

TEAM 4 CONTRIBUTION REPORT

1. Ayisha Fathima

- Roles and Responsibilities Played
 - Role: Developer in Scrum Team
 - Responsibilities:
 - Front End Development for C#
 - Back End Development for C#
 - Android Development
 - Analysis & Design
 - Documentation
 - Some project management (in terms of coordinating team to produce deliverables before the specified deadline – mainly for documentation)
- Deliverables and Interaction within the team
 - a. Front-End Development
 - Department Side UI (for Department Head, Department Representative (including Manage Collection Point) and Department Employee) - using Razor HTML Pages, jQuery and CSS
 - b. Back-End Development (Logic)
 - Create Requisition, Update Existing Requisition (Consolidation of one single requisition form for entire dept at any one time), View Requisition Details for all Dept Employees (including reps and heads)
 - Restricting requisitions seen in list of requisitions by Dept Head according to approval and submission status
 - Submission of requisition (for dept rep)
 - Manage Collection Point – Updating of Department's collection point when changed by Dept Representative
 - c. Android Development
 - UI view for department head requisition list and requisition details (which includes approve/ reject function with optional input text for specifying reason) and respective activities and recycler views associated with that.
 - Receive JSON data for Requisition List and Requisition Details
 - Getting data that was input in the Requisition Details form and sending that data by JSON to backend department controller to process approval status of requisition and reason for rejection (if any).
 - d. Documentation
 - Description of SDLC Activities in project

- User Manual for Department Head for Android and Department Employees for Web.
 - Screen prototypes for use cases
 - Sequence Diagram
 - Unit Testing for Use Case (Raise Requisition Form - A.K.A raising new requisition form)
 - e. Interaction with Team
 - As scope for Department functions were lesser, efforts such as pair programming were not necessary when I was assigned to program department side functions. However, engaged in discussions with various members of team whenever bugs were encountered or when unsure of how to build a specific function.
 - Coordinated the team for documentation deliverables – broke down deliverables and assigned each member parts to complete daily so that we could finish documentation punctually and at a reasonable pace.
- Major Challenges and Resolution
 - Was initially weak in technical capabilities, especially in C#.
 - Invested time in the first week and two days of second week to revise .NET Core, Entity Framework, etc... and gain practice from online tutorials.
 - Was unfamiliar with GitHub but worked together with team to figure out how to use GitHub and learnt to merge latest code properly (making use of fetch, pull, conflict resolution etc...)
 - Understanding why and how to implement technically complex things such as how to connect Visual Studio with Android with the aid of iisexpress proxy, how to implement HTTP API GET and POST. The internet and resourceful team mates who knew how to find the right articles on the internet helped resolve this.
- Quantum
 - 4 weeks – Approx. 145+ hrs (including time spent on weekends)
- Experience
 - Initially had very low confidence as programmer but division of work was structured such that everyone had to program a significant portion and this practice helped me reinforce what I had learnt in class and helped me gain more confidence to program.
 - Enums proved to be difficult and inconvenient to work on multiple occasion but learnt more about them and the various methods we can manipulate them with.
 - Gained more experience on how to debug a web app properly in the event of issues and what aspects to look out for to find source of error.

