPIs in predetermined charts associated to them
14. The system provides different chart and map configurations to citizens, accordingly to their category selection
15. Information about the distress is provided by the system to the citizen

Use Cases

Actors

1. Registered users:
   - 1.1. Registered users who are specialized persons: Users who have the access to advanced processing and maintenance suggestions, they can also access the database and perform updates there. Those users can also access what the normal citizens users can.
   - 1.2. Registered users who are normal citizens: Users who can visualize the data and apply some customizations, their use is restricted with read-only use. They can also keep track of the status of their request and the mean time of intervention.
2. Non-registered users: Users who can use the website without having credentials to sign in, but cannot see the status of their request made in Ep5.

Use cases concerning users

Member registration

- Use case name: RegisterMember
- Actors: Users who want to be Registered, as a specialized user or normal citizen.
- Entry Condition: User opens the web page.
- Flow:
  - The user selects the "register" option.
  - A window pops-up asking the user about his major (work environment), the registration form should be built based on the user major; in case he's registering as normal citizen he should be asked about his name, chosen username, his email, and his residence location.

        - In case he’s a specialized person; more information should be necessary to insert, such as his role and his department.
        - The user fills the form and confirms the process.
        - The username of the user is stored in the software’s database.
        - The registered user password is encrypted before being stored in the software’s database.
    - Exit Condition: The software stores the input information in the database.

Member login

- Use case name: LoginMember
- Actors: Registered members.
- Entry Condition: Member opens the web page.
- Flow:
    - The Member selects the “login” option.
    - The software opens a window asking the Member for username and password.
    - The Member submits his/her username and password
    - The software checks if the member’s credentials are stored in the database.
    - The software grants the Member access to the full properties of the web page if his credentials match the ones in the database.
    - The software returns an error message if the user provides wrong credentials.
- Exit Condition: The member is successfully logged in.

Uploading a maintenance request

- Use case: UploadingMaintenance
- Actors: Citizen Members
- Entry condition: User open the related section on web page
- Flow:
    - The user chooses the section of the web page where is displayed the link to epicollect
    - The user completes the request in epicollect page
    - The user return to the web page and confirm the completion of the new request in epicollect site.
    - The citizen member receives an email by the system with the track ID of his request of intervention in order to track the status
- Exit condition: The system redirects the user to the main page.

Authorities Notifications of new maintenance requests

- Use case: NotificationsAuthorities
- Actors: Authority Members
- Entry condition: A new request is detected in the "Maintenance requests" table of the system database
- Flow:
    - The "Maintenance requests" table of the system database is updated by a get request of the epicollect api after a citizen user confirms his/her uploading
    - The system alerts to the authority's users about the new request in the notification section
- Exit condition: The authority user confirms the notification.

Updating of maintenance requests

- Use case: UpdatingRequests
- Actors: Authority Members
- Entry condition: An authority select a request and press on "modify" option in the maintenance requests section of the web page
- Flow:
    - An authority user requires the modification of a maintenance request status
    - The system displays the related request section for making changes in the status
    - The user modifies the information and/or status of the request
- Exit condition: The system retrieves a message about the success in the modification.

Citizen Notifications of requests

- Use case: NotificationsCitizen
- Actors: Citizen Members
- Entry condition: A request is updated by an authority user
- Flow:
    - Once an authority user updates a request (by planning a date of intervention or by marking as "solved") the system send a notification to the related (notification section of the) citizen user
- Exit condition: The citizen user opens the request notification and verify its status

Filtering and visualization of charts

- Use case name: VisualizeData

- Actors: Both Members and unregistered users.
- Entry Condition: Members search for inquiry.
- Flow:
  - The user chooses the Data Visualization option from the main page of the website.
  - The Member asks to choose a visualization method based on the visualization type.
  - The software displays information according to the User's selection.
- Exit Condition: The software returns the desired data.

View Map

- Use case name: ViewMap
- Actors: Members and unregistered users.
- Entry condition: Members open the web page.
- Flow:
  - TheMember selects the "View Map" option.
  - The software redirects the Member to OpenStreetMap rendered Map in View Map page
- Exit Condition: The software opens OpenStreetMap in the View Map page.

Processing

- Use case name: Maintenance
- Actors: Members-only specialized members-.
- Entry Condition: Members looking for maintenance suggestions.
- Flow:
  - The Member selects maintenance option from the main page.
  - The Member is asked to choose a distress entity viewed on the map.
  - The software displays information according to the Member's selection.
- Exit Condition: The software returns the optimized maintenance based on the selected data.

Contact us

- Use case name: ContactUs
- Actors: Registered Members.
- Entry Condition: Members who want to reach the moderators and the body responsible for the web app.
- Flow:
  - The Member selects contact us option from the main page.

  - The Member is asked to fill a contact form including some information about the reason of contact, phone number, and the request text-body.
  - The member submits the form.
- Exit Condition: The software stores the input information in the database.

Requirements

Technical Requirements

- The system should be applied in python.
- The system should be displayed on the web.
- The system shall be created using -OUR PROJECT NAME- Epicollect5 dataset.
- The system should offer customized visualization of the data:
  - The system will use OpenStreetMap as a default base-map
  - The geographic data will appear as a on the top of the basemap where the data point location appears in a specific style.

Functional Requirements

- The System should allow the user to visualize all the distresses updated till the time of use.
- The users should be divided into two categories, the first category is for those who use the website for operational use (Authority users), the second one is those who use the system as citizen users for non-operational uses.
- The System should suggest the most appropriate repair approach based on the distresses type, **only** to the authority's users.
- The system should allow users to register using a username, a password, email address and his job environment.
- The user should use his username/email as credentials for signing in.
- The user who uses the website with signing in or without registering to the website can access just what the default user can.
- The system should update the Registered citizen user once the maintenance has taken place.
- The system should update authorities once a new request is added by the citizens.