



High-Watch

Prepared for: Alexander Babanski, Dr.

Prepared by: Benjamin Sykes, Malik Sheharyar Talhat, Liam Stickney

February 26, 2019

HIGH-WATCH

INTRODUCTION

Our motivation behind this application was, as frequent drivers ourselves, when we identified the need for a new and convenient way of tracking weather conditions on nearby roads and major highways that would save us time and stress. We knew this application would need to be intuitive and easy to use, but also very descriptive in order to give the user the best and most helpful experience possible.

Introducing High-Watch, the new, easy to use application that allows users to easily find and track nearby roadways based on their current locations, including the ability to check the clarity of the roads, visibility, if the roads are clear or not as well as live footage of the roads in real time. Users will be able to save their most travelled roads for a personalized experience which allows the user to easily find their saved roads. Overall, High-Watch is designed to remove the stress out of daily commutes and make the drive to work or school a smooth one.

MOTIVATIONS

This problem was first identified when, on several occasions, the traffic and weather conditions of highways we frequent were much different than we expected during certain times of the day. Sometimes, simply looking up the information online was not 100% accurate to the actual conditions of the roadways at that specific time, leading to either long, traffic-jam filled commutes, or dangerous slippery ones. High-Watch is designed to be able to successfully track roads in real-time. To track a road, a user can either search by the road's unique ID, road name, or the nearby coordinates of a road so the user is able to successfully see the conditions of the road relative to their current location.

The issue of inaccurate road conditions is one that affects us all at one point or another. At seemingly any given time during the day, commuters pack up into traffic jams along the highways that delay everyone's commute and generally cause unnecessary stress to those

HIGH-WATCH

involved. With the High-Watch app, user's can help better plan their commutes to avoid the bustling stress of a crowded highway.

While there is a solution to this problem, either simply listening for the road conditions over the radio or checking the conditions online, both have their own issues. Firstly, listening to a radio usually limits the road updates to once every 15 to 30 minutes. If the user in question is in a hurry, they often won't have time to wait the extra 15 to 30 minutes to get their traffic updates. Secondly, looking the traffic up online sometimes is very vague and unorganized, often giving you information on roads either not related to your desired location, or not enough information on the roads the user actually cares about. With the High-Watch, we remove both of these convenience issues, allowing for instant access to the roads the user cares about as well as much more specific and easy to read conditions.

High-Watch will be usable for any commuter in Ontario, at any time of day. High-Watch will remove all of the unnecessary hassle of commuting and increase the convenience of the traditional ways of getting one's traffic updates. We hope the allure of a more convenient commute will attract users from all across the province, and hope those who use it feel it has helped save them from the once routine long morning commutes.

PROPOSED SOLUTION

Architecture and Environment

As previously mentioned, the goal of the app is to provide the user with an easy to use and simple applications that will deliver their traffic updates with precision and speed. To target as many users as possible, High-Watch will run on both Android and Apple devices.

In order to provide the convenient application as possible, we will be using the typical HTML5, CSS3 and JavaScript techniques, as well as the JQuery framework for JavaScript to

ease the complexity of the application. Well will also be utilizing Bootstrap in order to create a clean and appealing interface, one that will standout to the user and provide them with a user-friendly experience. As for the hardware of the devices it will run on, the application will utilize the user's location in order to get the road conditions of roads closest to the user, in an effort to get more personalized road updates. Other than that, we will also utilize the notification feature present on mobile devices to give specific updates to the user based on their most frequented roads, such as extreme weather conditions and high traffic conditions.

In order to get the traffic conditions and updates accurately, we will be utilizing the [511on.ca](#) web API. The API uses HTTP get requests as well as specific parameters to accurately get certain traffic conditions such as visibility and clarity, which are returned as strings. We will also be using this to get the feed of the traffic, which we will be displaying to the user. To give each user their own unique experience, we will be utilizing Google Firebase database to store the unique ID's of each road the user selects, which returns that road's specific information. These saved pieces of information will be available in their own specific menu.

To code this application, we will be using IntelliJ software with NodeJS. We will create backups of our work using GitHub, which will allow us to not only ensure that our work is always safe, but so we may all collaborate on the project easier. We will also use Apache Cordova, which is a mobile development framework, to meld the project together.

In terms of project production, we have utilized Google Firebase before, so we have background knowledge of how Firebase functions. We have not, however, utilized the [511on.ca](#) API before, so we will have to experiment on what we can obtain from the API and how we can go about getting this information. We also have experience in using geolocation and notification services, so incorporating these into our applications should be relatively simple. We will need to be wary of certain permission settings on mobile devices, as often permission is needed for applications to access notifications and geolocations, but once again, we have developed a mobile application with these functions in the past.

