

Univerza na Primorskem  
FAMNIT

# Projektni Seminar II

Funkcionalna Specifikacija

Avtorji: Goran Tubić, Nikola Marin

Mentor: dr. Tatjana Zrimec

# Kazalo

<b>1 Uvod</b>	<b>2</b>
1 Namen	2
2 Ciljna publika	2
3 Obseg projekta	2
<b>2 Splošni opis</b>	<b>3</b>
2. 1 Perspektiva izdelka	3
2. 2 Lastnosti produkta	4
2. 3 Uporabniški razredi in značilnosti	4
2. 4 Delovno okolje	4
2. 5 Uporabniška dokumentacija	4
2. 6 Predpostavke in odvisnosti	4
<b>3 Sistemske lastnosti</b>	<b>6</b>
3. 1 Nastavitev temperature	6
3. 1. 1 Opis in prioritete	6
3. 1. 2 Stimulativna/odzivna zaporedja	6
3. 2 Nastavitev urnika	6
3. 2. 1 Opis in prioritete	6
3. 2. 2 Stimulativna/odzivna zaporedja	6
3. 3 Nastavitev izjem	7
3. 2. 1 Opis in prioritete	7
3. 2. 2 Stimulativna/odzivna zaporedja	7
<b>4. Zunanja oblika</b>	<b>7</b>
4. 1 Uporabniški vmesnik	8
4. 2 Strojno okolje	8
4. 3 Programsko okolje	8
4. 4 Komunikacijsko okolje	8

# 1. Uvod

## 1.1 Namen

Dokument vsebuje funkcionalno specifikacijo za izdelavo termostata. Namen tega dokumenta je opis vseh funkcionalnosti, ki naj bi jih izdelan termostat omogočal. Celoten opis vseh funkcionalnosti pa je zaradi prevelikega obsega krajši od dejanskega izdelanega termostata.

## 1.2 Ciljna publika

Dokument funkcionalne specifikacije je namenjen:

- Razvijalcem - lahko služi kot oporna točka, saj morajo izpolnjevati zahteve, ki so opisane v tem dokumentu,
- Preizkuševalcem - lahko služi kot seznam lastnosti in funkcij, katere je potrebno preizkusiti za zagotavljanje najboljše uporabniške izkušnje,
- Uporabnikom - uporabniki lahko s pomočjo tega dokumenta spoznajo idejo projekta, lahko pa tudi služi kot priročnik za uporabo,
- Piscem dokumentacije - lahko je osnova za pisanje preostale projektne dokumentacije.

## 1.3 Obseg projekta

Celoten projekt je namenjen izdelavi sistema za uravnavanje temperature v različnih fizičnih prostorih. Izdelava projekta se razdeli v tri dele:

- strežniški del
- termostatski del
- uporabniški del

Strežniški del zajema zajem in shranjevanje podatkov ter skrb za komunikacijo med različnimi akterji tega sistema. Strežnik mora skrbeti za:

- varno komunikacijo med napravami,
- varno dodajanje novih termostatov,
- avtentikacijo vseh akterjev, ki komunicirajo z napravo,
- shranjevanje podatkov v skladu z zakoni o shranjevanju podatkov,
- kontroliranje in pošiljanje ukazov izbranim posameznim termostatom,
- opozarjanje uporabnika za morebitne izpade

Termostatski del zajema pregled temperature v njegovem okolju in komunikacijo s strežnikom. Tako mora skrbeti za:

