

Reflection Report on SmartServe

Team 21, StoneCap Solutions

Max Turek *turekm*

Ryan Were *werer*

Sam Nusselder *nusselds*

Peter Minbashian *minbashp*

David Bednar *bednad1*

1 Changes in Response to Feedback

[Summarize the changes made over the course of the project in response to feedback from TAs, the instructor, teammates, other teams, the project supervisor (if present), and from user testers. —TPLT]

[For those teams with an external supervisor, please highlight how the feedback from the supervisor shaped your project. In particular, you should highlight the supervisor's response to your Rev 0 demonstration to them. —TPLT]

1.1 SRS and Hazard Analysis

For the SRS, our instructor gave us feedback that the boundaries of the system were not quite clear from the context diagram. They wanted us to include a list of tables and a list of diagrams. We were also advised to include the value range of monitored variables. Lastly, they suggested we provide more rationales for the requirements. Using this feedback, we made appropriate changes to the SRS. The context diagram was adjusted to further separate Smart Serve from the external environment that it is used in, and how it interacts with it. Adding the value range for our monitored variables helped us define what we wanted these variables to do and why we are monitoring them. It became clear that some variables we initially thought that we would monitor were not really important, and others that we had overlooked were necessary and thus added. For example, we had many variables relating to the type of drink, and its volume. We instead switched this to one variable that just handles the volume as its type is not relevant. Lastly, providing the additional rationale to requirements helped to clarify their purpose which benefited us as a group so we were clear on what we wanted/needed to do, and to others external to the team so they could better understand what we were aiming to accomplish. The last change we made was looking back on all the requirements and as a group discussing if we met them. If we didn't we discussed why there was a gap there and if it should be closed. For some requirements, we thought they

were important to still have so we implemented the functionality. For others, we realized they no longer made sense for the direction our project had taken. For example, we removed a requirement that had a notification be sent to the operator after every drink was made.

For the Hazard Analysis document, the only feedback provided to us was from the other teams. After reviewing each issue they raised, we edited our document to better clarify any issues they saw so the document made more sense. In our last edit, we removed some security requirements and updated some failures in the FMEA table as raised by the other teams.

1.2 Design and Design Documentation

1.3 VnV Plan and Report

The VnV Plan was updated to address issues and feedback from instructors, Tas, and test users. Some of these issues from the instructor and TA included updating and separating the validation plan from the verification plan. This ensured that the reader could be confident that the right system was built. The traceability matrix was also updated to include more of the functional and non-functional requirements as well as discuss any requirements that weren't included in the test cases. Some feedback from test users was to include more test cases for when the drink may only be half full as opposed to completely full. This was included by adding test ST-FR-DMT-04. The VnV Plan was also updated to ensure it aligned with preceding documents such as the SRS document by updating the corresponding requirement numbers as some requirements were removed from the SRS document.

The VnV Report was updated to address issues and feedback from instructors, Tas, and test users. There was no relevant feedback from the instructor or Tas. Some feedback from test users that was incorporated was explaining the scoring system used in certain non-functional requirement test cases in more detail. Test cases where quantitative data was mentioned as an output was discussed and included as a potential output to not confuse the reader. The SRS document was also included as relevant documentation. The VnV Report was updated to ensure it aligned with preceding documents such as the SRS document and VnV Plan by updating the corresponding requirement numbers in the SRS document and test cases in the VnV Plan.

2 Design Iteration (LO11)

Our method of design iteration was very hands on and often ended up being by trial and failure. Certain features like setting up a domain name to access the web app was initially something we intended to do however, after experiencing a variety of network issues with the school WiFi it was discovered that it would not be feasible in the given time frame and resources. Our process involved a

slow addition of a few functionalities then through verification sometimes it was determined that it would not be possible and a work around would need to be created. It was almost like a two steps forward one step backward approach in which there would a succession of successes then there would be a fatal flaw that would set us back. Basically an iterative process that involved a lot of failures that eventually allowed us to succeed upon perseverance or creative solutions. Our sensor to detect a cup stopped working, and it took a few days of troubleshooting to finally isolate the problem from a hardware issue to actually an overload of CPU resources on the raspberry pi. From this process our design finally evolved from its first version of simply just pouring out water from one pump, to a fully autonomous five pump cocktail creation system.

3 Design Decisions (LO12)

In our initial design process, we based our decisions on three separate groups: limitations, assumptions, and constraints. These decisions not only impacted our initial design but also the ones we would alter or add as we progressed.