Deliverables (see appendix figures for mockup and other useful figures)

The application should function as a sort of data visualization tool, with direct access to the 511 API giving the user up-to-date information about local roadways. The user can then sort their information into their favourites so that they have the most personalized experience possible.

The main feature of the app is to allow users to search for specific roads based on ID, location or name. The user can directly input these himself and information will be returned based on that input. On the output page, the user will be able to see information about the roadway in question, including the roadway name, a picture based on the roadway at that specific time, as well as a special spot for conditions pertaining to that roadway at that time, like the ID of that camera, the name of the city it resides in, visibility and road coverage. On this page, the user can also save this roadway to his favourites, which will be stored inside Google Firebase. Once a roadway is stored, the user can access these roads from the main menu. Clicking the favourites tab will redirect the user to their own specific list of favourites roads, and clicking on a road from this page will also link them to the road's information. Finally, the app will also alert users of certain extreme weather conditions or backed-up road conditions, which can be accessed from the main menu. This page will also list all of the alerts, similar to the favourite roads page.

In order to effectively incorporate as many functions into this application as possible, we will be dividing the production into phases of development:

Phase 1

In phase 1, we will make sure the basic portions of the application work correctly. This includes successfully using the 511 API so that we may obtain the road conditions successfully. This means also successfully designing our views where the user interacts with the roads and views the specific information. This also means successfully utilizing the geolocation feature of the mobile device to get nearby roads relative to the user's location. This would also entail sorting out the permissions of the mobile device.

Phase 2

During phase 2, we will begin to add some of the more advanced features of the application. This includes the Google Firebase portion, where we will decide how we will save and obtain references from the database. This will allow us to begin implementing the favourites tab, where the users can save their personal favourites.

Phase 3

During the final phase, we will apply some finishing touches to the applications, like improving the overall look of the application thorough various CSS and Bootstrap techniques. We will also implement the alerts in this phase, which should work based on how the favourite roads feature worked. By the end of development, we will hope to have created a very useful application that we could see many people utilizing in the future.

CONCLUSION

All in all, we hope that our application is one that we can see not only ourselves, but many others using. The application should be simple in design; that is one of the main ideas we are trying to achieve. With this in mind, the app should also deliver information that the user will find useful to him, cutting out all of the unnecessary information doesn't need to know. Hopefully, the simple design but helpful UI will encourage users to use this application that will enhance the commuting experience for years to come.

REFERENCES / RESOURCES

Ontario 511. (n.d.). Retrieved from <https://511on.ca/> [1]

Google Firebase (n.d.). Retrieved from <https://firebase.google.com/> [2]

Cordova Plugins. (n.d.). Retrieved from <https://cordova.apache.org/plugins/> [3]

Overview | Firebase. (n.d.). Retrieved from <https://firebase.google.com/docs/reference/js/> [4]

