**Povzetek**

**Abstract**

1. **Uvod**
   1. ***Problematika***

Opis problema? Zakaj sem se lotil zadeve – olajšanje dela natakarjev, hitrejša postrežba, več časa lahko natakar posveti sami izdelavi pijače,… Nisem hotel narediti aplikacije, ki bi zamenjala natakarje, vendar bi jim pomagala pri delu.

* 1. ***Ideja***

Aplikacija s tremi pogledi – gost, natakar in kuhar. Vsak pogled omogoča različne funkcionalnosti.

Funkcionalnosti vsakega: gost – oddajanje naročil in urejanje neoddanih naročil, natakar – popoln nadzor nad vsemi naročili, kuhar – pregled in spremljanje že pripravljenih jedi.

1. **Struktura in razvoj aplikacije**

Kako sem si zamislil aplikacijo? Podatkovna baza->strežnik->odjemalec sestavljen iz dveh delov (kuhar, natakar/kuhar). RESTful vmesnik, ki servira podatke odjemalcu. Kaj je to RESTful? Slika za lažjo predstavo.

Omejitve: en uporabnik na mizo, neomajno število miz, neomejeno natakarjev (100), en glavni kuhar

* 1. ***Podatkovna baza***

Struktura podatkovne baze. Tabele in povezave. Kako je nastala nova tabela zaradi M:N povezave. Čemu je namenjena vsaka tabela in kako je sestavljana. Jezik MySQL, bolj podrobno opisan v 3. poglavju.

* 1. **Strežnik**

Predstavlja RESTful vmesnik, napisan v Python-Flask bolj podrobno opisano v 3. poglavju. Kako se strežnik povezuje s podatkovno bazo in podatke prikazuje v JSON formatu na spletnem vmesniku. Websocket rešitev, ki ob vsakem novem zapisu v bazo opozori odjemalca o novih podatkih v bazi. Težave z večkratnimi zahtevki na enkrat. Rešitev Flask-MySQL knjižnica.

* 1. ***Odjemalec***

Aplikacija napisana v Vue, Vuex in Vuetify katere tehnologije so bolje opisane v 3. poglavju. Zakaj sem ga uporabil, kako dobra je reaktivnost. Samo delovanje aplikacije, kako je sestavljena (gost, natakar/kuhar skupaj kateri se loči glede na prijavo). Branje in prikazovanje podatkov pridobljenih iz strežnika/baze (axios). Rešitev osveževanje podatkov s pomočjo Websocket-ov.

1. **Uporabljene tehnologije in programske opreme**
   1. ***Vue.js***
      1. Kaj je reaktivnost?
      2. Delovanje reaktivnost v Vue
      3. Knjižnice in dodatki
   2. ***Python/Flask***

Opis Python in uporaba knjižnice Flask ter njenih ostalih knjižnic.

* 1. ***MySQL***

Opis MySQL in zakaj ravno MySQL.

* 1. ***Ostala orodja***

Opis Viusal Studio in GitHub ter njuno uporabo.

1. **Delovanje aplikacije**
   1. ***Vmesnik za gosta***

Slikovni prikaz vmesnika in vse njegove funkcionalnosti oziroma pogledi.

* 1. ***Vmesnik za natakarja***

Slikovni prikaz vmesnika in vse njegove funkcionalnosti oziroma pogledi.

* 1. ***Vmesnik za kuharja***

Slikovni prikaz vmesnika in vse njegove funkcionalnosti oziroma pogledi.

* 1. ***Primeri uporabe***

Dejanski primeri:

1.) gost naroči hrano in pijačo

2.) gost želi oddati novo naročilo, stanje se spremeni samo natakarju o dodatnem naročilu pijače

3.) natakar ugotovi, da nima na voljo pijače, zato spremeni naročilo

4.) kuhar ne potrdi naročila, saj ugotovi, da nima na voljo vseh sestavin

5.) gost želi izbrisati pijačo in hrano katera mu je že bila servirana – ni mogoče

6.) način zaključevanje naročila s strani gosta

7.) zaključevanje naročila s strani natakarja

8.) naročanje hrane in pijače brez aplikacije – tipka »call waiter«

1. **Diskusija**
   1. ***Implementacija programa v realnosti***

Dve rešitvi za restavracije. Cenejša različica bi bile QR kode na mizah, katere bi gost skandiral z mobilnim telefon in bi bil preusmerjen v aplikacijo. Dražja različica, da ima vsaka miza svojo tablico. Prednosti in slabosti… Dodatno bi lahko bila tudi kombinacija tablic in QR kod.

Strežnik bi postavili za celotno Slovenijo, ki bi omogočal storitev vsem restavracija po Sloveniji. V primeru QR kod bi morala biti dodatna varnost, saj bi lahko prišlo do napadov izven restavracije. Rešitev bi bilo geslo, ki se spreminja za vsako naročilo – izda ga natakar.

* 1. ***Izboljšave***

Statistika za lastnika restavracije, ki bi poleg vseh podatkov lahko računala oceno nabave za prihodnji mesec. To bi bil svoj ločen uporabniški račun, ki bi omogočal tudi nastavitve v pogledu za gosta –lahko bi spreminjal številko mize.

Dodatno spremljanje hrane in pijače na strani kuharja/natakarja – katera pijača in hrana je že bila postrežena.

Brezstično plačevanje s kartico direktno na strani gosta.

Več kuharjev, da bi lahko vsak kuhar vedel kaj more delati.

Če bi aplikacija delovala na centralnem strežniku, bi lahko za vsako restavracijo omogočali tudi dostavo hrane z enakim pogledom, ki bi bil vedno dostopen na skupni spletni strani – npr. kot ehrana in wolt.

Vpeljava sistema v restavracije brez potreb po natakarju. Podobne želje so imeli v Lars&Sven.

* 1. ***Konkurenca***

McDonalds, ki rešitev uporablja že nekaj let. Prednosti in slabosti – naročile se lahko izvede elektronsko samo na vhodu, ni mogoče ponovnega hitrega naročila,…

Spletne aplikacije, ki ponujajo dostavo vseh restavracij po Sloveniji npr. ehrana in wolt. To bi bilo mogoče narediti tudi za mojo rešitev. Prednost bi bila, da bi mi lahko restavracijam ponujal to kot paket.

1. **Sklepne ugotovitve**