UniPZ Project Documentation

Punoi: Mehmed Abidini

1 Përmbledhje

Shtresa   Teknologjia   Porti Zhvillimor

Frontend SPA   Angular 20 + Vite   4200
Backend REST   Spring Boot 3.3 (Java 21) 8080
DB   MySQL 8 (dev) / H2 (test) —
Auth (ops.)   Spring Security + JWT —

2 Arkitektura & Diagramet

2.1 Diagrami i Lartë (UML C4 — Container View)

[Browser] → [Angular SPA] → [/api] → [Spring Boot] → [MySQL]

Comunication: HTTP

Contract: JSON REST + CORS

2.2 Relacionet e Entiteteve (ERD)

Department 1—∞ Semester 1—∞ Subject

FK: semester.department_id, subject.semester_id

3 Backend

3.1 Strukturë e Paketave

```
com.unipz
  config          / CORS, Security
  controller       REST Controllers
  dto             DTO & Mapper
  model           @Entity JPA
  repository       JPA Interfaces
  service         Business Logic
```

3.2 Konfigurime

```
spring.datasource.url jdbc:mysql://localhost:3306/unipz?
useSSL=false&serverTimezone=UTC
spring.jpa.hibernate.ddl-auto    update (dev)
spring.jpa.show-sql   true
server.port      8080
```

3.3 API-të Kryesore

Metoda   URI   DTO Input DTO Output Përshkrim

GET /api/departments —   DepartmentDTO[] liston të gjithë departamentet me semestra & lëndë
POST /api/semesters   SemesterDTO SemesterDTO shton semestër (admin)
PUT /api/subjects/{id}   SubjectDTO SubjectDTO përditëson lëndë
GET /api/subjects/semester/{id} —   SubjectDTO[]   lëndët e një semestri

Shembull REQUEST / RESPONSE

```
PUT /api/subjects/4
Content-Type: application/json
{
  "id": 4,
  "name": "Databaza 2",
  "description": "Transaksione & indekse",
  "credits": 6,
  "classroom": "B-302",
  "time": "Fri 10:00-12:00",
  "professorName": "Dr. Ada",
  "semesterId": 3
}
→ 200 OK
```

3.4 Pseudo-kod për Business Logic

```
Subject update(Long id, SubjectDTO dto) {
  Subject s = repo.findById(id).orElseThrow();
  Semester sem = semRepo.findById(dto.semesterId()).orElseThrow();
  map(dto, s);
  s.setSemester(sem);
  return repo.save(s);
}
```

4 Frontend — Detaje Teknike

4.1 Strukturë

```
src/app
  models/*.model.ts
  services/department.service.ts
  components/
      home.component.ts
      about.component.ts
      departments.component.ts
      admin/
          admin-departments.component.ts
  app.routes.ts
```

4.2 Module-less Standalone Bootstrap

```
bootstrapApplication(AppComponent, {
  providers: [
```

```
    provideRouter(routes),
    provideHttpClient(),
    provideAnimations(),
    provideForms()
  ]
});
```

4.3 Fluksi i të Dhënave

Admin UI → DepartmentService.updateSubject() →
HTTP PUT /api/subjects/{id} → Backend Service → DB

4.4 Animacionet

```
export const semExpand = trigger('semExpand', [
  state('closed', style({ height: '0', opacity: 0, padding: '0',
pointerEvents: 'none' })),
  state('open',   style({ height: '*', opacity: 1, padding: '1rem',
pointerEvents: 'auto' })),
  transition('closed ⟺ open', animate('250ms ease'))
]);
```

Departamenti përdor fadeSlide 300 ms

Semestri përdor semExpand për shpalosje vertikale

4.5 Formulari i Adminit (Inline Edit)

```
<input [(ngModel)]="subj.name"/>
<select [(ngModel)]="subj.semesterId">
  @for (s of dept.semesters; track s.id) {
    <option [value]="s.id">Sem {{s.semesterNo}}</option>
  }
</select>
```

4.6 Error-handling në UI

```
saveSubject(subj){
  this.depSvc.updateSubject(subj).subscribe({
    next: () => this.toast('Ruajtur!'),
    error: () => this.toast('Gabim!', 'error')
  });
}
```

5 Database & Skema

```
DEPARTMENT  (id PK, name, description)
SEMESTER    (id PK, semester_no, academic_year, department_id FK)
SUBJECT     (id PK, name, description, credits, classroom, time,
             professor_name, semester_id FK)
```

Indekse

semester.department_id (FK + INDEX)

subject.semester_id (FK + INDEX)

6 Paneli i Administratorit

Hyrje direkte: http://localhost:4200/admin-departments

Ruajtja në vend (inline) → PUT /api/subjects/{id}

semesterId popullohet automatikisht në dropdown për të shmangur 500

8 Deployment

1. Frontend: ng build --base-href / → artefakti dist/

2. Backend: ./spring-boot run

3. Nën prodhim, Spring Boot shërben dist nga classpath:/static/ → kërkesa GET / hap app-in Angular.