APPENDIX

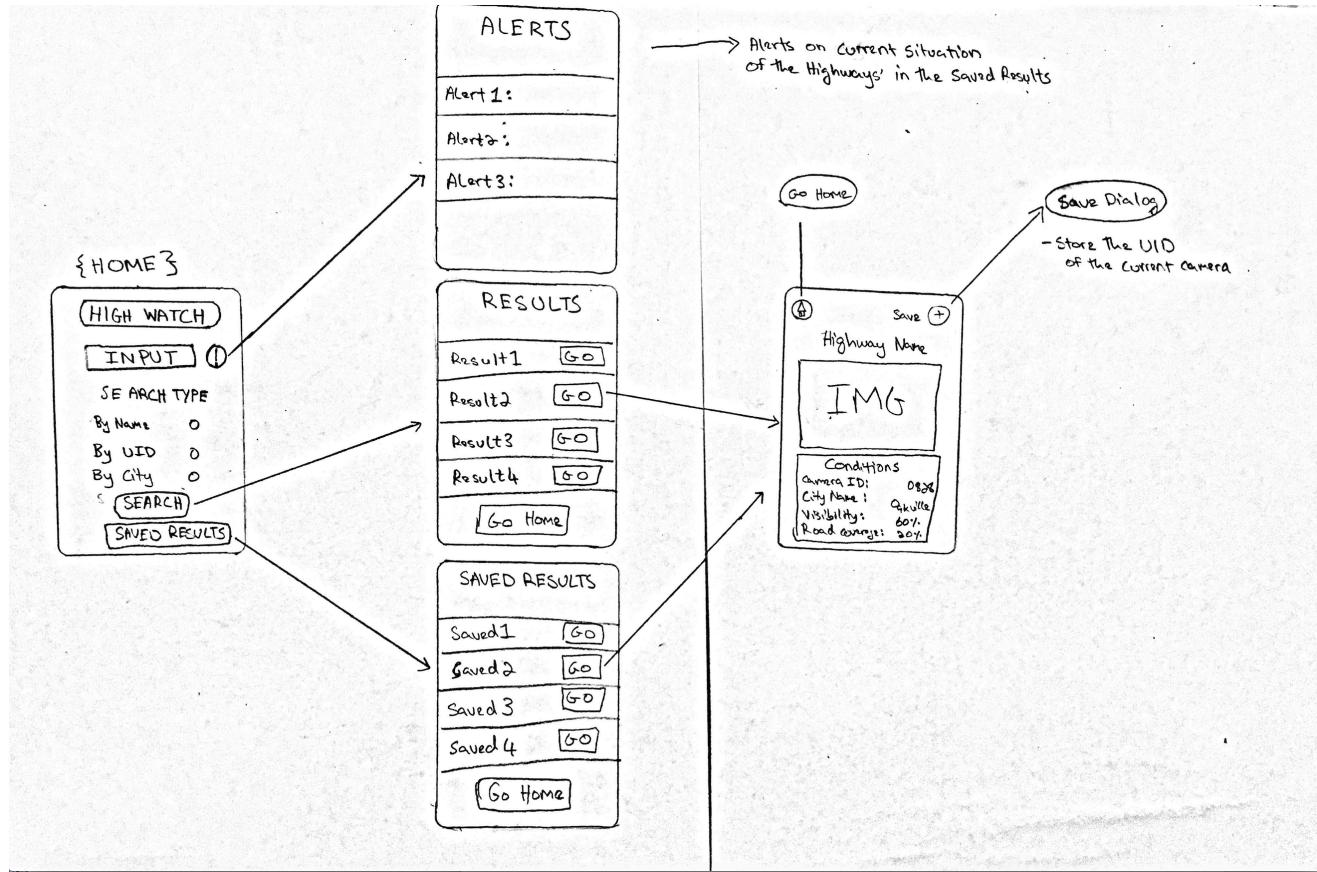


Figure 1.1: Mockup of the Application

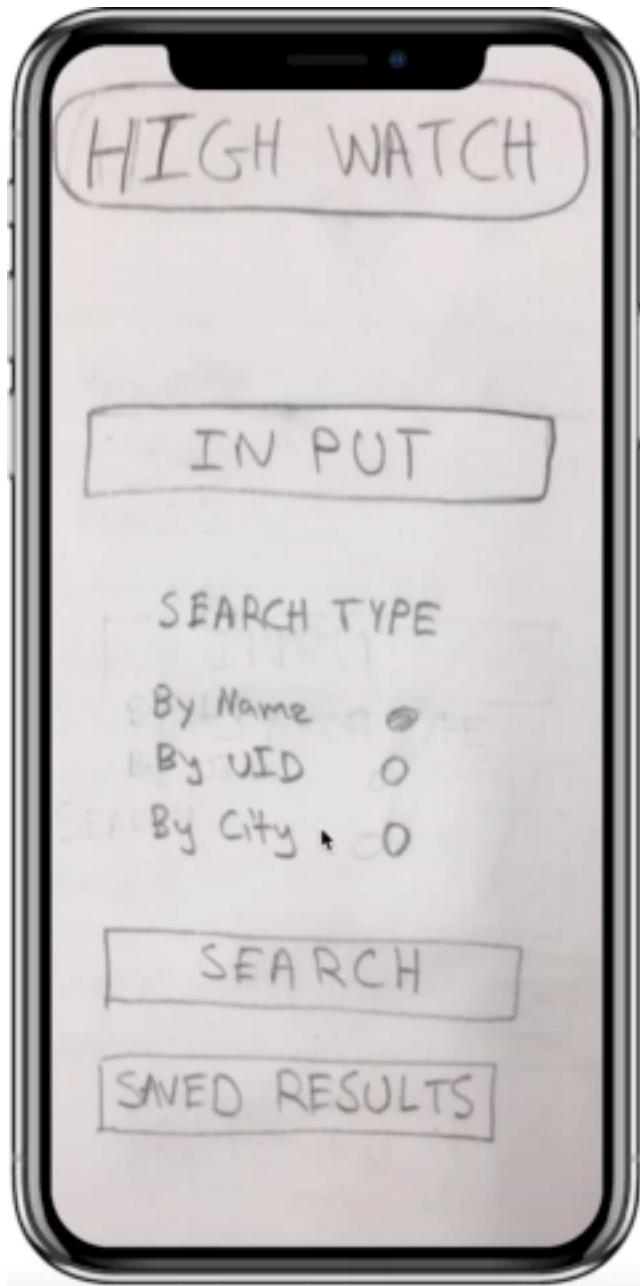


Figure 1.2: Demonstration of Limited App Capabilities (See Enclosed Video File for playable version)
A interactive version is also available at the following address: <https://marvelapp.com/43bch6c>