Water Service platform for underserviced communities

Chang Liu, Sneha Das

December 13, 2017

1 Introduction

We developed an android application to address the crowd- sourcing of the inspection and detection of water service malfunctions, in under- resourced communities. The application has options for civilians, who wish to report malfunctions and leakages, and plumbers to register with the KWSTF, to provide timely services to repairs, based on standard remuneration. Thus, this application is conceptualized to function as a mediator between the public, who thrive to maintain the water services for the community, and the plumbers available in the region. Not only does this accelerate the provision of services for maintaining water infrastructure when resources are constrained, but it could also helps in providing jobs and visibility to the plumbing community.

The application provides the users with two options: 1. to report leakages in the city 2. to provide plumber services. As a user reporting a leak, the app does not necessarily require the user to login or sign-up. However, a registered plumber with the KWSTF is required to have a profile which is made possible by signing-up and login-in to use the app, because based on the work history and feedback received by a plumber, their profile is updated, which adds a level of credibility of the plumber. The application begins with the home-page, mainly comprising of two buttons, for *reporting* and *registering*. Also, if the user already has an account, they have an option to *login*. If the user wishes to report a leak, they choose *report*, which re-directs them to the report form. This form requests all the details required to look for a plumber with potential to fix the malfunction. The user has options to *send report* or to *cancel* the action. If the user chooses *cancel*, they are taken to the home screen. On pressing *send report*, a pop-up is presented to the user with confirmation that the report was sent. The user can dismiss the pop-up by pressing *OK* on the pop-up, following which the screen with details of the expected arrival time of the plumber is displayed to the user. This view is designed to contain a counter, which is indicative of the plumber's arrival. In the completely functional app, the counter should be initiated and controlled by backend-coding and search and find action by the KWSTF to locate a plumber. When a plumber is found, the corresponding details are also displayed on this screen. Finally, when the plumber has fixed the issue, this information is updated to the user under *My report*. At any time after submitting the report, the user can access the status of the complain under *My report*. Also, in all the screens, a navigation screen is provided for users to go to the *Home*, *Setting*.

To register as a plumber, the user chooses *register* on the home screen, following which they are redirected to the registration form. this form seeks the basic details to evaluate the suitability of the plumber and the resource they can be. After submitting, the user is notified of the time the process could take and the contact details to get back to KWSTF. During the evaluation process, the plumber can stay updated about the application status by visiting *My profile*. The possible outcomes of the registration request are *Pending*, *Approved*, *Rejected*. If the registration is approved, then the details provided by the user is saved as the user profile, work history. etc. The back-end procedure for registration should include a rigorous vetting process and security-check among other evaluation criterion.

2 Background

In the design of the User-Interface, we have explored the following guidelines:

1. **Four Principles of Good Design (Shneiderman 1998)**
    - State and the action alternatives should be visible
    - Should be a good conceptual model with a consistent system image
    - Interface should include good mappings that reveal the relationships between stages
    - User should receive continuous feedback
2. **Nielsens Ten Heuristic Rules (1993)**
    - Simple and natural dialog
    - Speak the users language
    - Minimize users memory load
    - Consistency
    - Feedback
    - Clearly marked exits
    - Shortcuts
    - Good error messages
    - Prevent errors
    - Help and documentation
3. **Jakob Nielsens Ten Usability Heuristics**
    - Visibility of system status (Feedback)
    - Match between system and the real world (METAPHOR)
    - User control and freedom (NAVIGATION)
    - Consistency and standards (CONSISTENCY)
    - Error prevention (PREVENTION)
    - Recognition rather than recall (MEMORY)
    - Flexibility and efficiency of use (EFFICIENCY)
    - Aesthetic and minimalist design (DESIGN)
    - Help users recognize, diagnose, and recover from errors (RECOVERY)
    - Help and documentation (Help)
4. **Shneidermans Eight Golden Rules**
    - Strive for consistency
    - Enable frequent users to use shortcuts
    - Offer informative feedback
    - Design dialog to yield closure
    - Offer simple error handling
    - Permit easy reversal of actions

