<u>CAPSTONE PROJECT 1 – DOCUMENT STORAGE SOLUTION</u>

You are required to write a software solution for a document storage solution using Microservices Architectural Style. The software needs to include the following features:

1. It should store details of users
   - Username
   - Password
   - Email
   - Last login timestamp
2. It should store the details of the document uploaded (Internet and Intranet DB)
   - File type
   - File name
   - File size
   - File ID
   - Uploaded by
   - Uploaded timestamp

Functional Requirements:

1. All microservices should be REST APIs
2. Each microservice should have its own database, if necessary
3. User must login before uploading files
4. Files allowed for upload includes jpeg images, pdf and word documents not exceed 2 megabytes in size
5. Due to the data security restrictions within the organization the upload is done on an Internet facing web interface. The file upload microservice will do the following:
   a. Store the uploaded file temporarily on the document database on the Internet zone
   b. Push the files via MQ protocol from the Internet zone into the Intranet zone document database to store the files.
6. The Intranet zone will consist of a Java batch job which will pick up the files from the MQ on an hourly basis to process and store the files into the Intranet zone document database. After every run it will generate a reconciliation file which will be sent via a reconciliation microservice to the Internet zone application to notify them which files can be removed from the temporary document database.
7. Upon receiving the reconciliation file, another Java batch job will clean up the Internet zone document database to remove files which have been successfully stored in the Intranet zone document database.
8. High level functional flow as follows:

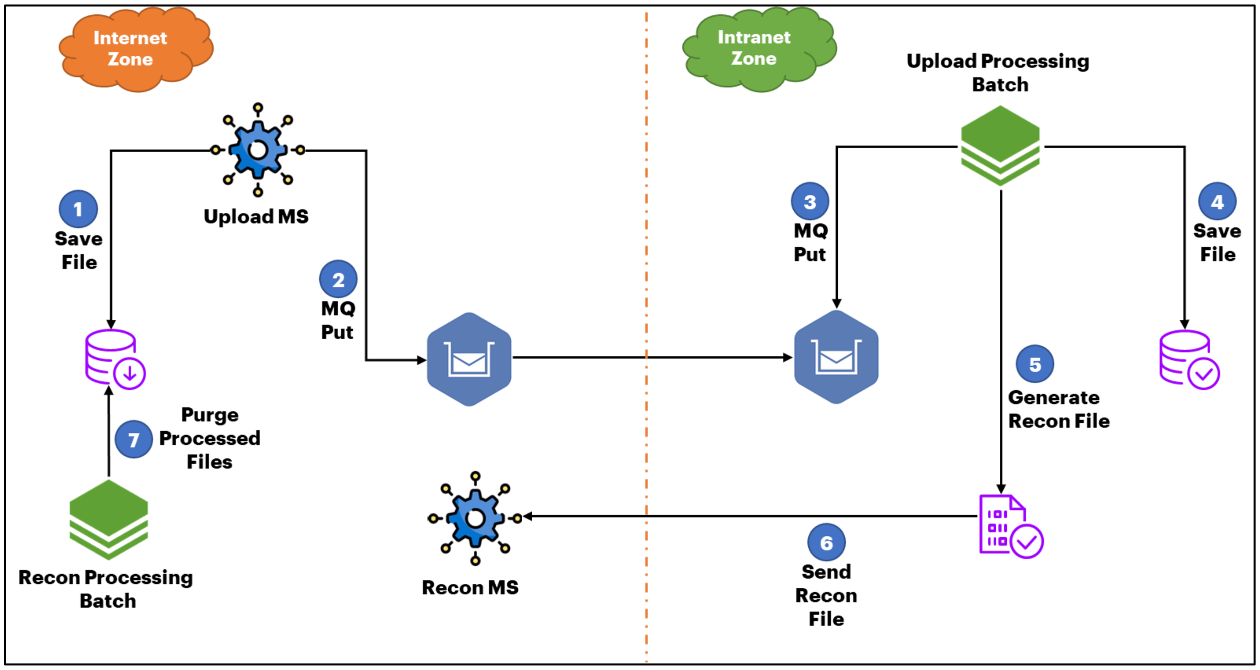

Figure 1: Capstone Project 1 - High Level Functional Flow

Submission:

