Raymond

Problems & Solutions

1. Lack of understanding when it came to working with the technology
    a. Do one hour of learning and research a day for the technology that we're using for the webapp

What went well?

1. Project is moving forward.

Brent

Problems & Solutions

1. **LLD Designs took longer than expected.**
   Due to inexperience with the concepts involved in my LLD, it took longer to develop these designs. This caused me to make multiple revisions to my designs, wasting time I could have spent doing other work items.
   Solution: Spend at least 2-3 more hours researching and consulting with my teammates to try and gain insight into how to develop meaningful LLDs.

What went well?

1. There was good team communication this sprint, especially when coordinating times to meet in person to work on deliverables.
2. I was able to receive client feedback for my LLDs, which helped to clear up confusion regarding my designs.

Gideon

Problems & Solutions

1. **Lack of in depth LLD Design**
   Solution: We do more research(at least 5 hours) for our LLD Design for the work items in order to get a better idea for how long it'll take us to accomplish these work items after being granularized.

What went well?

1. **Collaboration**
    - There was good, clear, and concise communication with all members of the teams as to when we were going to work, and more importantly what we were going to tackle that day.

Vivian

Problems & Solutions

1. Since I was not the original creator of the feature, I was slightly unsure exactly what the original creator of the feature had in mind. For example, what would everyone who is collaborating on the itinerary have the itinerary items on their feature?
    a. Spend 1 hour talking to the original creator of the feature before deciding on major features.

What went well?

1. It was easier to find time to work since we did not have to collaborate with others.

Joshua

Problems & Solutions

1. **Inexperience with creating LINQ on LLDs**
   Creating method implementation on LLDs ensures that it's easy to replicate if someones read it.
   Solution: Spend at least an hour to research LINQ and EF Core translations so that implementation of the LLDs transitions smoothly.

What went well?

1. Client input in order to create higher quality LLDs
2. Creating separate LLDs for error cases

Long

Problems & Solutions

1. **Lack of experience in web application deployment and server set up.**
   In the process of setting up our production environment, we encounters numerous errors such as permission and time out due to our lack of experience in deploying a web application which lead to a lengthy process of debugging.

   Solution: In order to improve on this, we decided that team members should do their own research on this topic in order to be more prepared when it comes to server configuration and set up to reduce time resources.

What went well?

1. **Setting up AWS Microsoft SQL Server.**
   We were able to back up the synchronous database server on the cloud to be in sync with our local database including, tables, stored procedures, functions and data.

Conclusion

There were a few things to take away from this sprint. While the nature of the work that we need to complete is mostly individually driven, we still need to consult with one another for a variety of reasons, some people in the team have to integrate features which they did not initially draft, so this leads to a loss in translation when looking at the BRD, to minimize confusion there will be a continuous feedback loop of the team members asking one another if their interpretation of how a feature on the BRD is describe is captured by the LLD and implementation. Additionally, the transition of our database from a local file system to a cloud production environment went quite smoothly. All of our tables, stored procedures, functions, and other database aspects were successfully transferred to AWS RDS. All members of the team were able to connect the remote database through SQL Server Management Studio and manipulate the data in our tables.