- Support internal locus of control
- Reduce short-term memory load

5. **Smith & Mosier: Data Display**
    - 2.0/1Necessary Data Displayed
    - 2.0/2 Only Necessary Data Displayed
    - 2.0/3 Data Displayed in Usable Form
    - 2.0/4 Data Display Consistent with User Conventions2.0/6 Consistent Display Format
    - 2.0/8 User Control of Data Display
    - 2.0/12 Familiar Wording
    - 2.0/15 Consistent Grammatical Structure

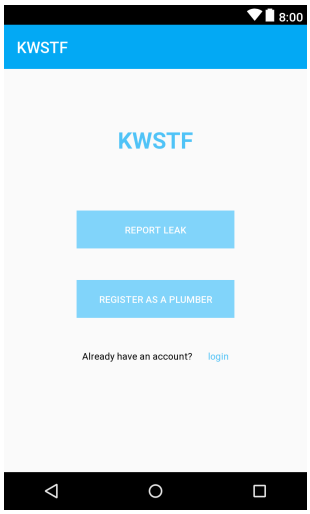
3 Methodology

The tools we are using includes design tools and software engineering tools. For the design tools, we used paper and hand drawing to draw the framework of the initial pages, and then we organize the initial pages together to a flow diagram, with a clear indication of how the pages are interacting with each other. For the engineering tools, we used Android Studio to put the design we made into an android project. We used xml language to define the layout of the page and then we used Java language to create the function behind the layout. Following are the details of the guidelines with illustrations of how they are reflected in or UI-design

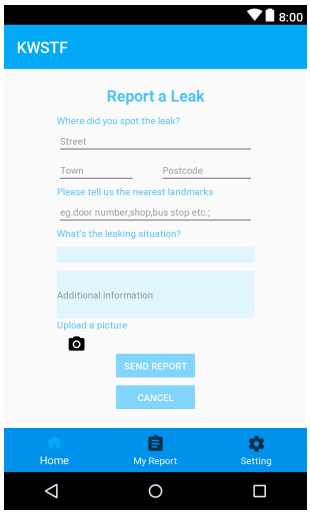
3.1 Guideline 01

Four Principles of Good Design (Shneiderman 1998):
2. Should be a good conceptual model with a consistent system image.

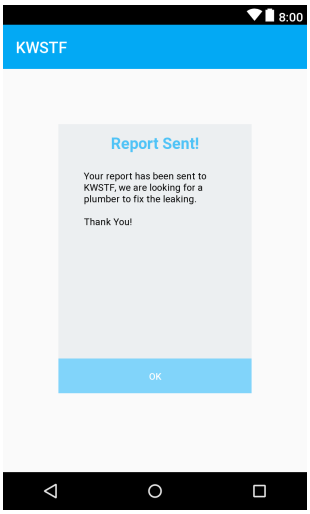
Implementation	
Content Items	**Style**
Page Title	20sp, dark blue color #4FC3F7, bold
Page content subtitle	12sp, blue color, #4FC3F7
Page content plain text	12sp, black color #000000
Buttons	background: blue color, #81D4FA text: 12sp, white color #FFFFFF
Text Links	12sp, blue color, #4FC3F7
Input Text Box	background: light blue color, #E1F5FE
Dropdown list	background: light blue color, #E1F5FE
Radio Buttons	12sp, black color #000000
Contact card	background: light blue color, #E1F5FE



