Final Project: Report

Rajat Varma

April 7, 2024

1 Problem

The given problem is that of developing a survey web page tailored for a recently introduced smartwatch, which involves presenting users with a set of questions focused on aspects such as aesthetics, battery performance, raise-to-wake accuracy, notifications, and overall comfort. Respondents will express their opinions using a Likert scale.

A Likert scale is a psychometric scale which is commonly used in research questionnaires and is the most widely used approach to scaling responses in survey research. We shall be using a numerical scale from 1 to 5, indicating the user's satisfaction with the feature.

The solution requires a locally hosted webpage that asks the respondents for the following questions:

- Their name

- Their email address

- Satisfaction with look and feel of product

- Satisfaction with the battery backup

- Satisfaaction with the raise to wake feature

- Satisfaction with notification support

2 Solution

2.1 Architecture

The design of the solution will have the following components:

- Locally hosted website: This website will have two webpages:

    - A webpage that the user interacts with, where they respond to the survey.

- An admin dashboard that displays the statistics of the collected responses, and where a comprehensive list of all the collected responses in a `.cav` format.

- An SQLite database that stores all the responses.

2.2 Tools and platforms

The following tools and platforms were used in the development of the survey webpage:

- **Haskell:** Haskell is a popular purely funcitonal programming language and will serve as the primary language in which this project is written.

- **Yesod:** Yesod is a webpabge building framework built for Haskell, and helps us abstract out many of the important aspects of writing webpages in Haskell, including generating monads for type-safe routes and I/O

- **HTML/CSS/JavaScript:** These are the core technologies for building web pages. HTML (HyperText Markup Language) was used for the structure of the web pages, CSS (Cascading Style Sheets) for the styling, and JavaScript for the functionality. In the context of Yesod, we shall be using the languages of Hamlet, Cassius and Juliet, that are modified versions of HTML, CSS and JavaScript, that aid with the communication between Yesod and the webpage.

- **SQLite:** This is a C library that provides a lightweight disk-based database. It was used to store the responses from the survey.

2.3 Implementation

The implementation of the survey webpage involved the following steps:

1. **Designing the Survey Form:** The first step was to design the survey form using HTML and CSS. The form included fields for the respondent's name, email address, and their satisfaction with various features of the smartwatch.

2. **Handling Form Submissions:** When the form is submitted, a POST request is made which stores the collected data in the SQLite database.

3. **Creating the Admin Dashboard:** The admin dashboard will display the statistics of the collected responses and provide a comprehensive list of all the collected responses in a CSV format.
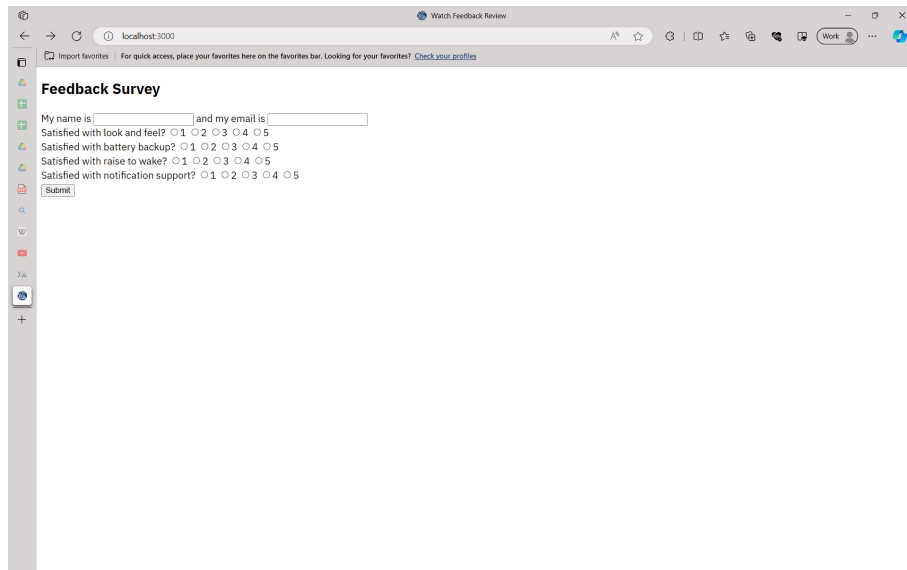
Figure 1: Screenshot of the prototype in its current state

2.4 Current Prototype

The webpage in its current form only has the bare-bones HTML, which collects the required data from the user. (Figure 1) Once the user has filled out the form, they must click on the submit button. (Figure 2) In the absence of any next steps or response, in its current state, upon submission the webpage simply displays the collected response. (Figure 3)

When the user clicks the submit button, Yesod's Persistent functionality stores the value in a local SQLite Database (Figure 4). Given below is the structure of the DB, which contains a singular table named "reviews".

