

Specifikacija projektnog zadatka

Inženjerstvo softvera za Internet/Web of things

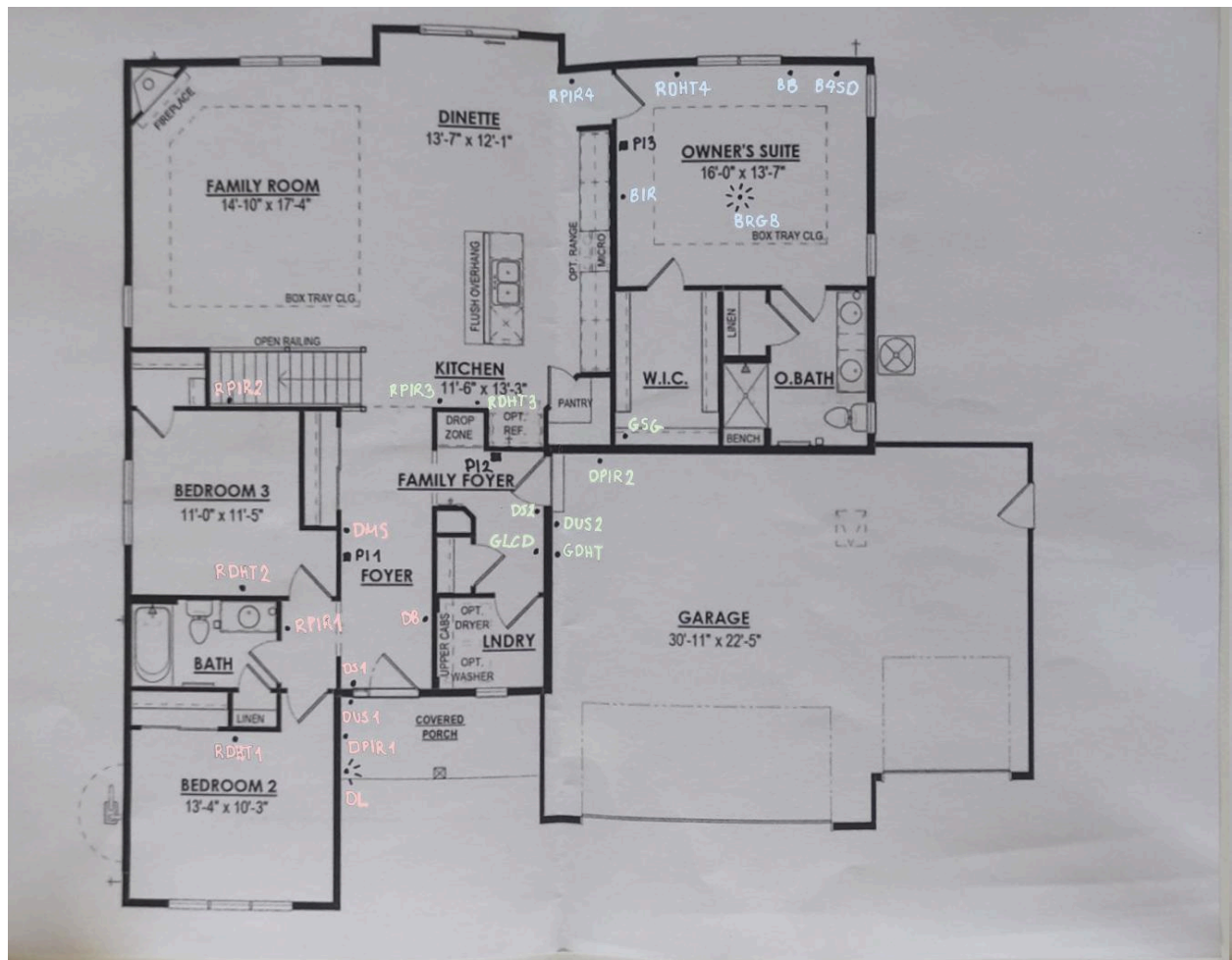
2023/2024

Timovi

Projekti zadatak se radi u **timovima od po dva students**. Timovi treba da se upišu u [tabelu](#), u tab koji odgovara njihovoj grupi vežbi.

Standardni projekat

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



Sistem pametne 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
	RPIR1, RPIR2	Room PIR
	RDH1, RDHT2	Room DHT
PI2	DS2	Door sensor (Button)
	DUS2	Door Ultrasonic Sensor
	DPIR2	Door Motion Sensor
	GDHT	Garage DHT
	GLCD	Garage LCD
	GSG	Gun Safe Gyro (Gyroscope)
	RPIR3	Room PIR
	RDHT3	Room DHT
PI3	RPIR4	Room PIR
	RDHT4	Room DHT
	BB	Bedroom buzzer
	B4SD	Bedroom 4 Digit 7 Segment Display
	BIR	Bedrom Infrared
	BRGB	Bedroom RGB

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.

Ulazne podatke sa svakog senzora potrebno je ispisati u konzoli.

Omogućiti upravljanje aktuatorima sa Raspberry PI uređaja kroz konzolnu aplikaciju.

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 konfigurabilnih informacija o uređaju, dokle god to ima smisla). Logiku je potrebno proširiti sa slanjem izmjerenih/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 djelovi koda koji se zaključavaju *mutex*-ima budu **minimalni**. Potrebno je implementirati server (preporučujemo *Flask* biblioteku ali može šta god želite od *Python* tehnologija) 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 u *Grafana* alatu. Potrebno je da svaki tip senzora ima svoj panel za prikaz.

Odbrana 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 da 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 RPIR1-4 senzora uključuje se ALARM.
6. Ukoliko GSG detektuje značajan pomeraj, uključiti ALARM.
7. Prikazati temperaturu i vlažnost vazduha sa GDHT na GLCD-u.
8. Prikazati trenutno vreme na B4SD-u.
9. Omogućiti podešavanja budilnika.
 - a. Putem Web aplikacije podesiti vreme kada će budilnik da se aktivira
 - b. Kada je budilnik aktivan, aktivirati BB dok se budilnik ne isključi. Takođe, za ovo vreme potrebno je da se prikaz na B4SD pali i gasi na 0.5 sekundi.
 - c. Budilnik se gasi putem Web aplikacije
 - d. Bonus zadatak: implementirati melodiju za buđenje
10. Omogućiti uključivanje, isključivanje i upravljanje bojama BRGB sijalice putem daljinskog upravljača i BIR senzora, kao i putem Web aplikacije.

Navedenu logiku realizovati na proizvoljan način.

Tokom odbrane projekata, za maksimalnu ocenu potrebno je povezati tražene senzore i/ili aktuatore 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.

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

1) Koja je razlika između "Room PIR" i "Door Motion Sensor"?

Isti su.

2) Da li je u redu da za PIR ispisujemo njegovo trenutno stanje, kao za DHT, umesto da koristimo event?

Koristite event, trebaće za dalju izradu u projektu.

3) Šta je Door Sensor?

Oponašamo ga preko dugmenceeta: dokle god je pritisnuto dugme, to znači da su vrata otključana a u suprotnom su zaključana.

4) Šta je Door Light?

Door Light je obična LED dioda.

5) Da li je Buzzer aktivan ili pasivan?

Aktivan.