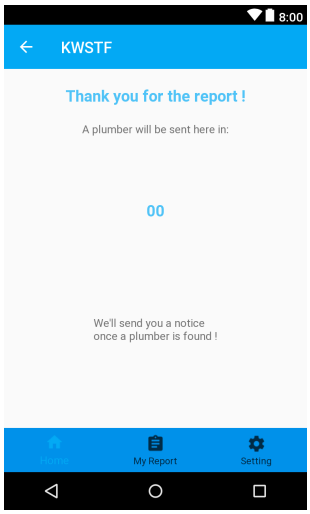
(a) Home screen



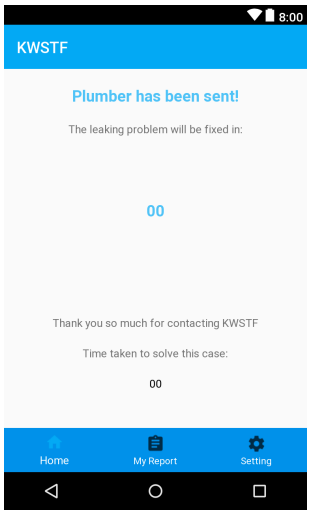
(b) Reprot leak form



(c) popup confirmation

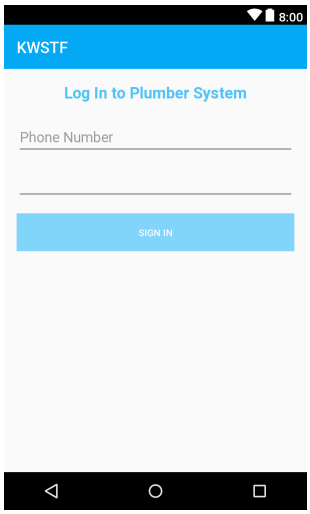


(d) Status of complaint (i)

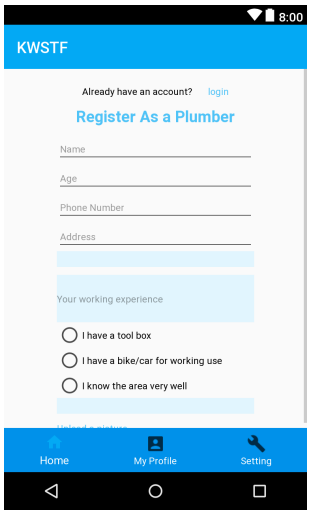


(e) Status of complaint (ii)

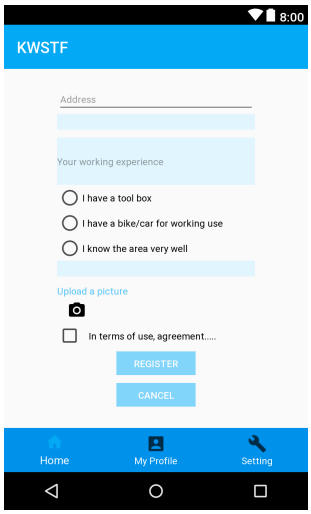
Figure 1: Guideline 01- Report Leak



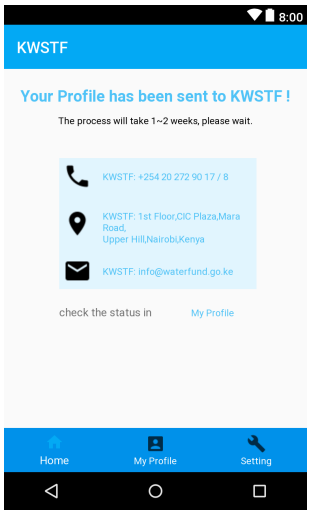
(a) Login screen



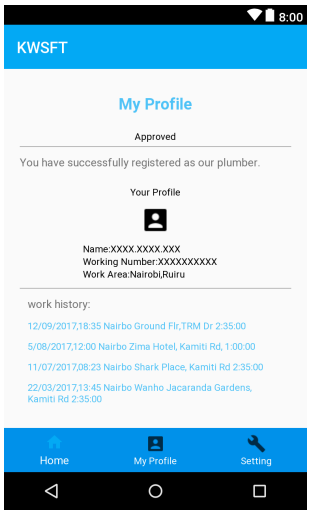
(b) Registration as plumber (i)



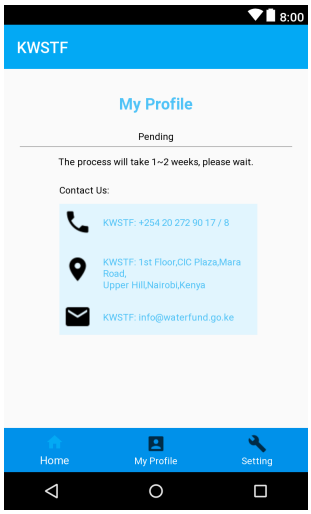
(c) Registration as plumber (ii)