2. Kyaw Thiha

- Roles and Responsibilities Played
 - a. Role
 - i. Developer in Scrum Team
 - b. Responsibilities
 - i. Front End Developing
 - ii. Back End Developing
 - iii. Android Developing
- Deliverables and Interaction within the team
 - a. Front End Developing
 - i. Data Table with Bootstrap JS and CSS
 - ii. Standardize the UI views
 - iii. Supplier List, Create, Details and Delete UI (with Lance)
 - iv. PO Create (with Hanh Nguyen)
 - v. Modal Confirmation for Delete Supplier
 - b. Back End Developing
 - i. Contribute when designing the ER diagram/ Models (with Team)
 - ii. Routing Security (with Saw and Lance)
 - iii. Supplier List, Create, Details and Delete (with Lance)
 - iv. PO Create (with Hanh Nguyen)
 - v. API for PO List, PO items List according to Supplier ID (with Hanh Nguyen)
 - vi. Saving the new PO create data from Android (with Hanh Nguyen)
 - c. Android Developing (with Hanh Nguyen, Saw and Lance)
 - i. UI view for PO List, PO items List and create by using Recycler View
 - ii. Receive JSON data and show for PO List
 - iii. Receive JSON data and show for PO Items List according to the Supplier ID
 - iv. Taking input data from Recycler View input box in PO create
 - v. Sending data by JSON to backend for PO newly created data
 - vi. Login/ Logout
 - vii. Routing
 - d. Documentation
 - i. Sequence diagram for Supplier and Department
 - ii. State diagram for PO (Purchase Order)
- Major Challenges and Resolution
 - a. Front End Developing
 - i. How to design the Web UI to be responsive and user friendly
 - ii. How to implement the Data Table and Modal
 - b. Back End Developing
 - i. Making sure the business workflow
 - c. Android Developing

- i. Getting the data JSON data and Posting
 - ii. Using the Recycler View and Taking data from the Recycler View Input box
- Quantum
 - a. Week 1-4 => 142 Hr
- Experience
 - a. Morning Scrum Meeting
 - b. Solving problems and fix bug
 - c. Security and validation
 - d. Time management
 - e. Finish project as a Team

3. Ling Teck Moh Benedict

Roles and Responsibilities Played

- Scrum Master / Member in the Scrum Team
- ASP.NET Core C# Developer
- Android Mobile Java Developer

Deliverables and Interaction within the team

- ASP.NET Core C# Development
 - Designed and coded the initial architectural backbone of the system (entities, controllers, views, database seeding)
 - Researched and designed API endpoints for both GET and POST requests
 - Collaborated with various SCRUM team members to develop, test and debug code (work distribution as documented in source code)
- Android Mobile Java Development
 - Researched and designed the utility classes GetRawData and PostJsonData which extends AsyncTask to handle GET and POST requests from API endpoints on background threads
 - Researched and implemented the RecyclerView and its adapter
 - Research and implemented Jackson library for JSON parsing
 - Researched and designed the GET and POST API endpoints in ASP.NET Core for the native android application to consume
 - Collaborated with various SCRUM team members to develop, test and debug code (work distribution as documented in source code)
- Interaction within the team
 - Organised and planned daily meetings for SCRUM team to provide progress updates, synchronise work, share and learn, remove obstructions to team progress
 - Researched on and explained various technical concepts such as APIs, callback methods, RecyclerViews and AsyncTask to the SCRUM team to empower them to achieve technical proficiency
 - Collaborated with individual SCRUM team members to debug code
 - Monitored project backlog to ensure that SCRUM team was on schedule
- Documentation
 - ERD Diagram
 - Class Diagram
 - Sequence Diagram
 - StateChart Diagram

Major Challenges and Resolution

- Implementing API endpoints in ASP.NET Core

- Spent weeknights and weekends researching extensively on forums such as StackOverflow and developer blogs
- Implementing logic in Android to consume JSON data from API endpoint
 - Spent weeknights and weekends researching extensively on forums such as StackOverflow and developer blogs
- Managing SCRUM team work distribution and progress
 - Having daily team meetings to gather feedback and progress reports
 - Pair programming

Quantum

- 4 weeks: approx. 163.5hrs

Experience

- ASP.NET Core and Native Android Development
- Git repository
- Agile Methodology
- SCRUM

4. Shermaine Lim Si Hui

Roles and Responsibilities Played:

- Developer in the Scrum Team
- Machine Learning, DevOps and Cloud, Web and Mobile Developer
- Pair Programming

Deliverables and Interaction within the team:

Learning New Technologies:

1. Azure Machine Learning to automate demand forecasting, train and compare models
2. Various regression models compared, and Decision Forest Regression had the best results
3. Creating Azure Machine Learning Web Services to integrate into ASP.NET with API
4. Create API and call API, test with POSTMAN
5. Configuring Cloud with ASP.NET and Android
6. DNS Configuration