Name	Email	Q1	Q2	Q3	Q4
TEXT	TEXT	INT	INT	INT	INT

Table 1: The structure of the table 'reviews' in the DB

Additionally, an administrative dashboard webpage has been created where, in the meantime the administrator of the survey can view all the responses to the survey. (Figure 5)

2.5 Next Steps

Firstly, the main functionality of being able to download the reviews in the form of a `.csv` file. That shall be achieved by simply querying the database, construcing a string in the CSV format, write it to a file and download it locally.

Figure 2: Screenshot after the user has filled out the form

Secondly, the UI of the website needs to be improved, making it easier to use on both the user and administrative side. Figures 6 to 11 contain the tentative UI mockups for the improved UI.

2.6 Testing Plan

The primary testing will revolve around the ability of the system to deal with DOS-like attacks, along with SQL injection and XSS (Cross-Site scripting attacks). The Selenium WebDriver client for Haskell will be used to simulate the above mentioned attacks, along with a range of other possibly system-breaking inputs containing special characters and emojis. The reports from the testing will provide a comprehensive list of possibilities that input sanitisation needs to be immplemented for.
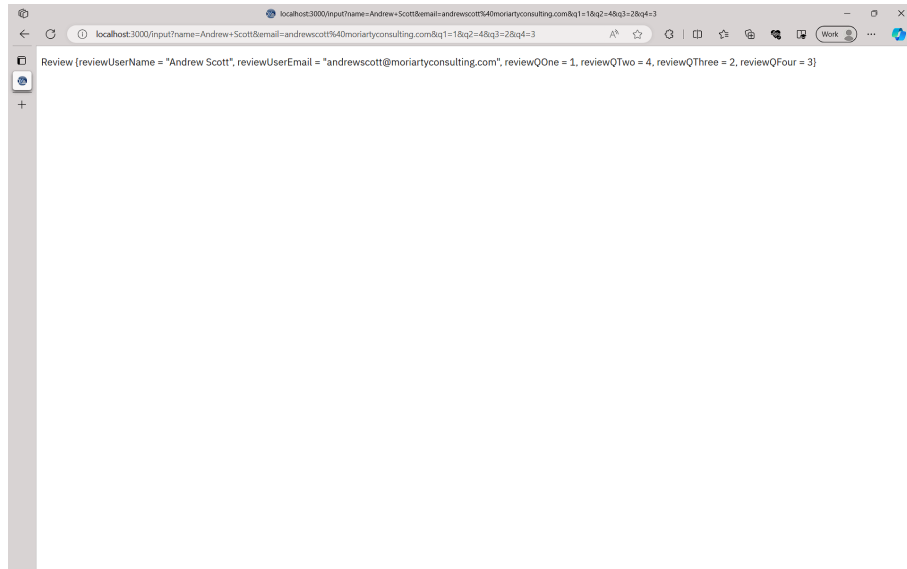
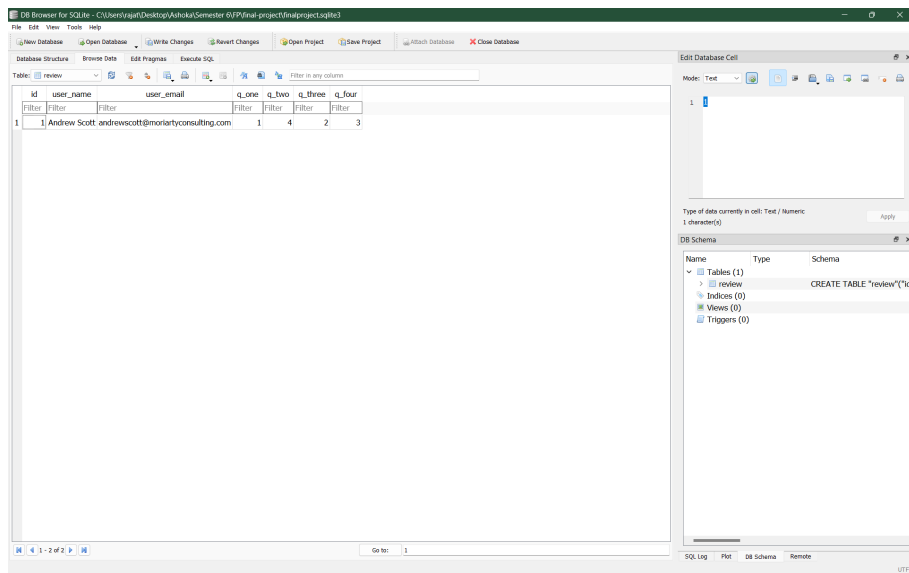Figure 3: Screenshot after the user has submitted the form

