



Improving the process of collecting and storing data for evaluating stream health

Ethan Despres

Mentored by Christopher Burley



Introduction

To teach children more about the environment and how wildlife and vegetation are affected by human development, the teachers at Harford Glen have students record a variety of data about several of the local streams. All of the student's observations and data are recorded by hand, and long term records consist of keeping the paper records from past groups. To improve the process of collecting and storing the information, the goal was established to create an iOS application to collect, analyze, and then store this data in a more shareable format. A version of the stream evaluator application was created last year, which demonstrated basic functionality. The major goal for this revision was to implement better data storage and export functions, as well as user interface improvements.

Materials and Methods

The new standard programming language for iOS applications is Swift, which is built on the older Objective-C language. The first version of the application did not include storage for multiple sets of data, so a free database tool called Realm (realm, 2017) was incorporated into the new version. Realm makes managing app data easier, by allowing data to be constantly updated as well as by simplifying the code required to store any input from the user.

Because the app will be used by elementary school students it is important to make sure that the application has a friendly user interface. To accomplish this, the new version provides more pictures and instructions. A new feature is an interactive dichotomous key, that will help students to identify the various organisms that they find in the streams.

One of the main purposes of the app is to calculate a numerical value for the health of the stream. Version 1 did that, but it was very inconsistent in when it would calculate the final value. Version 2 now constantly updates the values every time a new piece of data is added. Once all of the data is recorded, it can then be exported, so that the teachers may store all of the group's data to create graphs and charts of the stream's health.

Results

Figure 1: The student datasheet.

Figure 2: First page of the flow chart of the dichotomous key.

Figure 3: First page of the flow chart for the stream habitat evaluation.

Figure 4: Calculation page for the "Stream Corridor Habitat Rating" and the "Biological Water Quality Rating".

Results (cont.)

Figure 1 shows the first scene of the app is the student/stream datasheet where the student enters general information about the location and weather conditions. This will be used in the future to create graphs and charts by correlating the collected results, using the data from the same stream, time of year, what school collected it, etc. Figure 2 shows the first page of the organism flow chart section where the students use the interactive keys to determine what organism they found. The organisms' information is stored in a table where the students can see each individual organism they choose as well as delete any of them if needed. After that at Figure 3, the students go through a similar flow chart for the stream habit rating, which uses reference pictures to help the students quantify abiotic aspects of the habitat. The final evaluation is shown in Figure 4, where both a numerical value and explanatory text are provided. Finally, the results can be shared via email by exporting the data stored in the Realm database. This generates a CSV file, which is a common standard for data exchange to create graphs and charts.

Conclusion

The purpose of this project was to improve the process an iOS application uses to store and share data collected by students, allowing the application to export the stored data. This app will allow for the teachers at Harford Glen to better record the information they gather in the field with their classes and not depend on only having a hand written copy of everything. While all of the initial goals have been met, the application can still be improved in its functionality and appearance. There are some bugs present, the code can be simplified, and the user-interface of the app can be altered to allow for a more friendly experience with the users.

References

realm (2017) *Realm Swift 2.6.2*. Retrieved from <https://realm.io/docs/swift/latest/>