

Microservices Architecture Assignment 2025 – Weighting 40%

A. Distributed Microservice Application

The purpose of this assignment is to demonstrate your ability to apply the learning from the module to build a distributed application composed of multiple collaborating services in the form of microservices. The application is expected to cover the following aspects from the module:

- REST based microservices built with Spring Web and Spring JPA
- Configuration management with Spring Cloud Config (File)
- Service discovery with Spring Cloud Netflix Eureka
- API gateways with Spring Cloud API Gateway
- Building resilient microservices using Resilience4J
- Login and Authentication (JWT, OAuth2)
- Implementing observability and monitoring of services

At a high level the overall project deployment should look similar to the following:

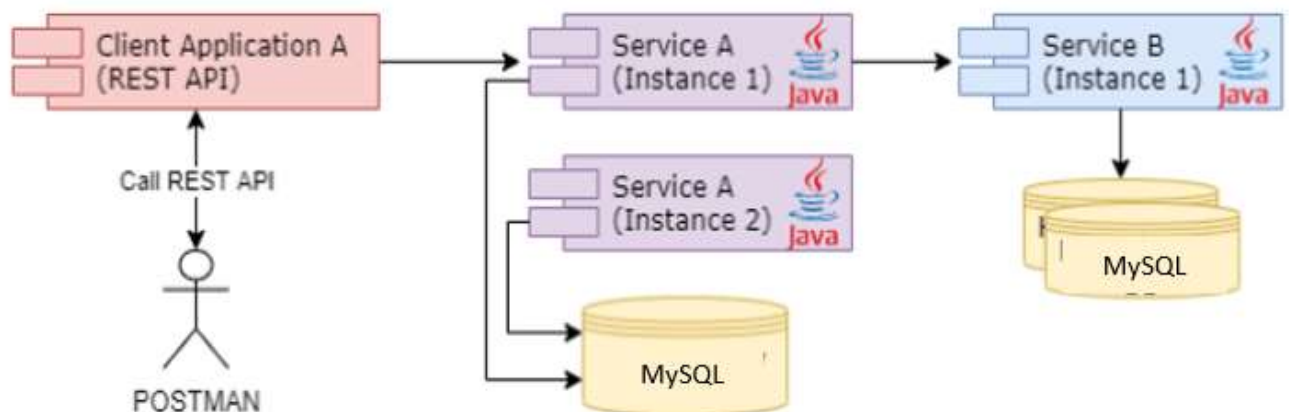


Figure 1 High Level Deployment

The application deployment shall consist of the following:

1. Minimum of **two** separate Microservices exposing resources. Be original.
2. Service B will expose entities stored in a database with CRUD (Create, Retrieve, Update and Delete) operations exposed via a REST API.
3. Service A will expose entities stored in database with CRUD (Create, Retrieve, Update and Delete) operations. This database will be shared by multiple instances of Service A. (Note : One instance of MySQL is sufficient for demo)

4. Service A will also make REST calls to retrieve data from Service B
5. A Client application will expose endpoints that can be invoked by POSTMAN (a REST client) and is responsible for invoking REST calls in Service A.

B. Application Domain

When deciding on the two Microservices and their entities, come up with a suitable Application idea. Model two resources that have a sensible relationship between them. Do not just name them “ServiceA” and “ServiceB”.

C. Detailed Deployment

The purpose of the application is to demonstrate the following:

1. Microservices development using Spring Web to expose REST API and using Spring JPA for Database interaction.
2. Each microservice instance should register with a Spring Eureka discovery service on start-up.
3. Each microservice should retrieve configuration data from a Spring Cloud Configuration Server on start-up. The configuration service can use a simple File System backend or git (recommended).
4. All REST calls between the Client and the Service are expected to be routed via a Gateway which uses the Eureka discovery service.
5. The application should use Authentication (e.g. JWT, OAuth2).
6. REST calls are expected to be traced or monitored.
7. The REST endpoints should use some form of data validation using the Validation API.
8. The REST APIs should conform to proper API design – with correct formulation of URL's and correct status codes (Ref ONAP link).

d. Deadlines

Note 1: Be mindful that late submissions might not be corrected in time due to cutoff dates for exam results. If you cannot submit on that date due to illness or other extenuating circumstance you will have to request a deferral from eng@tus.ie

Note 2: A live Q&A may be scheduled after the submission date if deemed necessary.