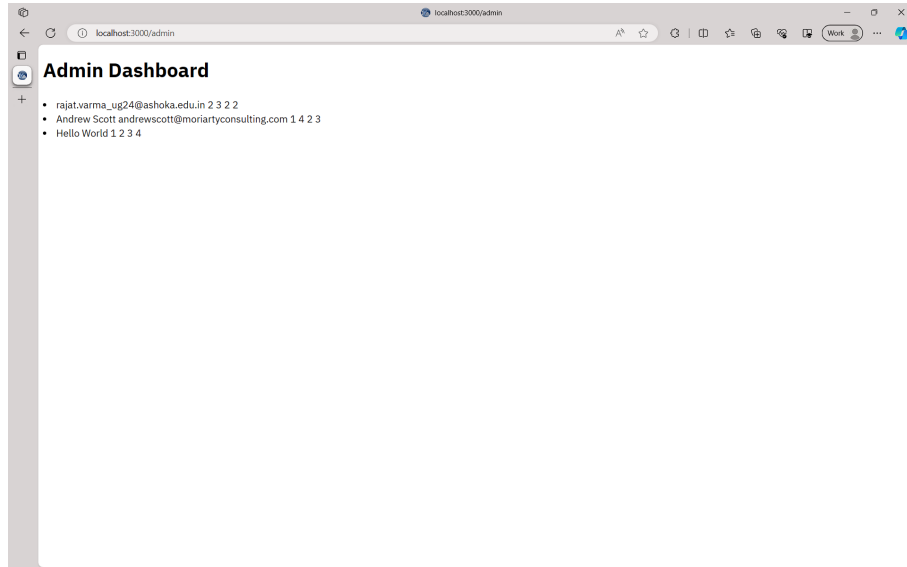Figure 4: Screenshot of SQLite database where the form response can be seen

Figure 5: Screenshot of admin dashboard webpage

s m a r t®
W A T C H™

Feedback Survey

About you

What's your name?

_____

What's your email address?

_____

[Next]

---

s m a r t®
W A T C H™

Feedback Survey

About WATCH™

Satisfied with look and feel?
◯ **1**　◯ **2**　◯ **3**　◯ **4**　◯ **5**

Satisfied with battery backup?
◯ **1**　◯ **2**　◯ **3**　◯ **4**　◯ **5**

Satisfied with raise to wake?
◯ **1**　◯ **2**　◯ **3**　◯ **4**　◯ **5**

Satisfied with notification support?
◯ **1**　◯ **2**　◯ **3**　◯ **4**　◯ **5**

[Next]

s m a r T® W A T C H™

Feedback Survey

Thank you!

We may reach out to you via email to know more
about your experience with Watch™

---

s m a r T® W A T C H™

Admin Dashboard

Insights

Total no. of responses collected: **35**

Response Sentiments

40%: Very Positive
20%: Positive
15%: Average
3%: Negative
22%: Very Negative

Average Ratings

Look and Feel	**4**
Battery Backup	**3**
Raise to Wake	**2**
Notification Support	**1**

Download responses as CSV