Shopping Trolley Monitoring Dashboard UX Case Study

By Michael Ong Ming Hao

Executive Summary

My team has successfully created an entire IoT Application to solve the issue of unreturned shopping carts. My role as the Front-end developer and UX designer was to create an intuitive shopping cart monitoring dashboard for its users and notify them if there someone was going to take a shopping trolley out of a shopping mall. I managed to create a prototype for the users to test and tweaked the dashboard according to their feedback and learnt a lot about UX design and dashboard visualisation. You can view the copy of the dashboard here.

The Situation

Unreturned Shopping trolleys is an issue with many supermarkets worldwide. Many people take them out of supermarkets' premises unreturned, causing hundreds of thousands of dollars spent on trolley recovery and replacement. My team was tasked to figure out a solution for people having to take the shopping trolley out of the shopping mall.

My role in the team is both a Front-end developer and UX designer, working closely with the Product manager of the team.

Users of the Application

- Supermarket Enforcement Officers - like security officers, these people are employed to ensure that the shopping trolleys are not misused
- Supermarket Enforcement Supervisors – these people are supervising the supermarket enforcement officers, usually to inform them of potential misusers.

Design Process

Here is the design process that I have undertaken by order, following design thinking process:

1. Gather all the requirements from the client. (Discovery and Define)
2. Identify the main users, and their priorities. (Discovery and Define)
3. Wireframing and Usability testing. (Ideation)
4. High Fidelity Prototyping and user acceptance test (Prototype and Validation)

Discovery and Define Stage

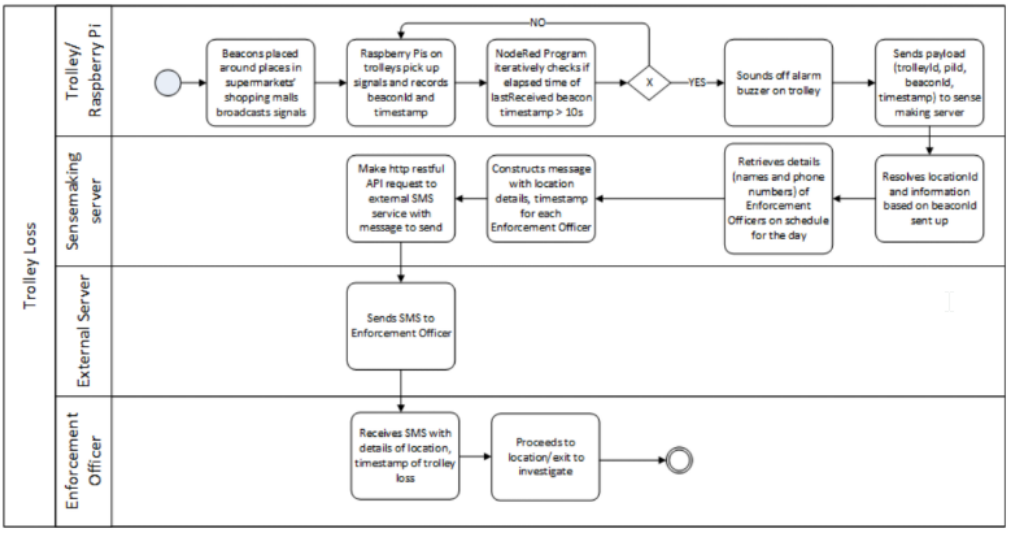
In the discovery and define phrase, we wanted to learn more about how the users are currently dealing with misused shopping carts. I conducted an interview with multiple supermarket enforcement officers and supervisors. This interview was great in helping the team understand the entire flow and the inefficiencies of the current approach. In addition, it also helped me to gain a deeper understanding of their pain points and behaviours.

During interview with the client, here are the main priorities that the users hopes to achieve.

- Ease of understanding the analysis via the dashboard as the people using them are not tech-savvy.
- Ability to look at past data to gather more insights in the future
- Having the dashboard easily scalable to other supermarket outlets in the future.

