Roll Of Project

We will have to accomplish the following items in our microservices implementation:

1. Each microservice exposes a set of [REST](https://capgemini.sharepoint.com/:p:/r/sites/COPNA/iTransform/_layouts/15/Doc.aspx?sourcedoc=%7baa652949-5bc0-4bc1-9ed6-daaaf5e80dfb%7d&action=edit&uid=%7bAA652949-5BC0-4BC1-9ED6-DAAAF5E80DFB%7d&ListItemId=44&ListId=%7bAF7717D0-65AD-4DE5-AA01-07A399F026B5%7d&odsp=1&env=prod)/ JSON endpoints for accessing business capabilities
2. Each microservice stores its own persistent data using MySQL server database.
3. Each Microservices must implement best practices such as, Exception Handling, Loggers, [Test Cases](https://capgemini.sharepoint.com/:p:/r/sites/COPNA/iTransform/Shared%20Documents/Java_Full_Stack/JUnit.pptx?d=wc2cf6dc123254f87b0d4f6fca415175f&csf=1&e=U8adCg), Static Code Analysis and build tools.
4. [Microservices](https://capgemini.sharepoint.com/:p:/r/sites/COPNA/iTransform/_layouts/15/Doc.aspx?sourcedoc=%7b6db783b4-14e2-4813-9b7f-b14d1baee132%7d&action=edit&uid=%7b6DB783B4-14E2-4813-9B7F-B14D1BAEE132%7d&ListItemId=55&ListId=%7bAF7717D0-65AD-4DE5-AA01-07A399F026B5%7d&odsp=1&env=prod) are built with Spring Boot, which has an embedded IIS server as the HTTP listener.
5. The UI, Website must be implemented using [Angular](https://capgemini.sharepoint.com/:p:/r/sites/COPNA/iTransform/_layouts/15/Doc.aspx?sourcedoc=%7baff37771-4cd8-43df-bc7f-cc1a825cf2b7%7d&action=edit&uid=%7bAFF37771-4CD8-43DF-BC7F-CC1A825CF2B7%7d&ListItemId=50&ListId=%7bAF7717D0-65AD-4DE5-AA01-07A399F026B5%7d&odsp=1&env=prod)

You are also supposed to implement necessary design patterns mentioned in the table below: -

|  |  |
| --- | --- |
| **Design Pattern** | **Design Pattern Summary** |
| Repository pattern in Web API | A Repository Design Pattern acts as a middleman or middle layer between the rest of the application and the data access logic. Repository Design Pattern Mediates between the domain and the data mapping layers using a collection-like interface for accessing the domain objects. |
| Singleton pattern in Angular | is a design pattern in which a certain class can have only one instance. This is useful when you need to have a single instance of a class, but you don't want to create a new instance every time you need it, and also useful if we want to share resources or data. |

Requirements:

**Scenario:**

Management of company is facing problem while Roll Off of employees. In early days it was happening only through mail so there was a chance of miscommunication. To improve this scenario there should be a common platform where this process can work smoothly.

**Solution:**

Develop a project which helps to Roll of the employee. This project should act as a communication among Accounts team, PSP and admin. This project provides the single page application due to this direct communication will be possible. We make a web application that automates and minimize the time off roll off process via email. Roll off must be shared within 30 days of notice period along with the complete details. In this application we can ensures that if employee are on probation and there is performance issue the respective Roll Off will not be considered. The respective program manager will have to connect with HR for confirmation or de-confirmation of probation period. User should select the exact reason of roll off it will make application transparent and more flexible.

**Features:**

**For Accounts –**

Sign Up /login:

* Sign/login up with basic details. A user can login using Email and Password.

Profile-

* Accounts can view details, roll off and employee list and fill feedback form

View details –

* Employee No, Name, Local Grade, Main Client, Email Id , Project Code , Project Name ,Global Group Id.

**For PSP Team-**

Sign Up /login:

* Sign/login up with basic details. A user can login using Email and Password.

Profile-

* PSP team can see the Roll Off request and can approved it or reject it.

**Admin Panel:**

Sign Up /login:

* Sign/login up with basic details. A user can login using Email and Password.

Profile-

* Admin can see the requests when there is critical case like accounts want roll off but PSP team don’t want.