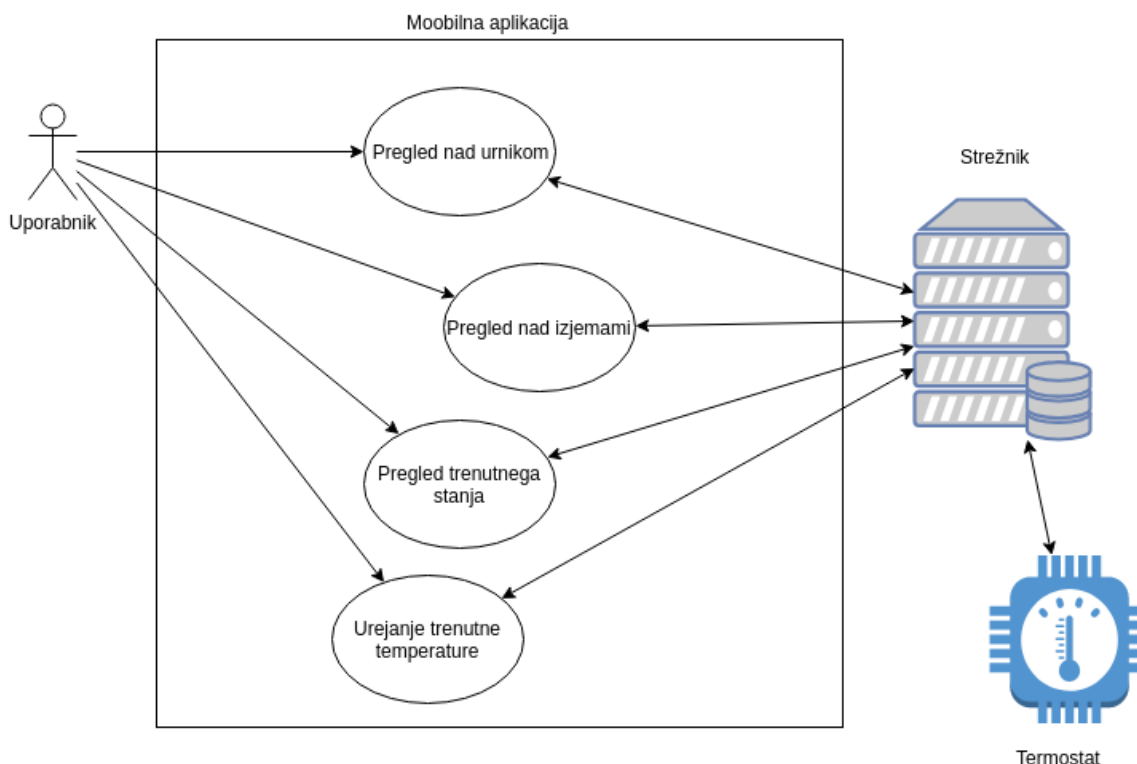
- učinkovito branje temperature v njegovem okolju,
- varno komunikacijo s strežnikom,
- korigiranje temperature glede na uporabnikove zahteve,
- prilagoditev temperature glede na zunanje spremembe,

- pošiljanje podatkov o očitkih narejenih preko lastnih senzorjev.

Uporabniški del zajema mobilno aplikacijo. Preko mobilne aplikacije uporabnik ima vpogled v trenutno temperaturo in upravljanje z njo. Tako mobilna aplikacija skrbi za:

- uporabnik lahko sporoči strežniku za njegove temperaturne občutke (toplo/hladno),
- uporabnik ima pregled nad urnikom termostata,
- uporabnik lahko nastavi izjeme (počitnice in podobno)
- uporabnik je obveščen ob izpadih termostata.

Diagram primera uporabe:



Slika 1: Diagram primerov uporabe (angl.: Use case diagram)

## 2 Splošni opis

### 2. 1 Perspektiva izdelka

Celoten sistem bo uporabljen predvsem od obstoječih uporabnikov termostata in seveda novih/bodočih strank pridobljenih v nekem časovnem obdobju. Aplikacija bo lahko uporabljena iz strani ene osebe, družin ali več različnih oseb.

## 2. 2 Lastnosti produkta

Uporabnik sistema zelo enostavno upravlja termostat preko mobilne naprave. Možnost dostopa naj bi bil tudi preko računalnika vendar je to stvar nadaljnjega možnega razvoja. Aplikacija bo narejena v angleškem jeziku, kar bo omogočalo uporabo tudi govornikom ne slovenskega jezika. Za pridobitev novih strank bo poskrbelo podjetje X.

## 2. 3 Uporabniški razredi in značilnosti

V celotnem sistemu lahko dodamo nove jeziki in razširimo krog uporabnikov z ljudmi, ki morebiti tudi angleško ne govorijo. Poznamo pa le skupine uporabnikov (nekaj ljudi, družine, veliko ljudi, ...). Vsi uporabniki pa imajo iste pravice. V nadaljnjem razvoju se lahko dodajo administratorske pravice posebnim uporabnikom, ki bi potem urejali pravice članom svoje skupine.