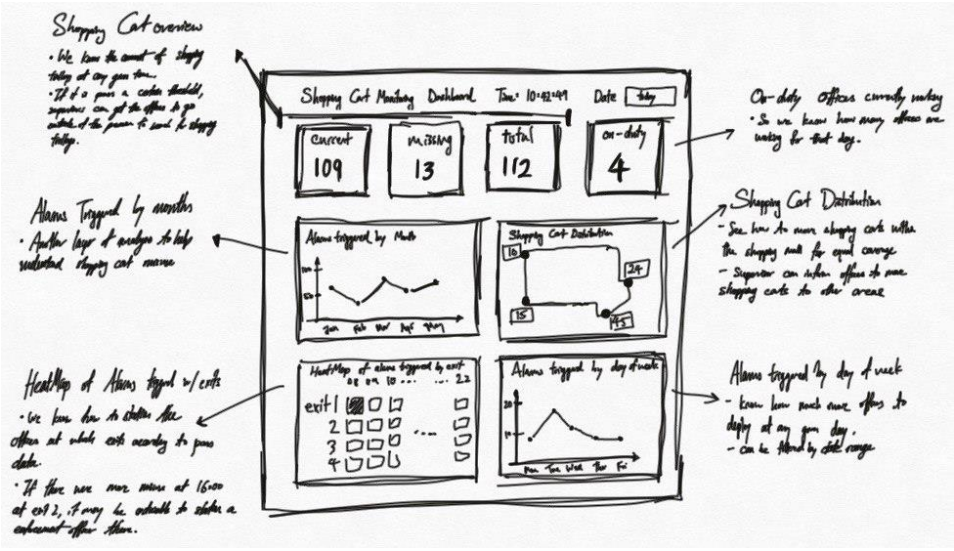
Wireframing and Usability testing (Ideation)

After the discovery and define stage, we have a clear picture of how we would want the entire system to flow. We created a user flow map of what we hope to achieve, and I set up to do the wireframing based on the requirements gathered from the client meeting and interviews.



Targeted Process flow of a shopping trolley misuse

Based on the requirements from the client meeting, here is the wireframe that I came out with.



There are 6 main parts of this dashboard:

Shopping cart overview cards. This visualisation shows the total number of shopping carts, missing shopping carts and number of shopping carts in the mall. This would also help the supervisor decide on the threshold to make better use of the officers to go out and search for unreturned shopping trolleys.

Alarms triggered by months line graph. Whenever a shopping cart is taken out of the shopping mall, it would trigger and alarm. This visualisation tracks the number of alarms triggered by month. This would help the supervisor to see trends, if any.

Heatmap of alarms triggers by exit. This visualisation breakdown the number of alarms triggered in each exit for that day. This visualisation would give an idea of how to position the officers so that they can stop people from removing shopping carts from the premises.

Shopping cart distribution area map. This visualisation would show the number of shopping carts at each shopping cart dispensing point. This would give the supervisor the full picture of all the shopping carts around the shopping mall. Resulting in making good decision in informing officers to move shopping carts from one area to another.

Alarms triggered by day of week line graph. Whenever a shopping cart is taken out of the shopping mall, it would trigger and alarm. This visualisation tracks the number of alarms triggered by day of week. This would help the supervisor to see trends, if any.

Off-duty Officers current working card. This visualisation shows the number of officers that is currently on duty. So that the supervisors knows how many officers he can utilize at any given period.

After I am done with my wireframing, I conducted a usability testing with my team and other participants to play the persona of both the supermarket enforcement officers and supervisor to figure out ways to improve the dashboard. I have collated their findings, and here are the top 3 feedback that I have received.

Our Usability testing

Feedback 1 – Include a dashboard to track shopping trolleys battery power. Since we are using a raspberry PI and an external battery to track shopping trolleys, by having a dashboard to track the power of the shopping trolleys, we know which shopping trolley we need to charge.

Feedback 2 – The visualisation tracking the number of officers on-duty is too simple. Since the supervisor would have to call the officers frequently to do a certain task, it would be better to include more information like who is on duty and the mobile numbers used to contact the officers.

Feedback 3 – Both graph tracking alarms triggered by months and day of week can be combined into one. Since they are showing similar metrics, it would be better to have them combined into one visualisation where the user can filter the way they want to view the information.

