Prvi pogled vmesnika za gosta vsebuje napis za dobrodošlico, slika~\ref{Gost}, kateri bi lahko zamenjalo oglaševanja, predstavitev restavracije ali karkoli bi si potencialni kupec zaželel. V zgornjem desnem kotu se nahaja gumb call waiter za klic natakarja in števec skupne cene artiklov v nakupovalni košarici. Gumb call waiter je namenjen gostom, ki aplikacije ne želijo uporabljati ali v primeru kakršnihkoli težav. V spodnjem levem kotu se nahaja status in identifikacijska številka naročila. Meni na levi strani je pravzaprav seznam vseh vrst hrane in pijače, ki kažejo na podstrani z artiki, slika~\ref{Gost\_3}. Gostu meni predstavlja jedilnik s katerim v nakupovalno košarico dodaja in briše artikle ali spreminja njihovo količino. Nakupovalna košarica (ang. cart) je skupno mesto vseh artiklov potencialnih za naročilo, slika~\ref{Gost\_4}.

Naročilo se odda s klikom na place order, ki gosta preusmeri na prvo stran in izpiše pojavno sporočilo prikazano na sliki~\ref{Gost\_5}. Ko je naročilo oddano lahko gost ponovno dodaja artikle v košarico, vendar je že oddanim artiklom onemogočeno zmanjšati količino ali jih izbrisati. Naročilo je sprejeto ko ga natakar potrdi, s tem se spremeni statusa naročila in prikaže se pojavno sporočilo prikazano na sliki~\ref{Gost\_6}. V primeru, da natakar zavrne naročilo, ga mora gost pregledati in v primeru sprememb ponovno oddati. Tudi v tem primeru je gost obveščen s statusom in pojavnim sporočilom, slika~\ref{Gost\_8}. Naročilo se zaključi s klikom na gumb request receipt, ki odpre pojavno okno prikazano na sliki~\ref{Gost\_7}, kjer je gost izbere način plačila (gotovina, kartica). Novi gost lahko začne naročati, ko natakar potrdi plačilo oziroma natisne račun.

Opiši izgled artikla!!!

Opiši izgled artikla v košarici!

Koraki:

a.) Uporabnik doda željene artikle v košarico

b.) Uporabnik pokliče natakarja (CALL WAITER) -> naročilo se več v tem sistemu

Uporabnik odda naročilo

Kuhar potrdi/zavrne naročilo.

V primeru zavrnitve natakar obvesti gosta -> 4.)

Natakar potrdi/zavrne naročilo in o tem obvesti gosta

V primeru zavrnitve more gost ponovno oddati naročilo -> točka 1a) in 2.)

Kuhar zaključi s pripravo hrane in naročilo označi kot DONE

Natakar postreže naročilo ter ga označi kot SERVED

Uporabnik zahteva račun (gotovina/kartica)

Natakar natisne račun (INVOICE) in s tem zaključi naročilo

Prvi pogled vmesnika je enak tako za natakarja kot kuharja, saj gre za skupno aplikacijo kjer se pogledi razlikuje glede vlogo uporabnika, ki je določena v podatkovni bazi. Nismo naredili ločene aplikacije, saj ni bilo potrebe, namreč oba uporabnika imata zelo podobne funkcije. Slika~\ref{NatakarGost} prikazuje prijavno okno. Po prijavi natakarja ali kuharja se uporabniško ime izpiše v levem spodnjem kotu. Na zgornji levi strani se prikaže meni z dvema podstranema in odjavnim gumbom sing out. Prva podstran imenovana dashboard ali prva strani ob uspešni prijavi, prikazuje napis za hitrejšo razlikovanje med vlogami (slika~\ref{Dvapogleda}). Natakar ima poleg tega v desnem zgornjem kotu še števec čakajočih in zaključenih naročil.

Kuharju se v meniju food orders pojavijo vsa nezaključena naročila, ki vsebujejo hrano (slika~\ref{Kuhar\_4}). Vsako naročilo vsebuje naslednje podatke: identifikacijska številka naročila, čas oddaje naročila, status kuharja (Chef status) in številka mize. Kuhar more naročilo najprej pregledati z gumbom check details (slika~\ref{Kuhar\_3}) ter ga sprejeti ali zavrniti z gumbom confirm in unconfirm. Naročilo bi lahko zavrnil v primeru pomankanja sestavin. Ko kuhar zaključi s pripravo hrane o tem obvesti natakarja s klikom na gumb done, ki izbriše naročilo iz seznama. Ob vsaki izvedeni akciji se pri vsakem naročilu spremi status kuharja, ki je lahko: updated, confirmed, unconfirmed in done. Status updated je v primeru dodajanja artiklov k naročilu s strani gosta.