## 2. 4 Delovno okolje

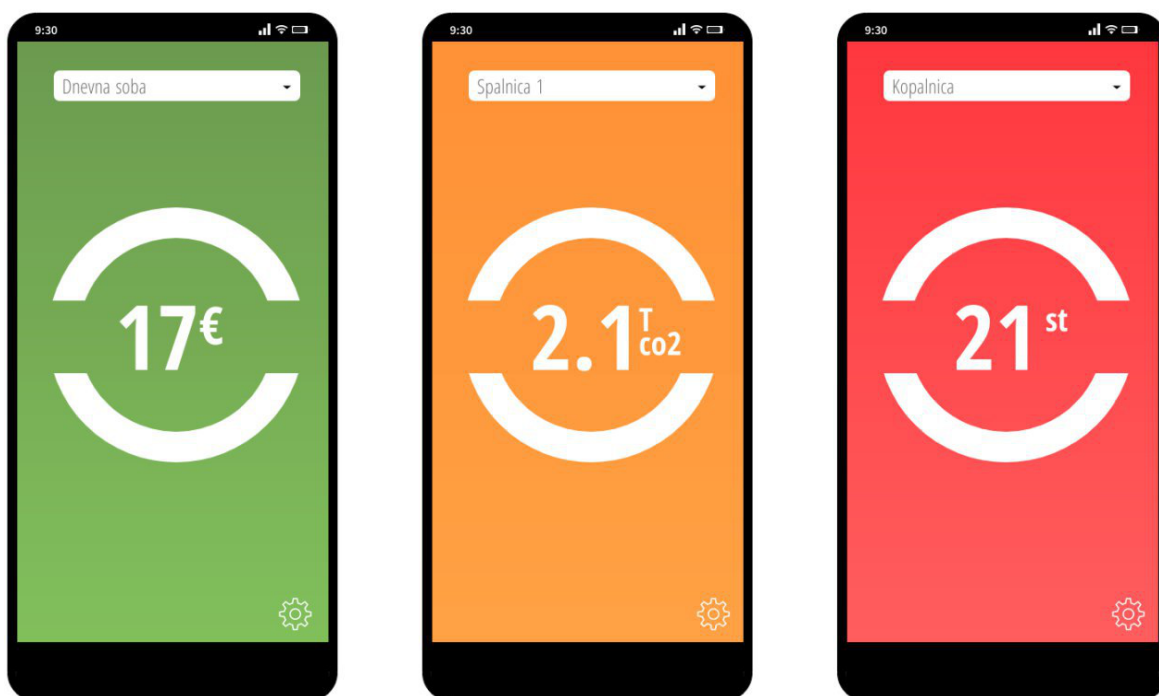
Delovno okolje je dokaj preprosto. Uporabnik enostavno uporablja mobilno aplikacijo za potrebe nastavitve temperature v svojem okolju.

