Phase #2: Specification and Design

Schedule: **Week #3 – Week #6**

Deadline: **Mar 31, 2026**

Points: **30/75**

Instructions

Prepare specification and design documentation of your cloud application project. Document must include requirements analysis, architecture design and functional description.

You will present specification and design of your cloud application project in week #7 during lecture and practice sessions. **Duration of the presentation should not exceed 12 minutes and each member of the team must present a part of the presentation.**

Upload your specification and design document in PDF format.

Hints

Requirements analysis should focus to answer following questions:

- What are the use cases and usage scenarios of the application?
- What and from where are the data collected? (If your application is either in IoT or in data processing domains.)
- What is the link to the OpenLab environment (or other external resources)? (If your application requires OpenLab infrastructure and/or other external resources.)
- How do you distribute the application functionality?
- What is the workload of individual components?
- How and where does the application handle status and data, and how does it consider multi-tenancy data management?
- How are specific components (user interface, processing, data access) designed with respect to multi-tenancy?
- How is the elasticity and liveness of the cloud application managed?

Specification and design documentation of your cloud application must contain:

- Use cases and usage scenarios description.
- Conceptual model or system architecture of the application with respect to microservices (or other components) distribution, backend/frontend layers, etc.
- Give a good reason for the application microservices distribution and its deployment to cloud.

- Microservices interaction via messaging, consistency model for data shards. (Include also protocols for external data gathering e.g. in IoT application).
- Backend services design, API and data model of microservices.
- Frontend design, scenarios of interaction with users, and interaction with backend services via API gateway.
- Evaluate possible impact of application deployment to cloud in production.
- Present selected cloud technology and used cloud services.
- Use standard modeling tools when designing your solution (UML, ERD, pseudo-code, etc.)

Use capabilities of selected cloud technology to implement backend services:

- Prefer microservices architectural style over monolithic architectural style.
- Prefer data shards across individual microservices over centralized SQL database service.
- Reuse existing cloud services: messaging, API gateway, key-value storage, blob storage, MapReduce, notifications, authentication etc.
- Integrate existing third-party services when appropriate (e.g. authentication via Google, Facebook accounts).
- Consider serverless approach for design and deployment of lightweight components.
- Prefer containers over virtual servers.
- Study capabilities supported by various public cloud platforms (Azure, AWS, Google Cloud, Firebase, Heroku, etc.)

Possibilities of frontend implementation:

- Web application (SPA, portlets, meshup, etc.)
- Mobile application (native vs. cross-platform vs. PWA/hybrid).