

Specifikacija projektnog zadatka

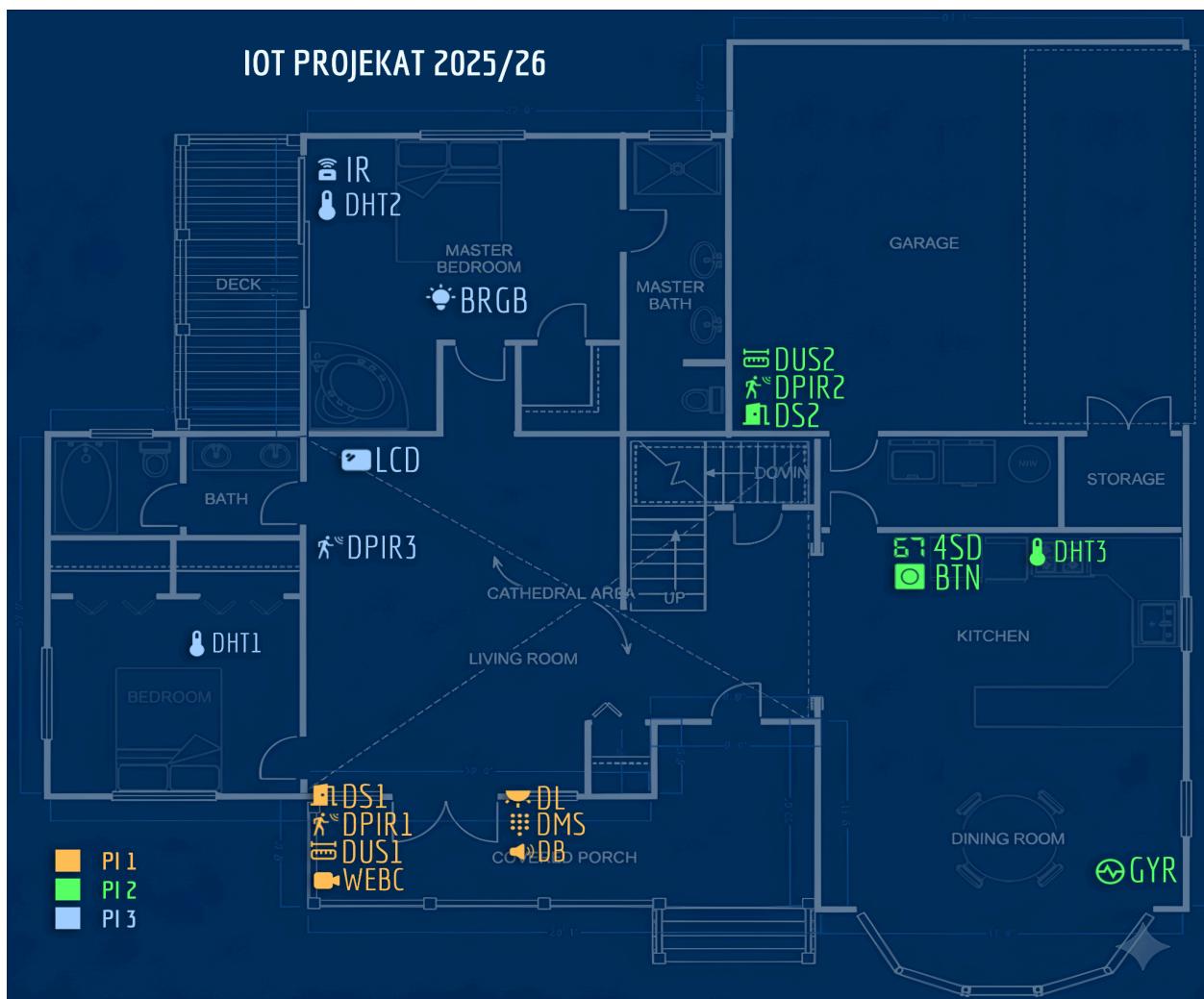
Inženjerstvo softvera za Internet/Web of things
2025/2026

Timovi

Projekti zadatak se radi u **timovima od po dva students.**

Standardni projektat

Projektni zadatak ima za cilj implementaciju uređaja pametne kuće. Nacrt kuće dat je na slici:



Sistem pamentne kuće sadrži tri Raspberry PI uređaja, koji su povezani sa različitim senzorima i aktuatorima. Spisak senzora i aktuatora dat je u tabeli:

