SLIIT
Discover Your Future

IT3030 Programming Applications and Frameworks

Assignment 2020 S1 – Group Project

- **Focused Learning Objectives**: LO1-5
- **Weightage**: 30%
- **Deadline**: 10th week – 19th April 2020
- **Submission**: As explained in the text

Scenario

HelthCare is a hospital management system where the registered users can make appointments with the registered doctors who visit the registered hospitals. The users can even make the payments for the appointments online.

Things to do

- Identify the requirements and workflows needed to implement this system.
- Identify possible web services to be implemented as micro-services (at least 5 – there should be a service per team member).
- Design the architecture for the **HelthCare** system, indicating the identified web services, database(es), and the communication between them.
- Design the API for each web service.
- Design the class diagram per web service (use styles and patterns like MVC)
- Identify the DB requirements and design the DB(s) (if there are multiple DBs each should be designed).
- Select the tools (concepts, technologies, frameworks, libraries, plugins, IDEs, etc…) to develop each web service and justify the selection.
- Develop the system and test (using test client(s) or any other proper testing tool).
    - **HINT**: It is not mandatory to develop a comprehensive client-side application.

Things to concern

- The entire system with integrated services should be able to handle complete workflows as a complete system.
- Web services may intercommunicate to complete workflow(s).
    - **HINT**: You may need to maintain some data in the client-side when communicating with multiple services.
    - **ADVANCED HINT**: The services may need to communicate with each other (specially for authentication and authorization).
- Each service should perform CRUD operations on DB(s) to execute the service relevant features.

Project management

- All the documents, source, and related artefacts should be maintained in a public repo.
    - Use GitHub. There should be a comprehensive commit log from the beginning of the project where all the members have committed.
    - The last commit should be on or before the submission deadline.
- Document the roles and the workload of each member in the team.
- Each member has to take care of a dedicated web service, including documentations, designing, development, and testing.
- Each web service needs a client app (a test client or any other tool) to test the service.
- There should be a way to test the overall system, demonstrating how the complete workflows are running.

Submission

- Prepare reports, following the given templates and submit to the course web, before the deadline.
- Every member in the group has to submit the same copy of the report.
- A submission link will be created in course web.
- Submit a hard copy of the report at the practical class. (It should be the same version you submitted to the course web).

Report template (overall report) – Max 20 pages including the cover page

- **PDF file name**: IT3030PAF2020_GroupProject_Group<group number>
- Cover page (Project title, assignment details, team number).
- Table of contents.
- Members' details (student ID, name, workload distribution) – You may use a table.
- Clickable link to the public VCS repo, followed by the commit log (commit log can be taken from the GitHub).
- SE methodology/methods (with justification).
- Time schedule (Gantt chart).
- Requirements
    - Stakeholder analysis (onion diagram).
    - Requirements analysis (Functional, Non-functional, Technical requirements).
    - Requirements modelling (Use case diagram).
- System's overall design (the diagrams should be followed by descriptions).
    - Overall architecture.
    - Overall DB design (ER) (only if you are using a single DB for all the services).
    - Activity diagrams (to show the overall workflows).
    - Any other relevant design diagrams for the overall system.
- Individual sections (use the template below).
- System's integration details. (techniques used, how was it tested, etc.).
- References.
- Appendix.

Individual sections may follow the template below.

- Service design (focusing only on your service. The diagrams should be followed by descriptions)
    - API of the service. (Discuss the design rationale)
    - Internal logic (Class diagram, activity diagrams, flowcharts, etc…).
    - Database for the service (ER) (only if you are using a separate DB for the service).
    - Any other relevant design diagrams.
- Service development and testing.
    - Tools used, including justifications for their selection. **HINT**: Use dependency management tools, testing tools, code quality checking tools, etc.
    - Testing methodology and results.
- Assumptions and any other relevant discussions should be discussed on relevant sections.

Marking scheme

Group mark (based on the overall report and demonstration)	
Criteria	**Weight**
1. Presentation (formatting, neatness, arrangement, proper sectioning and discussions)	05
2. Member details and roles	05
3. VCS repo management (with a comprehensive commit log, which shows how have you worked from beginning to the end?)	05
4. SE methodology	10
5. Realistic time plan	05
6. Requirements (stakeholders, analysed requirements, use case diagram)	10
7. System design	30
8. Any other design artefacts	10
9. Development (Demonstrate the overall workflows using a test client)	20

Individual mark (based on the report's individual section, demonstration, and viva)	
Criteria	**Weight**
1. API	20
2. Internal logic design	20
3. Database design	10
4. Any other relevant design diagrams	05
5. Development tools selection and justification	05
6. Testing methodology and results	05
7. Code quality	05
8. Development (CRUD operations. Features should work without bugs. Demonstrate the functionality using the test client)	20
9. Viva	**10**

Final mark of the student = (Group total x 40%) + (Individual total x 60%)

NOTE:

- Even you have good marks for the 1 to 7 components of the individual section, if you do not perform well for the viva, only 50% of your individual mark will be given based on the assumption that you have completed your part getting help but you do not have enough knowledge.
- If you do not contribute enough for the group work (have complaints and issues), only 50% of the group mark will be given.
- If you do not attend for the scheduled viva, you may have to submit valid reasons with documents, then another date will be scheduled based on the validity of your reason and documents.