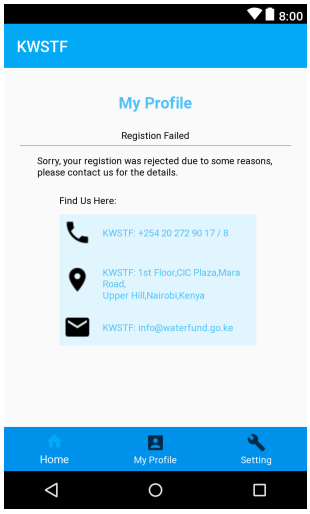
(d) Registration sent



(e) Registration approved

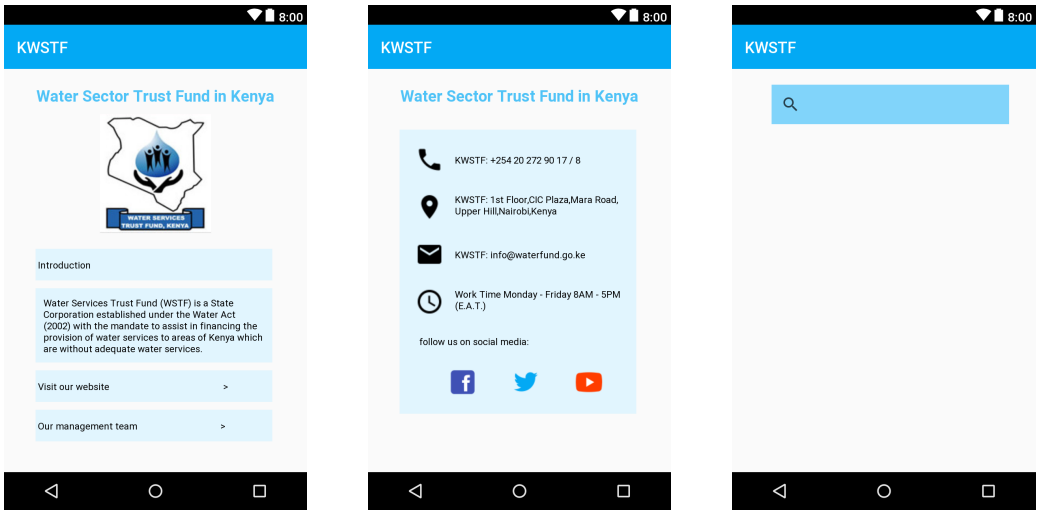


(f) Registration pending



(g) Registration rejected

Figure 2: Guideline 01- Register as a plumber



(a) About (b) Contact (c) Search

Figure 3: Guideline 01- Menu

3.2 Guideline 02

Four Principles of Good Design (Shneiderman 1998):
4. User should receive continuous feedback.

Implementation:

