

Cypress Design Document

Overview:

Status: Completed version 1.0

Stakeholders:

Ademide Akinsefunmi (front-end GUI developer)

Kevin King (back-end developer)

Abigail Allen (back-end developer)

Shiva Sanjeev Kumar (back-end developer)

City of Toronto (Client and front-end users)

City Council (Back-end users)

Stakeholder Name	Class of use cases	Use cases	Description of use cases
Abigail	Use case related to System authorization	Login	<i>Login in to CYPRESS</i>
		Cancel	<i>Moving away from login page</i>
Abigail / Ademide	Use case related to the select language	English	<i>Move User to English page</i>
		French	<i>Move User to French page</i>
Abigail	Use case related to Registering	Enter Information	<i>User enter their information to register</i>
		Create Username	<i>User chooses a username</i>
		Create Password	<i>User chooses a secure password</i>
Abigail	Use case related to information change	Change Information	<i>User changes their information</i>
Shiva	Use case related to creating report	Create Report	<i>User chooses to create a report about a specific location</i>

Shiva	Use case related to editing report	Edit Report	<i>User chooses to edit the report that they have created</i>
Shiva	Use case related to deleting report	Delete Report	<i>User chooses to delete the report</i>
Shiva	Use case related to rankings of report	Rankings	<i>Rank each report based on the location and how many complaints of the same report has been received</i>
Shiva	Use case related to resolution	Report resolution	<i>System notifies city council about the report</i>
Kevin	Use case related to notification	Notification	<i>System notifies the user if their report has been taken in to consideration</i>
Kevin	Use case related to FAQ	FAQ Questioning	<i>System show a list of common question and answer for the user</i>
Kevin	Use case related to Contacting	Contacting	<i>System provides a list of contact information for the user</i>
Kevin	Use case related to logout	Logout	<i>System logs the user out and saves the last saved input</i>

This project aims to implement an automated system, for the city of Toronto, that provides its citizens the ability to report problems they notice on the streets and allows them, said citizens, to track the resolution of those problems.

Context:

The current process for reporting problems to the City Council is manual and tedious. Upon noticing a problem, a citizen has to find the right phone number to call, as different service

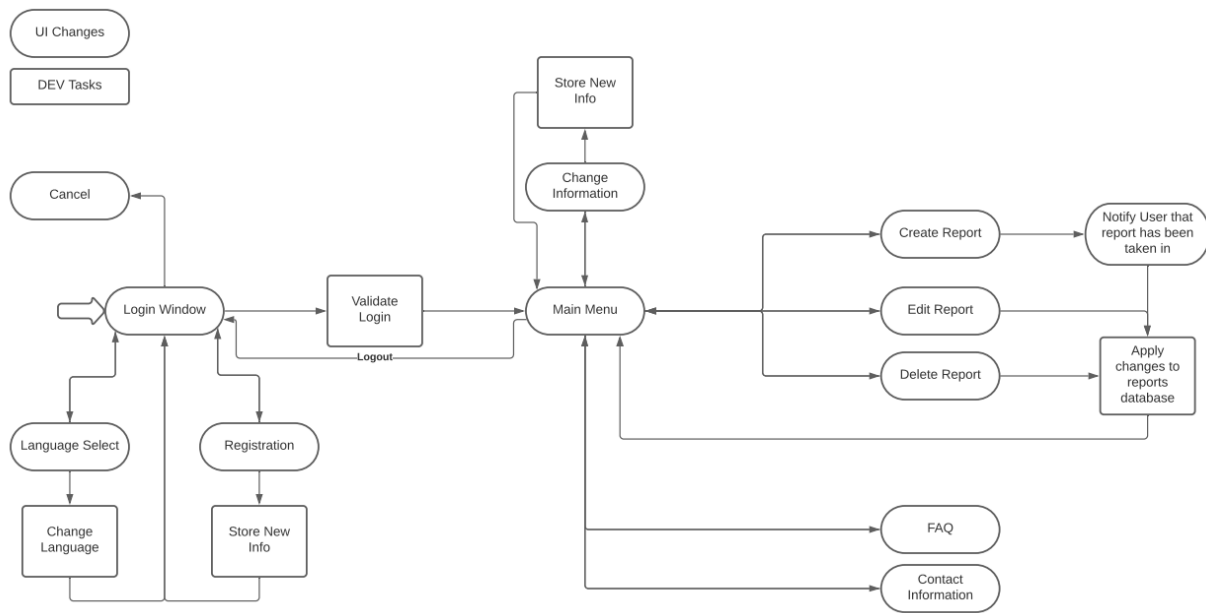
organizations deal with different types of problems. Furthermore, upon reporting a problem, there is virtually no way that a citizen can track the process of resolving the problem.

Goal

To create an application for that enables citizens of Toronto to report problems on they notice on the streets. The application will have the following functionality:

- to create/edit/delete an account to report/edit/delete problems and track their resolution.
- View a list of all reported problems ranked in terms of problematic areas and most frequent problems.
- Users will be able to post solutions to reported problems and vote on solutions proposed solutions.
- Users will have access to an FAQs and contact information for respective departments.
- Users will be able to “Tell a friend” about the application.

Flowchart



Problems

- Manual slow process of reporting problems discourages user engagement.
- Difficult for users to find correct firm/department to report problems.
- Lack of transparency on problem resolution.
- Lack of information of problematic area and most frequently occurring problem.

Milestones

March 9th: Planning and brainstorming on database architecture.

March 10th: Deciding on stakeholders for implementing application.

March 12th: Researching various types of GUI.

March 15th: Selecting GUI and planning around how data will be captured in GUI.

March 28th First Sprint implementation of application (version 0.1)

March 4th Second Sprint implementation of application (version 0.5)

March 11th Third Sprint implementation of application (version 0.7)

April 12th: Finish bug testing and GUI flow.