

## ID303911 - Mobile og distribuerte applikasjoner

### Obligatorisk Oppgave #2

Oppgave 2.9.2019

Avdeling for ingeniør- og realfag

NTNU i Ålesund

### Om Laboratorieoppgavene

Laboratorieoppgavene må gjennomføres individuelt. For å få godkjent en oppgave må du vise følgende:

- Leverer inn oppgaven i Blackboard innen fristen
- Viser at programmet fungerer som instruert i oppgaven.
- Viser at du har benyttet SCMS (git) for oppgaven.
- Besvarer muntlig noen kontrollspørsmål.

## Innledning

Det skal utvikles en tjenerløsning for klienten utviklet i oppgave #1.

Tjenerdelen skal implementeres i Java og benytte seg av EJB, JAX-RS, JSON-B, Bean Validation og JPA fra Java EE 8 spesifikasjonen. Klienten implementert i Obligatorisk Oppgave #1 skal tilpasses til å benytte seg av tjenesten implementert i denne oppgaven. Et løsningsforslag for Oppgave #1 vil bli gjort tilgjengelig og kan evt. benyttes. Systemet skal demonstreres til foreleseren.

## Oppgaver:

1. Autentisering og autorisering (Java EE Security)
  - a. En skal kunne opprette en ny bruker
  - b. Brukeren skal kunne logge seg på systemet og redigere sine ressurser.
2. Både registrerte og uregistrerte brukere skal kunne se hvilke varer som er til salgs
  - a. Alle entiteter skal ha fornuftige sjekker på sine attributter ved hjelp av Bean Validation. (f.eks. @Email)
3. En innlogget bruker vil kunne legge til og kjøpe varer.
4. Om en bruker kjøper en vare, vil kjøperen bli registrert i systemet som kjøper.
5. Opprette en tjener på autodeploy.uals.no med Ubuntu 18.04 som template
6. Kjøre oppgave #2 i Docker på tjeneren.

## Artikler om Java / Jakarta EE 8

- JAX-RS - Java API for RESTful Web Services
  - [Overview of JAX-RS \(Part 1\)](#)
  - [What are JAX-RS Annotations? \(Part 2\)](#)
  - [Overview of JAX-RS Annotations \(Part 3\)](#)
  - [What is javax.ws.rs.core.context? \(Part 1\)](#)
  - [What is javax.ws.rs.core.context? \(Part 2\)](#)
  - [What is javax.ws.rs.core.context? \(Part 3\)](#)
  - [What is javax.ws.rs.core.context? \(Part 4\)](#)
  - [Foundations of RESTful Architecture](#)
- JSON-B - Jakarta JSON Binding
  - [I - The JSON Binding API in a nutshell](#)
  - [II - Default mapping with JSON-B](#)
  - [III - Custom binding with JSON-B](#)
- Bean Validation
  - [Bean Validation 2.0 - What's in it?](#)
  - [Bean Validation in Jersey](#)
  - [Java Bean Validation Basics](#)
- EJB - Enterprise JavaBeans
  - [A Detailed Guide to Enterprise Java Beans w/Code Examples](#)
  - [Contexts and Dependency Injection for the Java EE Platform](#)
- JPA - Java Persistence API
  - [Getting Started with JPA](#)
  - [JDBC Best Practices](#)
- Java EE Security
  - [Java EE Security Essentials](#)