

What went well

- Using Django for form creation and templating was very straightforward; the documentation was clear and well documented.
- Existing Django libraries drastically simplified temporary ID generation and database obfuscation.
- Gaining an idea of how each team member works and what each member's strong suit is.
- Ensured the database was properly configured and verified that data could be stored locally without errors.

What didn't go well

- Scrum meetings were often longer than expected due to lack of communication between team members outside of scheduled meeting times.
- The initial learning curve and troubleshooting for configuring Google Secret Manager took more time than originally estimated and was difficult to test in a secure way without the ability to deploy the application yet.
- Learning how to use applications (Docker, Django, etc) that have never been explored before.
- Some development time was spent resolving environment configuration issues (Docker rebuild and setup inconsistencies).

What could be improved

- Earlier commitment and prioritization when handling scrum tasks before meetings.
- Breaking down complex tasks into more manageable subtasks would help improve time estimation and communication
- More research time is placed into learning the basic concepts of new applications.
- Continuing to refine how tasks are broken down in Jira will help improve clarity and overall team efficiency.

Challenges

- Some difficulty communicating project scope and initial project requirements with project sponsors and IT department due to lack of technical understanding from sponsors.
- Staying consistent with both intake forms to exactly how the sponsor wanted was somewhat difficult as the fields on the manual form were somewhat confusing, even with help from the sponsor. (A bonus now is that the clients should now have an easier time navigating the new web forms)
- Gaining a true understanding of the project scope. Sponsor had one idea while the team had another idea which caused confusion in the beginning.
- Navigating evolving project requirements while simultaneously building backend functionality required ongoing clarification and adjustment.