## 2. 5 Uporabniška dokumentacija

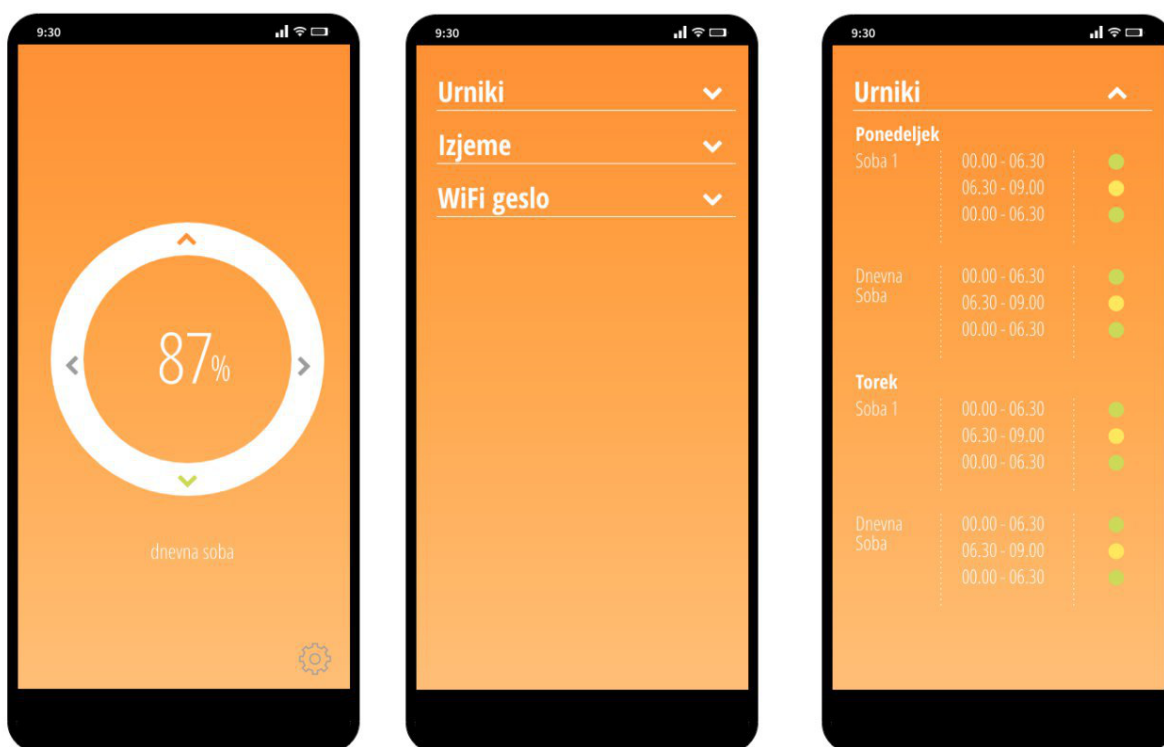
Glej dokument uporabniške dokumentacije.

## 2. 6 Predpostavke in odvisnosti

Za delovanje sistema uporabnik mora omogočiti internetno povezavo s termostatom. To lahko naredi preko WireLess tehnologije ali pa preko UTP kabla. Če uporabnik želi uravnavati temperaturo to lahko opravi preko termostata ali pa preko mobilne aplikacije. Torej uporabnik je odvisen od pametne naprave (skoraj da jo imajo vsi).



Slika 2: Predlagan izgled mobilne aplikacije za več sobe



Slika 3: Po nekaj iteracija Predlagan izgled mobilne aplikacije z vsemi podstranmi.

## 3 Sistemske lastnosti

### 3. 1 Nastavitev temperature

#### 3. 1. 1 Opis in prioritete

Nastavitev temperature je glavna točka celotnega sistema. Je zelo pomembna ker glede na njo porabnik uravnava temperaturo v prostoru.

Prioriteta: viska (10)

#### 3. 1. 2 Stimulativna/odzivna zaporedja

- uporabnik odpre mobilno aplikacijo
- uporabnik klikne na plus/minus za povečevanje/zmanjševanje trenutne temperature
- uporabnik se posodobi status termostata

### 3. 2 Nastavitev urnika

#### 3. 2. 1 Opis in prioritete

Nastavitev urnika je točka, ki je pomembna za uporabnika. Glede na njegov urnik lahko potem termostat bodisi vklapljamo ali izklapljamo. Uporabnik je npr. lahko v določenih urah odsoten, ni na lokaciji in lahko nastavi bodisi zniža gretje ali bodisi zniža hlajenje. Namen je da se uporablja čim manj električne energije. V naslednji iteraciji naj bi pametni algoritem to počel sam tako, da bi ugotovil lokacijo uporabnika preko mobilnega telefona.

Prioriteta: srednja (7)

#### 3. 2. 2 Stimulativna/odzivna zaporedja

- uporabnik odpre mobilno aplikacijo
- uporabnik izbere nastavitveni pogled
- uporabnik izbere zavihek urniku
- uporabniku se odpre celotedenski urnik
- uporabnik ureja urnik z izbiro ur in temperature okolja