- Popup-window
- Time-track and Estimated Time Notice

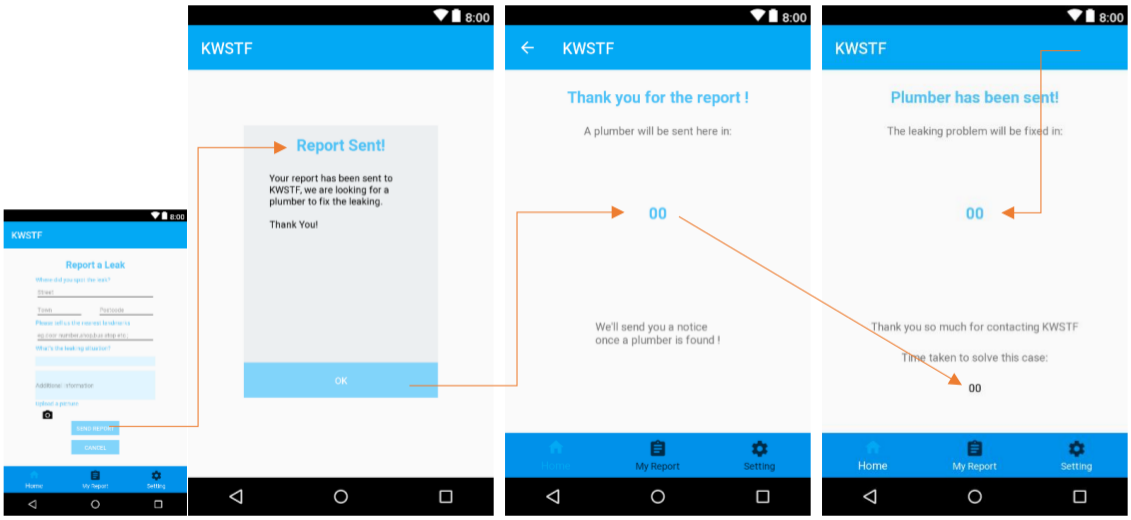


Figure 4: Guideline 02- When the user want to report a leak and right after the user hit the Send Report button, a pop-up window will be activated and displayed on My Report page, give the user positive respond that the report has been sent. At the same time, the timer was sent to track the time that has passed since the user submitted the case, so the user could have a clear image that how fast the KWSTF are responding to the report. After KWSTF has found a plumber, My Report will be updated, a Count-down timer will be activated according to the time the plumber estimated to spend on fixing the leak, so the user could be clear that the leak will be fixed in a specific time. At the same time, the previous timer will stop counting and display the time used to find a plumber.

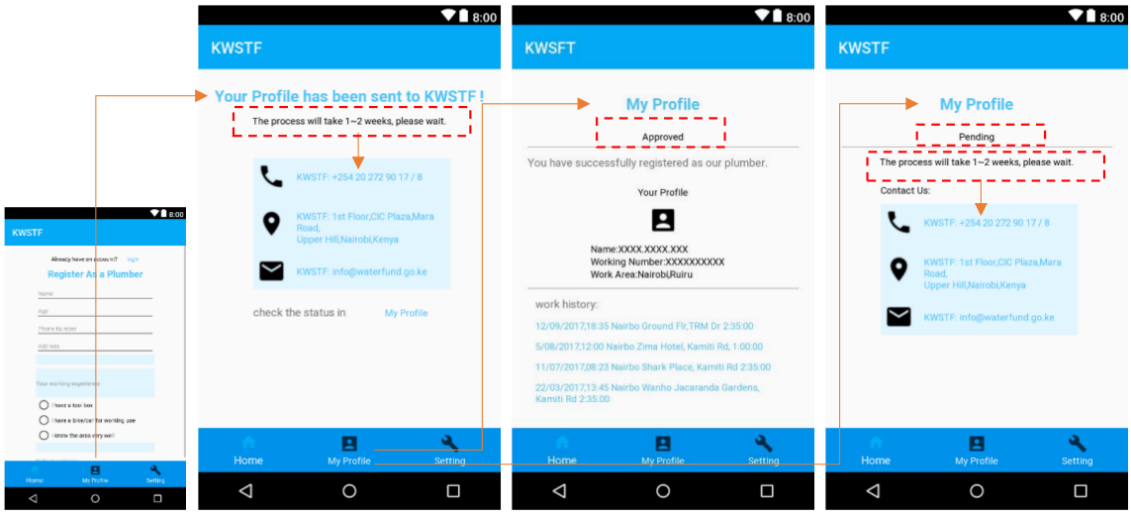


Figure 5: Guideline 02- After the plumber has submitted the registrations form, the user will be sent to My Profile, with a notification on the title saying the profile has been sent to KWSTF, the estimated time for the process will be displayed under the title, so the user would be clear when he could get the respond from KWSFT, besides that, the contact information are displayed under, the user could choose to call or send an email to inquire about his case. When a decision is made by KWSTF, My Profile page will be updated, with Approved or Pending or Rejected status displayed under, if its still in Pending, KWSTF contact info are available on the page for user to take the next step.