Regarding limitations, we faced two main challenges - budget and technology. Our limited budget forced us to change our original decision of hosting the website externally and instead host it on the raspberry pi to save money. Additionally, we had to simplify our designs and feature implementations due to the technological limitations of our equipment. For example, we had to drop the drink recommendation feature and use web sockets instead of timed API calls for real-time updates due to the limitations of the raspberry pi's processing power.

We also had assumptions about our abilities and the technology we were using, which impacted our initial design decisions. Initially, we underestimated our abilities and made simple design decisions that we later altered to add more complex and interesting features. For example, we initially thought it would be challenging to implement a queue page, but as we became more familiar with the technology, we found it was not challenging to implement and would make our project better.

Lastly, our decision-making constraints were mainly around time. We had to factor in the time constraints of the course and other semester work, resulting in some complex features being eliminated or simplified to ensure timely delivery. For instance, we had to reduce the analytics feature to display surface-level analysis instead of recommending ingredients based on user demographics, given the time constraint. In summary, we had to be mindful of limitations, assumptions, and constraints throughout the design process to make the best decisions for our project.

4 Economic Considerations (LO23)

When developing a product it is important to consider the economic factors that can determine its success and profitability. In the case of our unique drink-making machine, we recognize that the demand for the product is a crucial aspect that needs to be explored further. This would involve conducting market research, reaching out to different bars, restaurants, and private owners, as well as examining competitors to determine if there is a viable market for the product.

Production costs are another key consideration as the machine requires expensive hardware components such as a raspberry pi and peristaltic pumps along with other mechanical and electrical components. The cost of manufacturing and assembling the product including labor costs would also need to be taken into account. To determine the scale of the operation it would be helpful to have an understanding of the demand beforehand.

Based on the uniqueness of our product, we estimate that it could be priced anywhere from 500 to 1000 dollars. This would depend on the inclusion of upgraded components that could not be added for the first revision of the product.

In terms of software, the code would need to be sold alongside the smart-serve machine as either a subscription to the web application or rights to host the application. We believe it is important to keep the code closed-source to prevent any tampering with the product which poses obvious safety risks, especially given the potential for alcohol-related hazards. The product could also never be open source due to the expensive hardware needed to run it.

In conclusion, economic considerations play a critical role in the development of any product. While we believe our drink-making machine has the potential to be successful among bars, restaurants, and hobbyists, there are many factors that need to be considered to ensure its profitability and viability in the market. By continuing to explore the potential for the product and addressing any challenges that arise, we are optimistic about its possible future prospects.

5 Reflection on Project Management (LO24)

5.1 How Does Your Project Management Compare to Your Development Plan

Overall our team performed well as a group with everyone being committed and staying true to deadlines. For the first few months we had regular team meetings weekly with a proper agenda and delegated tasks, however as the semester progressed meetings became less important and people were expected to finish their tasks independently. Everyone stayed true to their roles with the software

team developing the web app and the hardware team responsible for the python code and physical Smart Serve machine. In regards to our workflow plan, for the most part separate branches were used to work on but the majority of the work was done on the Web-App branch and then merged into main occasionally. All the technology we set to use we did, expect for testing most of the testing was done through testing use cases and not technologies

5.2 What Went Well?

We were able to execute what we wanted to deliver as an end product, and that is something we can be proud of. We made sure that we didn't stretch too far in terms of goals and requirements of the product we wanted to deliver. Furthermore, during pain points in our testing or development stages everyone rose together to work and solve the issues. There was a sensor issue causing our machine to spin our turntable infinitely, and needed to be resolved before our final presentation. We spent three consecutive days trying to resolve the issue and wouldn't quit until it was solved. Fortunately, this type of perseverance helped our team deliver a phenomenal end product.

5.3 What Went Wrong?

I believe what went wrong was perhaps not having someone take the lead on the project management. It would have been very helpful to have a formal structure in regards to the path of our project because at some times there was some ambiguity in terms of deliverable and missed communication between team members. Having a team member being able to constantly raise issues on our project board and create tasks and assign them, would have streamlined our development and created a much more efficient team.

5.4 What Would you Do Differently Next Time?

As mentioned above, having a project manager for our group would have been ideal. Furthermore, I think doing more research and planning before committing to a project would have been a much better way to determine what and how to implement the certain features. For example we initially figured it would be easy for the end user to connect to the web app through a domain name, however since the machine would be presented in the school on their network, it would be tough and too much work to implement when it is locally hosted on our raspberry pi. The school WiFi presented many issues we didn't anticipate something that could have been resolve through better planning.