PI	Kod	Naziv
PI1	DS1	Door Sensor (Button)
	DL	Door Light (LED diode)
	DUS1	Door Ultrasonic Sensor
	DB	Door Buzzer
	DPIR1	Door Motion Sensor
	DMS	Door Membrane Switch
	WEBC	Door Web Camera
PI2	DS2	Door sensor (Button)
	DUS2	Door Ultrasonic Sensor
	DPIR2	Door Motion Sensor
	4SD	Kitchen 4 Digit 7 Segment Display Timer
	BTN	Kitchen Button
	DHT3	Kitchen DHT
	GSG	Gyroscope
PI3	DHT1	Bedroom DHT
	DHT2	Master Bedroom DHT
	IR	Bedroom Infrared
	BRGB	Bedroom RGB
	LCD	Living room Display
	DPIR3	Living Room Motion Sensor

Kontrolna tačka 1

Za prvu kontrolnu tačku, potrebno je implementirati skriptu koja se pokreće na uređaju **PI1**. Omogućiti konfiguraciju skripte, tako da se bilo koji uređaj može, a ne mora simulirati.

Ulagane podatke sa svakog senzora potrebno je ispisati u konzoli.

Omogućiti upravljanje aktuatorima sa Raspberry PI uređaja kroz konzolnu aplikaciju.

Nije potrebno implementirati Web kameru.

Kontrolna tačka 2

Za drugu kontrolnu tačku, potrebno je proširiti skriptu koja se pokreće na uređaju **PI1**, a koja je bila implementirana za **KT1**. Proširiti konfiguraciju skripte, tako da ona sadrži dodatne informacije o tome na kojem *PI*-u radi i koje je ime uređaja (ovo je minimum, možete dodati još proizvoljan broj konfigurable informacija o uređaju, dokle god to ima smisla). Logiku je potrebno proširiti sa slanjem izmerenih/simuliranih vrijednosti MQTT protokolom na određeni topic (topic može biti konfigurabilan iz settings fajla a može biti i zakucan za svaki tip senzora), prilikom slanja specificirati putem *tag*-a da li je poslata vrijednost simulirana ili nije.. **Obavezno** je da se izmjerene vrijednosti šalju u *batch*-evima putem *daemon* niti (može i iz zasebnog procesa ko hoće da proba 😊). *Daemon* nit/proces može biti implementirana generički (1 nit/proces za sve senzore) a može se implementirati tako da imate za svaki tip senzora jednu *daemon* nit (**ko radi proces mora ga implementirati generički jer drugačije nema smisla**). Kako biste dobili sve bodove za KT2 neophodno je da se skripta implementira tako da **ne ulazi u deadlock**, kao i da delovi koda koji se zaključavaju *mutex*-ima budu **minimalni**. Potrebno je implementirati server (preporučujemo *Flask* biblioteku ali može šta god želite) koji će preuzimati poruke iz *MQTT broker*-a i čuvati ih u *InfluxDB* bazu podataka. Kod modelovanja baze podataka moguće je sve čuvati u jedan *bucket* ukoliko vam je tako lakše, ali isto možete podijeliti slogove (*record*-e) po *bucket*-ima kako god želite. Potrebno je omogućiti vizualizaciju podataka sa senzora i aktivacija aktuatora u *Grafana* alatu. Potrebno je da svaki tip senzora ima svoj panel za prikaz. Nije potrebno implementirati web kameru.

Obrana projekta

Implementirati skriptke koje se pokreću na sva tri *PI* uređaja. Omogućiti konfiguraciju skripti, tako da se bilo koji uređaj (*PI*, senzor, aktuator) može, a ne mora simulirati.

Implementirati serversku aplikaciju, koja će putem MQTT protokola prima podatke sa *PI* uređaja i skladišti ih u *InfluxDB* bazu podataka. Omogućiti vizuelizaciju svih podataka putem *Grafana* alata.

Drugim rečima, implementirati zahteve sa KT1 i KT2 za sve uređaje.

Implementirati korisničku Web aplikaciju u proizvoljnoj tehnologiji. Aplikacija treba da ima mogućnost prikaza podataka iz *Grafana* alata, kao i prikaz trenutnog stanja svakog elementa sistema.

Implementirati logiku na osnovu ulaza senzora:

ALARM - predstavlja stanje uzbune u objektu. Tokom ovog stanja, potrebno je da DB i BB bude uključen. Događaje ulaska i izlaska u ovo stanje potrebno je čuvati u bazi podataka i prikazati putem *Grafana* alata, kao i obavestiti korisnika putem Web aplikacije. Iz ovog stanja izlazi se unosom PIN-a na DMS-u, ili putem Web aplikacije.