## 3. 3 Nastavitev izjem

### 3. 2. 1 Opis in prioritete

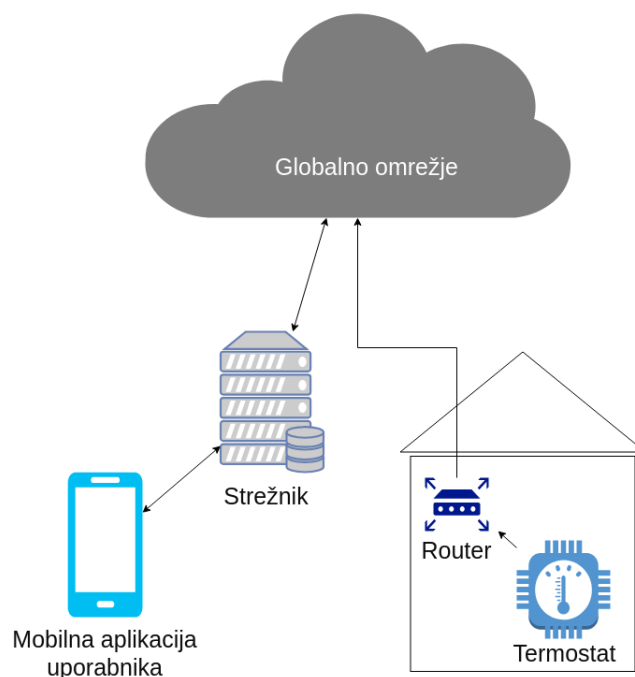
Uporabnik se lahko v določenem trenutku odloči, da bo izklopil termostat. Razlogi so lahko različni, od počitnic do nepotrebnega delovanja v določenih dnevih ali časovnih območjih. Uporabnik tako nastavi te datume in se termostat glede na njegovo odločitev iklopi.

Prioriteta: srednja (7)

### 3. 2. 2 Stimulativna/odzivna zaporedja

- uporabnik odpre mobilno aplikacijo
- uporabnik izbere nastavitveni pogled
- uporabnik izbere zavihek izjeme
- uporabniku se odprejo izjeme
- uporabnik ureja izjeme in/ali dodaja nove

## 4. Zunanja oblika



Slika 4: Predstavitev arhitekture celotnega sistema



## 4. 1 Uporabniški vmesnik

Uporabniški vmesnik se sestoji iz glavnega predstavitvenega okna. V temu oknu lahko uporabnik nastavlja temperaturo termostata. Ker ima lako uporabnik več termostatov ima prav tako možnost pregleda vseh teh in hkrati urejanje temperature na vsakemu posamezno. Naslednje pomembno okno je okno za nastavitve. Tukaj uporabniku prikažemo različne zavihke preko katerih lahko ureja določene nastavitve. V zavihku urniki uporabniku prikažemo predviden plan ogrevanja/hlajenja za določen dan v tednu. Sam pa lahko to še dodatno spreminja. V zavihku izjeme pa uporabniku pokažemo izjeme za izklop delovanja termostata v različnih primerih glede na potrebe uporabnika. Prav tako uporabnik lahko dodaja poljubno število novih izjem.

## 4. 2 Strojno okolje

Za uporabo celotnega sistema uporabnik potrebuje naš termostat (uporablja ESP8266 čip). Poleg tega potrebuje mobilni telefon za uporabo mobilne aplikacije in router, ki bo skrbel za povezavo z našim strežnikom.

## 4. 3 Programsko okolje

Celoten sistem deluje v različnih okoljih in se razvrsti v naslednji tabeli:

Fizično okolje	Programsko okolje
Termostat	C
Mobilna aplikacija	Ionic framework (Angular)
Strežnik	NodeJs (typescript)

Skrb za delovanje termostata bo napisana v C jeziku. Le ta bo skrbel za hranjenje avtentikacijskih podatkov in komunikacijo s strežnikom. Strežnik bo skrbel za posredovanje podatkov in bo postavljen v NodeJS okolju. Za pisanje kode bomo uporabljali typescript (super množica JS). Za mobilno aplikacijo pa bomo uporabili Ionic framework. Ta temelji na HTML tehnologijah (CSS, JS, HTML).

## 4. 4 Komunikacijsko okolje

Ker je varnost takšnih sistemov zelo pomembna bomo vso komunikacijo kriptirali z uporabo SSL certifikatov. Tako bo povezava potekala preko HTTPS protokola. Uporabnik bo pa preko push notifications direktno obveščen v mobilni napravah. V primeru izpadov bomo uporabnike obveščali še preko SMTP protokola za pošiljanje email-ov.