Broad Objectives for the Capstone Projects:

As a summative evaluation, learners in small groups are required to take up case study and complete. Intent of the case study is to provide a time bound activity for learners in small groups to implement and demonstrate all the knowledge and skills they have gained during the full program. Each small team would include minimum of 3 learners. Each group will get a abstract problem statement that briefly states the problem or very high-level outcomes. Each group must work closely as an Agile team - to understand the broad requirements, decompose the problem statement, select the use cases to be done, design the solution using micro services based architecture, design and build SPA, integrate with back-end APIs, build test cases, persist data as needed and containerize the application for deployment if need be.

Below checkpoints are evaluated/are checked in every project. By covering below characteristics in the selected product/project, a lot can be learned at design level as well at code level from the participants.

Sample Checks:

1. **Micro services** - Business process & computational challenge driven microservices, with neater separation of concerns
2. **DevOps** - Git branch strategy, Automated unit testing, API testing, UI BDD testing, deploy every merge to identified branch
3. **Single Page App** - Having 3 to 4 good views, where component hierarchy and cross component state management is happening

Additional Challenges / CheckPoints

1. **Polyglot data model system**
    a. Multiple data model (Mysql and MongoDB) in the system implementation factored
    b. 12 Factor App / Deployment Strategy
2. **Reactive system**
    a. System reacts to data
3. **Multi Device Experience if it's an Open App**
    a. Consistent multi device experience

Name : MedicalService System

The objective of the project is to provide essential medical services online to users irrespective of their location. Users can connect through their home internet or approach any nearby kiosk to get these services. The motivation to build the system is that very few or no doctors are available at remote locations, limited hour services and lack of sophisticated medical equipments and no patients history/lab data management. The proposed system is a web based application which is available all the time. The system provides details of medical services online and allows users to interact with doctors and other medical personnel.

Functional Requirements

Administrator module:

Administrator regularly takes back up of all kinds of data. Administrator can view the log information. Administrator also generates system reports. Administrator also provides online help manual for patients.

Doctors module:

A doctor must register with the system. Doctor's can manage their profiles. Doctor's give appointments to patients, e-prescriptions, and view patient's history. Doctor can interact with patient using live chat.

Patient module:

A patient must register with the system. Patients can make online appointment, view their previous health records and doctor's prescriptions. Patients can manage their profiles. Patient can interact with doctor using live chat. In case of any medical error (wrong medication or lab report) patient can register a complaint. Patient's grievance and feedback goes to Admin he can forward it to any doctor to answer.

Kiosk module:

Kiosk Manager views and adjusts appointments, performs day open and close activities and calculates his commission.

Non functional requirements

1. Application should be responsive to display consistently across multiple device screens.
2. Application needs to have separate front-end and back-end.
3. Application needs to interact with a database for relevant data processing.
4. Application should have been tested with success/failure instance

Tools and Technologies to be used	1. Front End : Angular 2. Back End : Spring Boot with Microservices Patterns. 3. Database : MySQL & MongoDb 4. Testing : Postman, Swagger, Mockito, Junit5, Cypress