3.3 Guideline 03

Nielsens Ten Heuristic Rules (1993):
1. Simple and natural dialog 2. Speak the users language 3. Minimize users memory load 7. Shortcuts
Implementation:

- Bottom Navigation Bar
- Menu Items
- Page Update

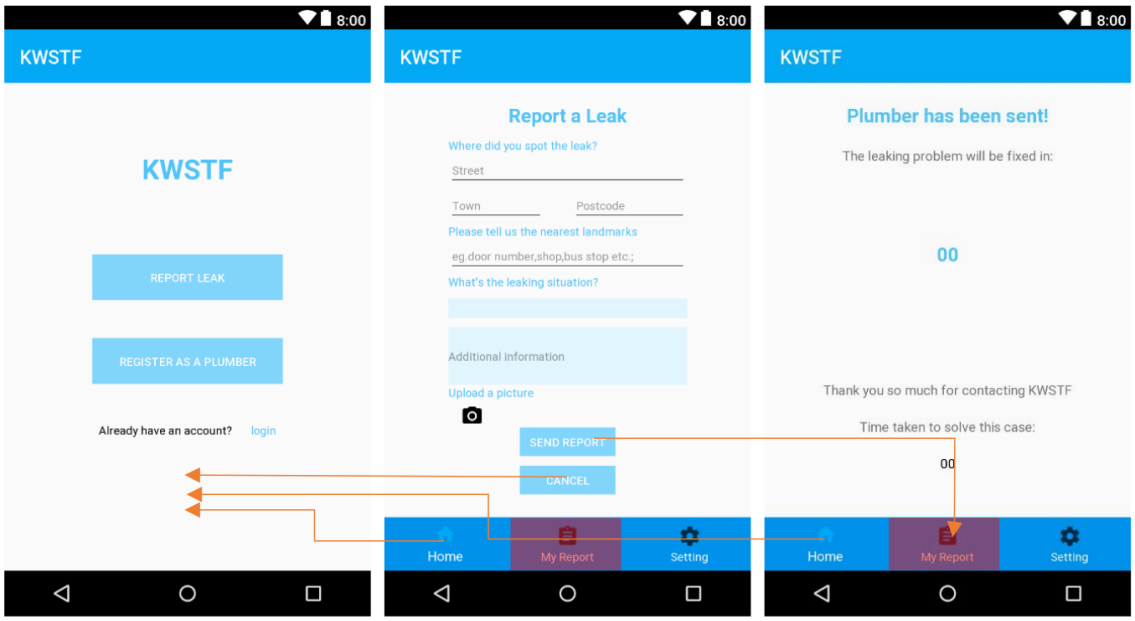


Figure 6: Guideline 03- The app use users language and give users hint with blue text to prevent erros. Bottom Navigation bar is the shortcut for the pages. User can always go back to home page and start over again. After the report was sent, it will be stored in My Report, users could go to the page and check the updates.