1. Kada DPIR1 detektuje pokret, uključiti DL1 na 10 sekundi
2. Kada DPIR1 detektuje pokret, na osnovu distance detektovane pomoću DUS1 u prethodnih nekoliko sekundi ustanoviti da li osoba ulazi ili izlazi u objekat.

- a. Isto logiku primeniti na DPIR2 i DUS2
 - b. Čuvati brojno stanje osoba u objektu.
3. Ukoliko se detektuje signal sa DS1 ili DS2 na duže od 5 sekundi, uključiti ALARM dok se stanje DS-a ne promeni. Ovo simulira otključana vrata.
4. Omogućiti aktivaciju sigurnosnog alarma putem DMS komponente.
 - a. Kada se unese četvorocifreni PIN kod, sistem se nakon 10 sekundi aktivira.
 - b. Ukoliko je sistem aktivan, nakon detektovanog signala na DS1 ili DS2 senzoru, uključiti ALARM ukoliko se ne detektuje ispravno unet PIN na DMS komponenti.
 - c. Unosom PIN-a ALARM se isključuje i sistem se deaktivira.
5. Ukoliko je brojno stanje osoba u objektu spram tačke 2 jednako nula (ukoliko nema osoba u objektu), detektovanjem pokreta na nekom od RPIR1-3 senzora uključuje se ALARM.
6. Ukoliko GSG (koji je zakačen za slavsku ikonu) detektuje značajan pomeraj, uključiti ALARM.
7. Prikazati temperaturu i vlažnost vazduha sa DHT1-3 na LCD-u, tako da se ispis sa različitih DHT-ova smenjuje na nekoliko sekundi.
8. Omogućiti podešavanja kuhinjska štoperice.
 - a. Omogućiti da se putem Web aplikacije podesiti vreme na štopericu. Vreme prikazati na 4SD komponenti.
 - b. Omogućiti da se pritiskom na dugme (BTN) dodaje N sekundi na štopericu. N je broj sekundi koji se konfiguriše putem web aplikacije.
 - c. Kada istekne vreme na štopericu, potrebno je da 4SD treperi sa 00:00 na prikazu. Pritiskom na dugme (BTN) zaustavlja se treperenje.
9. Omogućiti uključivanje, isključivanje i upravljanje bojama BRGB sijalice putem daljinskog upravljača i IR senzora, kao i putem Web aplikacije.
10. Prikazati video sa web kamere na web aplikaciji.

Navedenu logiku realizovati na proizvoljan način.

Tokom odbrane projekata, za maksimalnu ocenu potrebno je povezati tražene senzore i/ili aktuatora na PI uređaj i demonstrirati njihov rad u sistemu, gde će ostali senzori/aktuatori biti simulirani. Za realizaciju ovoga studenti imaju 30min, i na raspolaganju sve materijale sa predmeta i pristup internetu. **VODITI RAČUNA** da se vaša skripta može izvršiti na PI uređaju - ukoliko koristite bilo kakve alate i biblioteke koje nisu bile korišćene na vežbama nemate garanciju da će biti dostupne na PI uređaju u učionici.

Ukoliko studenti nemaju implementirane čitave funkcionalnosti sa KT1 i KT2, automatski dobijaju 0 bodova na odbrani projekta.

Neophodno je jasno demonstrirati sve funkcionalnosti u roku od 15min. Sve funkcionalnosti koje ne budu demonstrirane u ovom vremenskom okviru neće biti ocenjene. Preporučeno je da omogućite način da se alarmi za svaku funkcionalnost po potrebi isključe kako bi prikaz bio jasan, kao i da implementirati scenarije koji mogu ručno da se okinu za svaku funkcionalnost.

Proizvoljni projekat

Studenti imaju mogućnost da sami definišu predmetni projekat sa proizvoljnom temom.

Za potrebe projekta, potrebno je da studenti sami obezbede neophodnu opremu.

Kontrolna tačka 1

Za prvu kontrolnu tačku, potrebno je priložiti specifikaciju projekta na *GitHub* repozitorijumu, u direktorijumu */docs*. Specifikacija treba da sadrži sve neophodno detalje za izradu projektnog rešenja. Za više informacija, konsultovati se sa predmetnim asistentom.

Kontrolna tačka 2

Potrebno je implementirati podskup funkcionalnosti koji se dogovori sa asistentom na vježbama. Možete implementirati više od toga ukoliko tako želite **ali ne i manje!**

Odbrana projekta

Realizovati sve zahteve iz specifikacija. Demonstrirati rad čitavog sistema.

FAQ