Prototyping (High-Fidelity) and User Acceptance Test

Before moving into the prototype,

Architecture

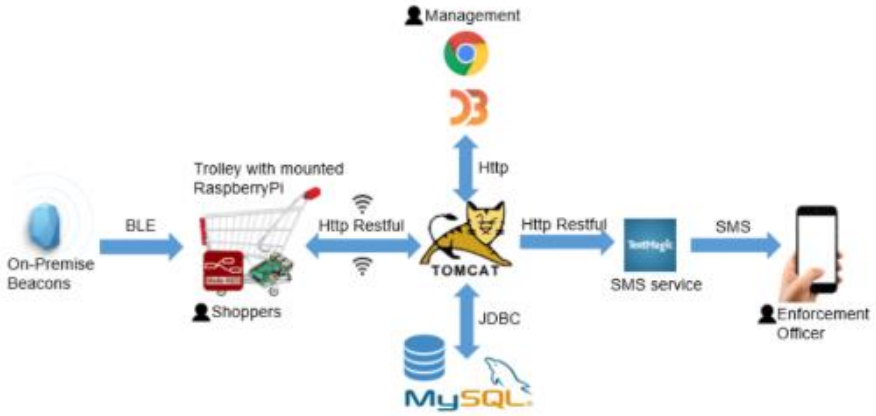


Figure 2: System Architecture

The diagram above shows the entire system architecture of our IoT solution. Starting with Bluetooth beacons, it would track each trolley with a mounted raspberryPi. This raspberryPi would constantly communicate to a tomcat server via a REST API which would update both the database and the monitoring dashboard. The constant stream of data would be consumed by the dashboard to provide live analysis.

User Acceptance Test

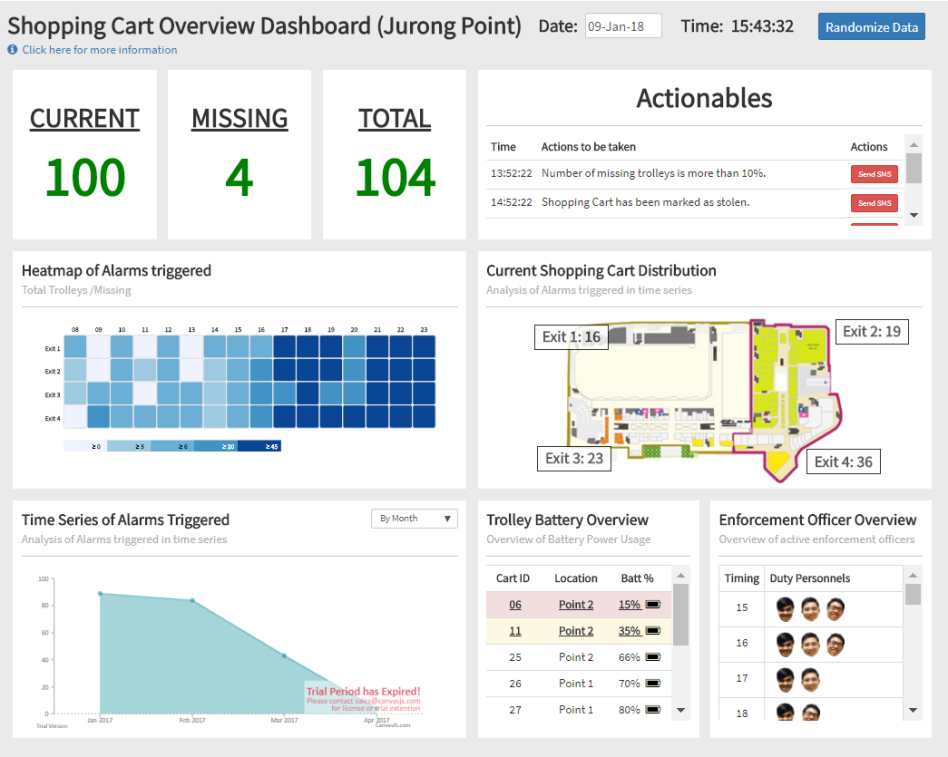
After we were done with the entire application, we showcased to the client for an entire user acceptance test. The client was very happy with our solution as it met all the users' needs, requirements and business processes. However, they suggested some improvements to the system, here are the top feedback that we have accepted and adopted.

Feedback 1 – The simple graphs that we used were still not easy to understand. The graphs that we used were easy to understand for tech-savvy people, but some users were not tech-savvy and did not know how to understand or analyse the dashboard. To counter this, I've added an actionable table to tell the users what to do based on certain metrics/parameters that we set. This worked great as everyone would know what are some actions that they can take for example

Feedback 2 – The simple graphs were still not easy to understand (2). On top of the actionable table, we have also decided to colour coded to a traffic light. Having the graph be at green means there is no action that needs to be done. If the graph changes to yellow, it means that if something is not done, it may be an issue. Finally, when a graph has turned red, means there should be a rectifying action.

The Shopping Cart Overview colour coded as green

Final Product



Outcomes & Learnings

From this project, I learnt a lot about UX design and data visualisation. Firstly, it's about asking the right questions can really help make your project that much better. The questions that was asked during the initial client meeting and the user acceptance test brought great insight to building a better dashboard for the user. Next, this is the first time that I am working with non tech-savvy users, I learnt how to make visualisations easy to understand and how they can gather their own insights. Lastly, an actionable table would be great in all my dashboards. This would reduce the number of analysis that a user has to make and decide to go through with the recommended actions provided.