Natakarju se v meniju all orders pojavijo vsa nezaključena naročila. Vsako naročilo vsebuje naslednje podatke: identifikacijska številka naročila (Order number), čas oddaje naročila, status naročila (Order status), status kuharja (Chef status), način plačila (Payment) in številka mize (Table ID). Izvaja lahko popoln nadzor nad naročili, vendar ob vsaki storjeni akciji aplikacija obvesti gosta, (slike pojavnih sporočil iz prejšnjega poglavja). Novo naročilo mora natakar najprej potrditi ali zavrniti z gumbom confirm in unconfirm oziroma urediti z gumbom check details. Da lahko sprejme to odločitev mora počakati na kuharjevo potrditev/zavrnitev naročila o hrani. Ko je naročilo potrjeno s strani kuharja, natakar čaka na njegovo potrditev o pripravi hrane, da jo lahko postreže. Natakar to tudi primerno označi v naročilo z klikom na gumb served. V primeru, da gost zahteva račun, se natakrju izpiše način plačila v zadevi Payment. Natakar ob kliku na gumb Invoice zaključi naročilo.

Dve rešitvi za restavracije.

1. Cenejša različica bi bile QR kode na mizah, v katerih bi bilo zapisano ime restavracije in številka mize. To kodo bi gost skeniral z mobilnim telefon in bil preusmerjen v aplikacijo. Celoten sistem bi bil na javnem Internetu tudi za kuharje in natakarje. Potrebno bi bilo zagotoviti, da ne bi prihajalo do »fantomskih« naročil, kjer bi nepridipravi oddaljeno oddajali neveljavna naročila.
2. Dražja različica je, da bi imela vsaka miza svojo tablico. Sistem bi bil dostopen samo lokalno, znotraj restavracije, kar bi bilo iz vidika varnosti precej bolj varno. Restavracija poleg tablic potrebuje še lokalni spletni strežnik.

V prvem primeri bi bil strežnik postavljen za celotno Slovenijo, in 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.

Našo aplikacijo bi potencialnim kupcem ponudili v dveh različicah.

Prva različica, ki bi bila cenejša, bi predstavljala aplikacijo v obliki QR kod na vsaki mizi v restavraciji. V te kodi, bi bilo zapisano ime restavracije in številka mize. To kodo bi gost skeniral z mobilnim telefon in bil preusmerjen v aplikacijo. Celoten sistem bi bil na javnem Internetu tako za goste kot tudi natakarje in kuharje. Potrebno bi bilo zagotoviti, da ne bi prihajalo do »fantomskih« naročil, kjer bi nepridipravi oddaljeno oddajali neveljavna naročila. To bi lahko zagotovili na primer, da bi moral biti uporabnik prijavljen na lokalnem brezžičnem omrežju restavracije s čimer bi overili njegovo dejansko prisotnost v restavraciji. Seveda bi pri tem morali zagotovi še vrsto drugih varnostnih mehanizmov.

Druga oziroma dražja različica bi bila uporaba tablic za vsako mizo. Sistem bi bil dostopen samo lokalno, znotraj restavracije, kar bi bilo iz vidika varnosti precej bolj varno. Restavracija poleg tablic potrebuje še lokalni spletni strežnik.

Našo aplikacijo bi potencialnim kupcem ponudili v dveh različicah.

Prva različica, ki bi bila cenejša, bi predstavljala aplikacijo v obliki QR kod na vsaki mizi v restavraciji. V kodi bi bilo zapisano ime restavracije in številka mize. Kodo bi gost skeniral z mobilnim telefon in bil preusmerjen v aplikacijo. Celoten sistem bi bil na javnem Internetu tako za goste kot tudi natakarje in kuharje. Strežnik bi bil postavljen za celotno Slovenijo, in bi omogočal storitev vsem restavracija po Sloveniji. Potrebno bi bilo zagotoviti, da ne bi prihajalo do »fantomskih« naročil, kjer bi nepridipravi oddaljeno oddajali neveljavna naročila. To bi lahko zagotovili na primer, da bi moral biti uporabnik prijavljen preko lokalnega omrežja s čimer bi overili njegovo dejansko prisotnost v restavraciji. Seveda bi pri tem morali zagotovi še vrsto drugih varnostnih mehanizmov. Ena od rešitev bi bilo geslo, ki se spreminja za vsako naročilo in ga lahko izda samo natakar.

Druga oziroma dražja različica bi bila uporaba tablic za vsako mizo. Sistem bi bil dostopen samo lokalno, znotraj restavracije, kar bi bilo iz vidika varnosti precej bolj varno. Restavracija bi potrebovala poleg tablic potrebuje še lokalni spletni strežnik.