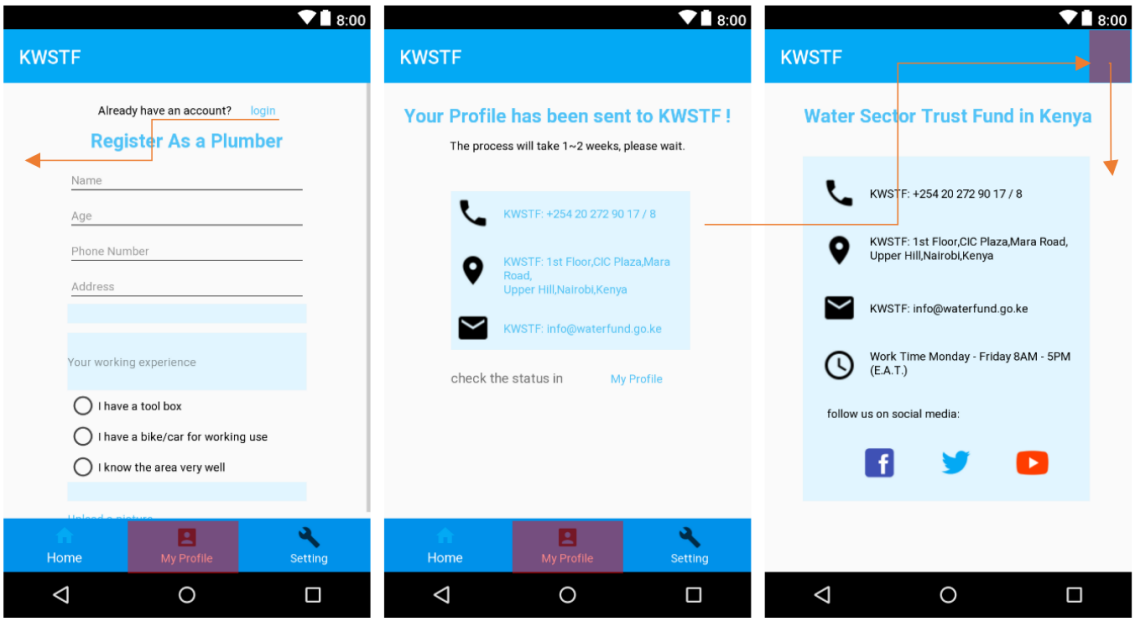


Figure 7: Guideline 03- Login link was listed on top of the page, so if the user already have an account, he can use the shortcut to login. The KWSTF contact card will be displayed every time there is a need, and the contact info are listed in the menu items, users can also use Menu to quickly locate the contact info.

3.4 Guideline 04

Jakob Nielsens Ten Usability Heuristics:
2. Match between system and the real world (METAPHOR) 6. Recognition rather than recall (MEMORY) 8. Aesthetic and minimalist design (DESIGN)
Implementation:

- Drop-down list
- Radio Button

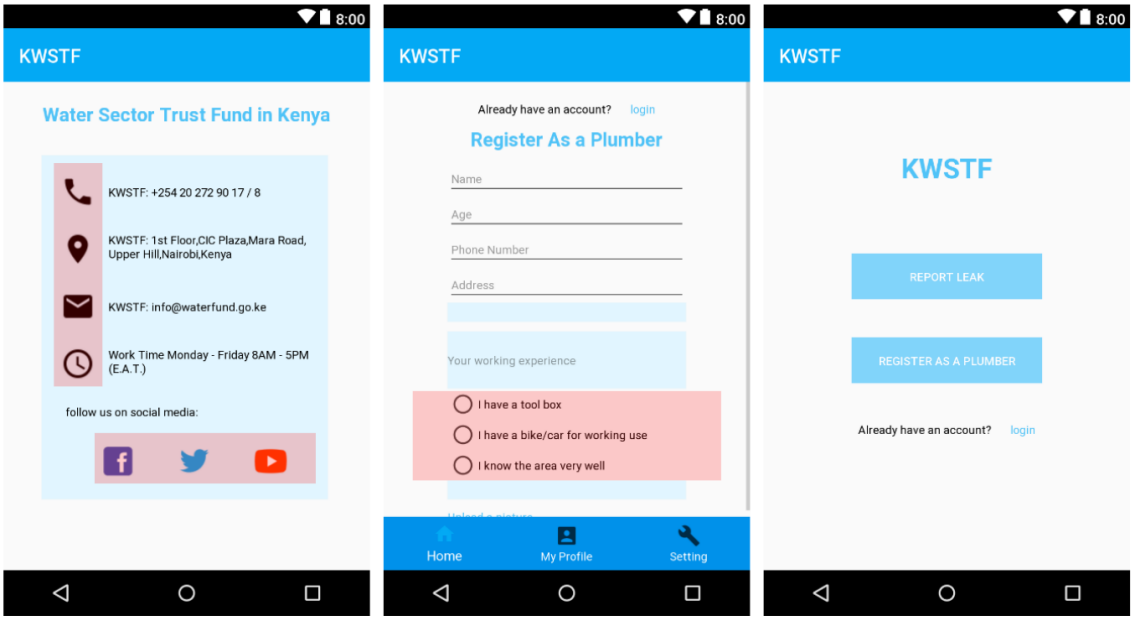


Figure 8: Guideline 04- A lot of Icons are used to make the app works closely with users real life. When we need to have the input from users, we provide Dropdown menu or Radio Button, so the users dont need to recall but only to choose from the current. The design have only one main theme color: blue, when there is a need to distinguish, different blue tone is used instead of other colors, which makes the app design is simple and not color-blind users.

