

NAČRT TESTIRANJA:

OR-ECOSYSTEM

**Avtorji: Aljaž Kodrič, Rok Fonovič, Luka Car**

**Skrbnik: Alen Rajšp**

## **1. Namen in cilj testiranja**

Cilj tega načrta testiranja je opredeliti strategijo, obseg in potek testiranja sistema OR‑Ecosystems, ki vključuje dve spletni aplikaciji:

* **Aplikacija 1**: Prikaz realnočasovnih simuliranih medicinskih podatkov
* **Aplikacija 2**: Upravljanje medicinskih naprav in servisov

Testiranje bo zagotovilo, da so vse funkcionalne zahteve izpolnjene, da so izpolnjene varnostne, performančne in skladnostne zahteve ter da je uporabniška izkušnja stabilna in zanesljiva.

## **2. Področje testiranja**

**Vključene komponente:**

* Simulacija in pretok realnočasovnih podatkov (ISO/IEEE 11073)
* Frontend prikaz podatkov (grafi, preklapljanje naprav, izbira operacijske sobe in pacienta)
* Komponente za upravljanje medicinskih naprav (prikaz, filtriranje, dodajanje, urejanje)
* Integracija s HAPI FHIR strežnikom za pacientove podatke
* Generiranje PDF poročil medicinskih naprav
* Upravljanje uporabnikov (medicinsko osebje, administrator)
* Relacijska baza (PostgreSQL) in časovna baza (InfluxDB)

**Izključene komponente:**

* Fizična integracija z napravami (namesto tega simulacije)
* Mobilna aplikacija

## **3. Testne faze in vrste testov**

1. **Enotno testiranje (Unit Testing)**
   1. Test posameznih komponent backend API‑ja.
2. **Funkcionalno testiranje (Functional Testing)**
   1. Celovito preverjanje aplikacije 1 in aplikacije 2 v testnem okolju.
   2. Testiranje scenarijev: izbor pacienta, izbira sobe, vklop/izklop naprav, pregled grafov.
   3. Testiranje vlog: zdravnik, tehnik, administrator.

## **4. Testno okolje in orodja**

* **Okolje:** Docker kontejnerji na Kubernetesu (testni namespace)
* **Podatkovne baze:** PostgreSQL (relacijski) in InfluxDB (časovna)
* **Simulator:** knjižnica sdc11073 za generiranje in pošiljanje HL7/ISO podatkov
* **Orodja za testiranje:**
  + Jest / RTL za enotne teste frontend -JUnit / pytest za enotne teste backend
  + Postman / Newman za integracijske API teste
  + OWASP ZAP za varnostno skeniranje
  + JMeter ali k6 za performančne teste
  + Cypress za end‑to‑end testiranje