**PROJECT JOURNAL**

**Application Structure**

We have followed service-oriented architecture as a result of which the application has been decomposed into five microservices namely,

* Login
* Adding new accounts - Checking, Savings
* Closing existing accounts
* Transfer between 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

**Weekly Development**

**Week 1 (Nov 07 - Nov 13)**

**MoM (notes)**

* Researched the various possible applications that we can develop
* Decided on the Application Domain
* Discussed the depth of scope that we want to cover

**Result**

* Decided to develop Bank Application with the above components
* Decided to use Microservices development technique

**Week 2 (Nov 14 - Nov 20)**

**MoM (notes)**

* Decided the services to decompose
* Discussed about the requirements and resources needed
* Constructed the necessary folder structure
* Discussed about the relevant Database options

**Result**

* Everyone was assigned a microservice to work on.
* Go and Nodejs were decided to be the languages for writing APIs
* Decided on suitable databases for each microservice

**Week 3 (Nov 21 - Nov 27)**

**MoM (notes)**

* Creation and configuration of databases
* Started writing APIs
* Discussed about the architecture

**Result**

* Decided about the architecture

**Kowshhal**

* Started creating API for login and Set up recurring or one-time Bill payment for external payees

**Yash**

* Started creating API for add account and closing existing accounts

**Ramya**

* Started creating API for transfer between accounts - one time or recurring

and Set up recurring or one-time Bill payment for external payees

**Nivali**

* Started creating API for View and search Transactions – for credits/debits/checks/fees

**Week 4 (Nov 28 - Dec 4)**

**MoM (notes)**

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

**Result**

* Finished writing the necessary APIs
* Integration of microservices using KONG API Gateway
* Tested the application end to end

**Kowshhal**

* Created API for login and Set up recurring or one-time Bill payment for external payees

**Yash**

* Created API for add account and closing existing accounts

**Ramya**

* Created API for transfer between accounts - one time or recurring and Set up recurring or one-time Bill payment for external payees

**Nivali**

* Created API for View and search Transactions – for credits/debits/checks/fees