3.5 Guideline 05

Smith & Mosier: Data Display
2.0/13 Consistent Wording 2.0/15 Consistent Grammatical Structure Implementation:

- Grammatical Structure goes like: Report/Register + A leak/ As a plumber
- Wording goes like: I have/know + tool box/the area

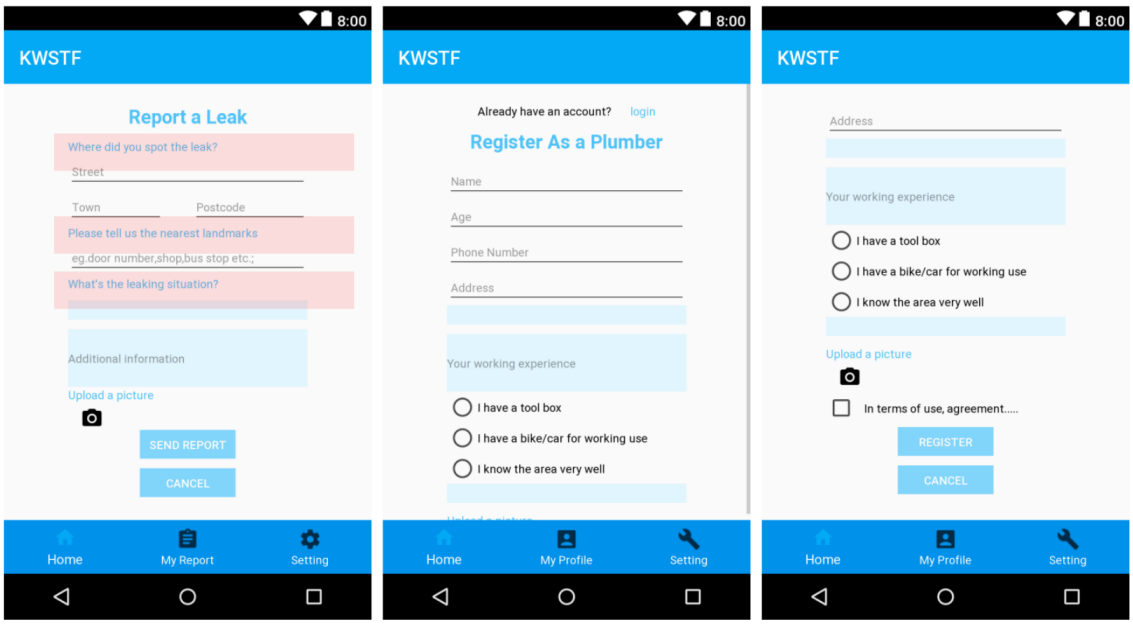


Figure 9: Guideline 05

4 Result

Based on the conceptualization of our application, we successfully implemented it in android-studio, front end User design and backend coding to describe the flow of activities in the application. A Video of the functionality of the application is provided in the submission folder, along with the code and the report.

References

[1] Google. Android developers, 2017.

[2] Google. Material design: Color - style - material design, 2017.

[3] Jakob Nielsen. Ten usability heuristics, 2005.

[4] Ben Shneiderman. Shneiderman's eight golden rules of interface design. *Retrieved July*, 25:2009, 2005.

[5] Ben Shneiderman. *Designing the user interface: strategies for effective human-computer interaction*. Pearson Education India, 2010.

[6] Sidney L Smith and Jane N Mosier. *Guidelines for designing user interface software*. Mitre Corporation Bedford, MA, 1986.

[7] Bruce Tognazzini. First principles of interaction design. *Interaction design solutions for the real world, AskTog*, 2003.