Demand Response Portal
Future Work and Limitations Report

Updated on November 27, 2023

Under the direction of

Eric L. Brown

Table of Contents

1. Introduction

1.1 Purpose of the Report

The purpose of this report is two-fold. Firstly, it will help anyone working on this project in the future better understand some of the unmet goals of the project. Secondly, it will explain some of the limitations that kept us from being able to accomplish those goals.

1.2 Scope

In this report, we will look at the next few steps that could be considered for future work. There are many paths a project like this one can go down, and while it is possible that a future team would want to completely scrap what we have done and start over, for this report we will be assuming that they will be building upon our work so far. We also will not be looking far ahead so that the goals will be able to remain flexible and the team will be able to adapt to new requests from the client.

2. Future Work

2.1 Potential Enhancements

Over these two semesters we have added many features to the portal that we are very proud of, however there will always be more features that could be added.

We discussed early on adding support for multiple data models instead of the single ARIMA model that we have currently.

Professor Burchfield, an advisor for the project, has recommended switching from a python-based server to a JavaScript one like NodeJS if the project were to switch to a more scalable approach, however this would require a lot of work.

We recently discovered that the website does not score well on performance tests that we can run, improving that score would be a great step in improving the project.

We had originally planned to include a light/dark mode toggle to change how the page is displayed.

Currently the VTN that runs the simulation that creates our meter data exports these values through a spreadsheet that is then plugged into the database through the db-importer container. Changing this so that the VTN uploads directly to the database would allow the website to act as a live feed of the VTN.

Currently the server code does not log API call interactions. It would be preferable to have the IP addresses of users interacting with the system logged along with what the interaction is.

Though not originally planned, it would be extremely helpful to host the website somewhere instead of relying on localhost to display the content.

2.2 Research and Development

The biggest part of research and development will always be staying in touch with the client and adapting to new requests as the project goes on. However, another key part is staying on top of the changing libraries and frameworks that we are currently using in the project. We are using several frameworks like Bootstrap, Bokeh, Pandas, and others in this project to get the portal working as smoothly as possible. The next team to work on this project will have to work to balance keeping these frameworks and libraries up to date while also keeping the portal functional and lightweight.

In addition to updating the current frameworks and libraries the project is using, the next team should be on the lookout for new ones to take their place as industry and client standards change. There may be more in-depth graphing frameworks than Bokeh, or more useful frontend frameworks for Bootstrap, the next team to work will have to decide early if they would like to keep the current frameworks and libraries or pivot entirely to new ones.

2.3 Timeline for Future Work

Naturally the first thing a new team working on this project will want to handle is the purpose of their work. Our team has laid the groundwork for the next team, they will want to clearly define what kinds of features they plan to add to the portal and think on how these features will work.

Once a goal is established, they should look and see if there are any parts of the project that they want replaced entirely. As mentioned in the Research and Development section, it is possible that there will be newer frameworks and libraries that the client, the professional world, or the new team themselves prefer to work with over what we have now. These changes should be made as early as possible in development because changing or removing a single framework can make other work done useless. Because of this the newer team should only change things if they are confident in their approach to the newer code.

Finally, the team needs to create and maintain features. Creating and assigning user stories will really help distribute the work of these new features. Once the team reaches this step it is where they will remain for the rest of their development cycle unless something needs to be changed later than preferred. Maintaining the system and its features includes running security checks, keeping track of project space, rewriting older code to be more useful and readable, and any other actions that would keep the website running smooth for longer.

Using this simple order of operations should help the next development team to maximize their time when working on this project.

3. Limitations

3.1 Technical Limitations

With any project there are bound to be some limitations and ours is no exception, however we did take care of many of our problems to the point where there are few remaining.

Firstly, we were unable to get the Bokeh graph to update instantly when you press a button. This is because it pulls its data directly from the server and we were focused on getting it running instead of making it perfect. Instead of pulling data every time the user clicks a button, it simply pulls new data and updates itself every 10 seconds. We were able to improve this problem slightly by using another container to cache data that the Bokeh graph is pulling, but this only helps with the problem instead of outright solving it.

Next, we were unable to improve our performance score on several webpage testing programs. This is likely due to the Bokeh graph embedded in the page, however it cannot be helped, since the graph is the core of the webpage and Dr. Rogers preferred us using a python-based data visualization tool.

Finally, we were unable to get a logging function working for the server. This is because the layer the function would need to be is not able to view the IP address of users contacting the server. We are technically able to implement logging now, but without being able to track the user's IP address it would be useless.

3.2 Resource Limitations

Thankfully, we did not have many resource limitations in this project. The only real resource that we would have preferred to access too would have been a way to host the website and database so we would not be testing on localhost. Other than this one problem we really did not have any other limitations here.

3.3 Data Limitations

There was only one limitation that was caused by data-related issues.

Firstly, we were only given one set of data over a year. This was only a limiting factor because it meant that there was only so much training we could do on the ARIMA model. The data model only gets more accurate the more data that it can see and learn from. This means that the prediction data will be slightly off and that anyone using it to reward customers will have to reward them more often for now. This problem could be fixed simply by getting more data from the VTN and plugging it into the current model.

3.4 Regulatory and Compliance Constraints

We did not face many compliance issues with the project. Although we have always made sure we are using anonymized data. We hope to minimize any risk to data by doing so, although in full production this would not really be necessary.

4. Conclusion

4.1 Summary of Future Work

Ideally, the future work done on this project will keep our current work intact and build upon it. However, if there is a large change in scope or direction for the project, it is highly likely that a lot of the work done so far will have to be redone. We built this project first and foremost to stick closely to the scope that we were given, and because of this most of the possible improvements for those working on this project in the future is to try to solve the limitations we ran into as the project went on, some of which might need the project infrastructure to be changed to be resolved.

4.2 Implications of Limitations

Most limitations were only apparent late into the development cycle. However, with the current scope, the project should be successful even with these limitations. If the scope were to change it could result in large rewrites and refactoring needing to be done to keep the project on course. All of this is to say that the limitations so far have not been major obstacles. However, if the scope were to expand rapidly, they could become major spikes in development fast, which is why we recommend the future team establish the scope as soon as possible to minimize time taken by these spikes.