DevOps and Cloud:

1. Publishing ASP.NET MVC to Cloud VPS
2. HTTPS Encryption with SSL, Performance Optimization on WOW64
3. Algorithmic Efficiency, Time and Space Complexity

Web Development:

1. Integrating Azure Machine Learning Web Services into ASP.NET for PO (Purchase Order) using Machine Learning Demand Forecasting
2. Store Clerk Bar Chart and Table for Trend Analysis with Google Chart.js
3. Store Clerk Trend Analysis Filters with Bootstrap, JavaScript and jQuery
4. Create API to use JSON in Android
5. Web Icon, Pressable Image Home and Browser Icon

Android Mobile Development:

1. Call API from ASP.NET
2. Bar Chart for Trend Analysis & Filters
3. Recycler View of Table for Trend Analysis
4. Splash Screen
5. HTTPS JSON with Cloud connection

UI Screen Prototype:

1. Store and Department dashboard (Web & Mobile)

Full Documentations:

1. Azure Machine Learning Documentation (19 Pages)
2. System Architecture Documentation, 3 Tiers (2 Pages)
3. Reliability Documentation (1 Page)
4. Current Optimization and Future Improvements/Recommends (3 Pages):

- Cloud, HTTPS Encryption and Optimization Performance
 - Algorithmic Efficiency
 - Readability and Minimize Repetition of Codes
 - Optimization of Performance and Memory Consumption with WOW64
 - CI/CD with JIRA
 - Hotlink Protection
5. Merge Documentations into Project Report
 6. Sequence Diagram

Amount of time spent on Project:

- Week 1- 4: Total 215 hours

Major Challenges and Resolution:

1. Challenge and Resolution
 - Machine Learning
 - Optimal accuracy for models, and configuring Web Services into ASP.NET
 - Resolved with data engineering and extensive research and reading on Microsoft documentations
2. Challenge and Resolution
 - DevOps and Cloud
 - Cloud Deployment and configuration connection issue, 500 Internal Server Error
 - Resolved with logs and creating security account
3. Challenge and Resolution
 - Web and Mobile Development
 - Code and algorithms not displaying as desired
 - Resolved with extensive debugging, pair programming and discussions

Experience:

- Machine Learning Integration with ASP.NET MVC
- DevOps and Cloud Configuration
- Web and Android Native Mobile Development
- Teamwork and Coordination, Pair Programming
- Git Hub and Agile Methodology (Minimum Viable Product)

5. Ngo Vu Hanh Nguyen

- Roles and Responsibilities Played
 - Developer in a Scrum Team
 - Full stack Developer
- Deliverables and Interaction within the team
 - a. Front-end Developing:**
 - Purchase Order UI
 - Data Table with Bootstrap JS and CSS
 - Department in Store side UI
 - b. Back-end Developing:**
 - Create the ER diagram (with Benedict)
 - Create Class Diagram
 - Department CRUD
 - Purchase Order CRUD (work with Kyaw Thiha)
 - Update Stock Quantity upon receiving items in a Purchase Order from Supplier
 - API for PO List, PO items List according to Supplier ID (with Kyaw Thiha)
 - Saving the new PO create data from Android (with Kyaw Thiha)
 - Validation for Purchase Order, Department CRUD, Supplier
 - c. Android Developing** (work with Kyaw Thiha)
 - UI view for PO form
 - Receive JSON data for Purchase Order
 - Input data in Purchase Order form
 - Sending data by JSON to backend for PO newly created data
 - Create Option Menu
 - d. Documentation**
 - Store UI Specification in Webpage and Android
 - Sequence Diagram
 - ERD Diagram
 - Class Diagram
 - Unit Testing for Use case (Purchase Order)
- **Major Challenges and Resolution**
 - Learn to connect Visual Studio with Android
 - Learn to develop native Android with HTTP API POST
 - Learn to use GitHub and resolve the conflict when merge the code
- **Quantum**
 - 4 weeks =>152 hours