Submission Date
Monday 14 th April 2024

e. Submission

	What to submit
Report	Introduction to application, rationale and context
	List of User Stories that are completed
	Description of Microservices and their relationships
	Description of the entity classes in the application including the validation annotations
	Description of Cloud Native features implemented (e.g. service discovery, config)
	Test results – screenshots of the output (POSTMAN response body/headers, traces, show Cloud Native behaviours)
	Evaluation – Evaluation (1-2 pages) of how well you adhered to the project brief and any problems encountered.
Code	A .zip file of all code
Demo	Brief context and rationale of your application. Demo of key User Stories. Demo cloud native behaviours used (e.g Service discovery, config, resiliency) NB! Maximum time 10 minutes

f. Marking Rubric

Elements	Excellent (70+)	Good (55%-69%)	Satisfactory (40%-55%)	Fail (0-39%)
Presentation (10%)	The presentation is audible. The presentation provides evidence of adequate preparation. Content is presented in a logical and coherent structure. Appropriate use is made of audio/visual materials, which are clear and well organised.	The presentation is audible. The presentation provides evidence of adequate preparation. Content is presented in a logical and coherent structure.	The presentation is audible and the presentation provides evidence of adequate preparation.	Presenter does not engage the viewer. Presenter does not adhere to the maximum time limit.
Context and Rationale (10%)	A concise synopsis of the context and rationale of the application is given. Presenter displays evidence of originality.	A concise synopsis of the context and rationale of the application is given.	A concise synopsis of the context of the application is given.	Presenter did not provide a context or rationale for the application.
User Stories/ Application Demo (50%)	Presenter demonstrated a complete application. The presenter described and rationalised the application. Evidence of understanding of the technology used. Ability to identify any key limitations. REST API well followed design principles – ref ONAP. HATEOAS considered. Cloud native behaviours(Eureka, Config, Resilience, Gateway, Security, Tracing) included.	Presenter demonstrated a more complete application with full CRUD and searches. Some cloud native behaviours included. Server side data validation. Evidence of understanding of the technology used	Presenter demonstrated an application with a good level of functionality (two microservices with CRUD actions communicating).	Presenter demonstrated no or minimal functionality. No server side validation of data. No cloud native behaviours (e.g Cloud Config, Service Discovery, Load Balancer)

Screencast (70%)

Ctd→

Elements	Excellent (70+)	Good (55%-69%)	Satisfactory (40%-55%)	Fail (0-39%)
Organisation and Presentation (5%)	Report demonstrates a high level of competency in the subject area. Content is presented in a logical and coherent structure. Appropriate diagrams. Consistent formatting.	Content is presented in a logical and coherent structure. Appropriate use diagrams and tables.	Content adequately presented. Inconsistent formatting. Minimal use of diagrams.	Content is inadequate. Poor formatting.
User Stories, 5%	User Stories follows INVEST and fully reflects the functionality and vice versa. Acceptance criteria listed.	User Stories follows INVEST with Acceptance criteria.	User Stories partly follows INVEST principles.	Poor User Story formulation,
Architecture and REST APIs)5%	Good architecture description and REST end points documented. API documented using Swagger. Good API design and HATEOAS considered.	Good architecture description and REST end points documented	No or minimal architecture description. REST end points partially documented	No architecture description. REST end points not documented.
Cloud Native Behaviours & Security 5%	High level of competency and use of cloud native behaviours demonstrated. Good use of diagrams were appropriate.	Good use of some of the Cloud Native behaviors	Partially use of some of the Cloud Native behaviors.	No Cloud Native Behaviour described or implemented(Discovery, Load Balancer, Config, Tracing and Authentication.
Evaluation(10%)	Bugs and improvements in application identified and suggested solutions described. Any limitations of the architecture identified. Application evaluated against the project brief.	Bugs and improvements in application identified and suggested solutions described. Application evaluated against the project brief.	Bugs and improvements in application identified and suggested solutions described.	Minimal or no evaluation

Report and Code (30%)

Live Q&A: If I have questions or queries on your submission, a Live Demo/Q&A session over zoom will be scheduled after the submission date.

CheckList:

- Upload to moodle by Monday 14th April. 2024 17.00. .zip with your **code, screencast and report.**

- Live Demo/Q&A via zoom : You will be notified via student email if you need to participate in a Live Q&A.