

Iteration #2: Work Completed

Iteration #2 Project Leader: Zack Matey

Jordan and Dan Team – Work Completed:

Jordan and Dan worked on the GUI this iteration. Using DTOs set up earlier in the development process, the GUI sends either a new visit or recurring visit DTO to the controller, which parses the fields and decides which are correct, and sends back a DTO with information about incorrect fields. The GUI was set up in a rough state for this iteration. More extensive development and some refactoring will be needed in future iterations. For now, though, the GUI can accept data and send it off to the controller and can display error messages received from the controller. Some items haven't yet been integrated but considering the timeframe of the iteration much of the GUI logic has been completed.

Riley and Hamza Team – Work Completed:

Riley and Hamza worked on the Controller this iteration. The controller receives input from the GUI and sends back a DTO with incorrect fields based on the system's parse. Riley and Hamza implemented a parsing system which allows this to be done elegantly using a validation library. The controller will probably require less refactoring in the future but given that the GUI was a more time consuming and more code-intensive development process, this is to be expected.

Significant Accomplishments:

This iteration, we were able to complete a number of significant accomplishments. We've implemented controller and GUI interaction, using DTOs, which allows us to set up a robust and expandable GUI and controller mostly independently. This is in line with the MVC pattern and enables our codebase to be modifiable and expandable. I identified some code which may need to be refactored above.

I wouldn't think that any significant issues remain, however, there are some minor issues with the GUI and controller which need to be resolved in Iteration #3. As I have written in other places, I consider this to not be a problem – we were able to complete a significant amount of development this iteration, and the fact that some things aren't perfect isn't a big deal. In keeping in line with the agile process, we can simply address these issues in the next iteration.

Revised Risk Assessment:

We were able to eliminate a significant risk this iteration, that being the feasibility of using a Java Swing GUI to receive user input. Our current GUI is somewhat rough, but it is functional. We were able to implement a GUI using this system. We were also able to maintain class and object separation while doing this, which allows for an expandable codebase in line with the MVC model.

Looking forward, potential future risks mostly consist of ill-defined "changes" present in Iteration 5. We're eventually looking at a web-based deployment, which I, at least personally, do not have much experience in. However, due to the nature of this class, I anticipate that this material will be covered. Given that, I would say that we have eliminated our largest outstanding risk, and that, looking forward, our future iterations consist of continuing to iterate on this codebase, without as much major refactorization as we saw at the beginning of this iteration.