- **Experience**

- Learn to resolve the conflict when merging the code
- Collaborate with team.
- Learn HTTP API Post and Get to pass data between Android and Visual Studio
- Validation, time management

6. Saw Htet Kyaw

- Roles and Responsibilities Played
 - Full stack programmer
- Deliverables and Interaction within the team
 - Screen design for store dashboard (with Lance and Shermaine)
 - State chart diagram for delegate employee and department requisition (with Lance)
 - Sequence diagram for delegate employee use case (with Lance)
 - Implement Bar Chart and Trend Analysis (with Shermaine)
 - Implement Delegation of Employee (with Lance)
 - Login Validation for web (with Lance)
 - Routing Security (with Kyaw Thiha and Lance)
- Major Challenges and Resolution
 - Deciding which function is more important for users as the timeline is very strict and we have only 7 members
 - Assigning tasks to right person to get the synergy and complete the project more effectively
- Quantum
 - 4 weeks (approximate 200+ hours)
- Experience
 - Finished a project as a team
 - Making decisions is important
 - Experienced how to handle in such a tight timeline

7. Yeo Jia Hui (Lance)

- **Roles and Responsibilities Played**
 - Web and Mobile developer
 - Responsible for backend and frontend of web and android
- **Deliverables and Interaction within the team**
 - **UI Prototype design**
 - Screen design for delegate employee
 - Screen design for store dashboard (with Saw Htet Kyaw and Shermaine)
 - **Documentation**
 - State chart diagram for delegate employee and department requisition (with Saw Htet Kyaw)
 - Sequence diagram for delegate employee use case (with Saw Htet Kyaw)
 - Use Cases diagram for department and store side
 - User volume table (with Ayisha)
 - Test Plan for delegate employee
 - User manual – screen annotations for department web version
 - User manual-screen annotations for store mobile version (with Hanh Nguyen)
 - **ASP.NET C#**
 - Implement delegation of employee (with Saw Htet Kyaw)
 - Login Validation for web (with Saw Htet Kyaw)
 - Routing Security (with Kyaw Thiha and Saw Htet Kyaw)
 - Implement API to android, GET Request for Department Controller (with Benedict)
 - Implement POST API to allow user to login from android (with Benedict)
 - Implement GET API to logout user from android (with Benedict)
 - Implement GET API and POST API for store controller of store clerk (with Benedict)
 - Implement store supplier controller methods (with Kyaw Thiha)
 - **Android Studio**
 - Implement delegate employee method in android (with Saw Htet Kyaw)
 - Implement disbursement list view in android (with Benedict)
 - Implement disbursement packaging view in android (with Benedict)
 - Implement the display of requisition form for employee in android (with Benedict)
 - Implement the Employee requisition form activity to pass data to recycler view in android (with Benedict)
 - Implement the retrieval of items list according to supplier data in android (with Kyaw Thiha)
 - Implement class to download JSON data from API (with Benedict)
 - Implement class that downloads JSON data using GetRawData class and converts the JSON data to a DeptRequisition object and passes the

DeptRequisition object to storeclerkRequisitionList Activity in android by making use of the call-back design pattern (with Benedict)

- Implement login activity that uses PostJsonData class to communicate with login API (with Benedict)
- POST JSON data and GET response from API in android (with Benedict)
- Implement PurchaseOrderCreate method in android (with Kyaw Thiha)
- Implement StockListActivity class to display a list of stocks for the store clerk in android (with Benedict)
- Implement StoreClerkDisbursementListActivity to display a list of disbursements for the store clerk (with Benedict)
- Implement StoreClerkDisbursementPackingActivity class to allows the store clerk to select a date for the dept rep to collect the disbursement (with Benedict)
- Implement StoreClerkRequisitionDetailActivity to display the details of a specific requisition (with Benedict)
- **Major Challenges and Resolution**
 - Learn to master new technology like API calling
- **Quantum**
 - Week 1 – 4 (200 hours)
- **Experience**
 - Learn to work under pressure
 - Learn to step out of my comfort zone and embrace new technology