1. Submit source code in GitHub repository
2. Write up on solution considerations and application demo

CAPSTONE PROJECT 2 – APPLICATION SUBMISSION PORTAL

You are required to write a software solution for an application submission solution using Microservices Architectural Style. The software needs to include the following features:

1. It should store details of users
    - Username
    - Password
    - Email
    - Mobile Phone
    - Last login timestamp
2. It should store the details of the applicant (Internet & Intranet)
    - Name
    - Race
    - Date of Birth
    - Country of Birth
    - COVID Vaccination Status
    - Application Submission Date/Timestamp
    - Application ID
    - Application Status
3. It should store the notification details of the application
    - Notification ID
    - Application ID
    - Message
    - Recipient Email
    - Notification Send Date/Timestamp

Functional Requirements:

1. All microservices should be REST APIs
2. Each microservice should have its own database, if necessary
3. User must login before submitting application
4. User only allow to submit if they are 18 years of age and above
5. Application information will be stored in JSON format
6. Due to the data security restrictions within the organization the submission is done on an Internet facing web interface and the application processing microservice will store the new application details temporarily on the internet database
7. A daily Java batch job will consolidate the day's applications into an encrypted flat file to send over to the Intranet processing system via SFTP. After the SFTP is successful the day's applications will be purged from the Internet database.
8. The Intranet processing system will read the flat file and store the application information into the Intranet database for application processing.
9. Upon successful processing of application, the Intranet processing system will trigger the notification microservice to send out an email notification to the applicant to notify them on the application outcome.
10. High level functional flow as follows:

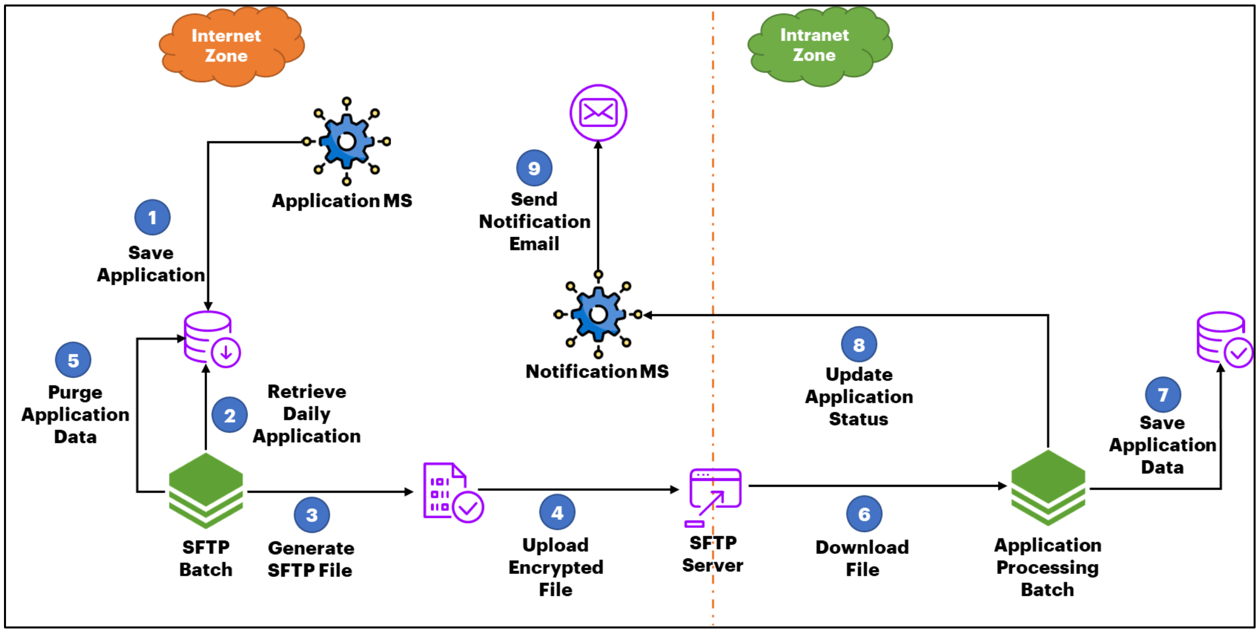

Figure 2: Capstone Project 2 - High Level Functional Flow

Submission:

1. Submit source code in GitHub repository
2. Write up on solution considerations and application demo