**Project Journal**

**Application Structure**

Team followed service oriented architecture as result of which application has been decomposed into pages namely

* Login
* (Admin only) - Add transactions - such as manual refunds on fees
* (Admin only) Adding new accounts - Checking, Savings
* (Admin only) Closing existing accounts
* Transfer (internal) between your own accounts - one time or recurring
* Set up recurring or one-time Bill payment for external payees
* View and search Transactions - for credits/debits/checks/fees - up to last 18 months
* (Admin only) - Add transactions - such as manual refunds on fees

**Sprintly Development**

**Sprint 1 (March 07 - March 13)**

**MoM (notes)**

* Researched the various possible approaches to develop the application
* Decided on the Application Domain
* Discussed the depth of scope

**Result**

* Decided to develop Bank Application using Django as backend and React as Frontend
* Decided on Postgres as suitable database for this application

**Sprint 2 (Mar 22 - Mar 28)**

**MoM (notes)**

* Discussed about the requirements and resources
* Discussed on frontend page views

**Result**

* Everyone was assigned an API to work on.
* Basic templates that can be used for frontend were decided

**Sprint 3 (Apr 08 - Apr 14)**

**MoM (notes)**

* Discussed issues with Integration of Frontend and Backend
* Discussed about the project architecture
* Discussed issues with Backend API functionality deployment

**Result**

* Decided about the architecture
* Team started to work on setting up APIs

**Sprint 4 (Apr 20 - Apr 26)**

**MoM (notes)**

* Discussed the appropriate deployment methods for APIs
* Weighted the pros and cons of various potential means

**Result**

* Finished writing the necessary APIs
* Finished integration of Backend and Frontend
* Tested the application end to end

**XP Core values**

Sudha VijayKumar

Courage

I tried encouraging everyone to participate in every discussion and made sure that everyone is heard so that every team member is equally valued and no one is left out.Showed each team member respect and integrity based on the opinions that they had for the expectations of our project

Asad Abbasi- Simplicity

I tried keeping the requirements simple, short and clear. I tried to keep the implementation simple in the beginning to have a basic working model so that the final application can be built on top of it.

Nithil Sethupathi - Respect

I was quick to set up meetings when any of the team member’s had integration issues. Also respected others opinions when building the application and tried to reach consensus among the team.

Venkata Lakshmi Pravallika Ramachandruni - Communication

I constantly encouraged my team to communicate with each other about their ideas and issues they faced. Updated the team on all the deliverables and project documentation.

**Burndown Chart**

