

Projektin nimi

Tiimi: OnnenOravat

Johdanto

Projektimme aiheena on lipunmyyntijärjestelmä TicketGuru lipputoimistoon. Järjestelmän on tarkoitus tulevaisuudessa toimia myös verkkokauppana, joten järjestelmä valmistetaan myös verkkokaupan edellytykset mielessä pitäen. Järjestelmään voidaan lisätä tapahtumia ja tapahtumiin erilaisia lipputyyppejä haluttu määrä. Asiakkaille myydään liput tulostamalla ne liikkeessä. Lippuihin tulostuu asiakkaan nimi ja uniikki lippukoodi. Järjestelmä varmistaa, ettei lippuja myydä liikaa tai esimerkiksi tapahtuman jälkeen. Myymättä jääneet liput tulostetaan tapahtumien ovelle myytäväksi. Lipunmyyjien työtä helpottaaksemme, lippujen etsinnän on oltava helppoa ja lippuja tulee voida ostaa ja tulostaa useampi kerrallaan. Lisäksi järjestelmä tarjoaa omistajalle liiketoiminnan kannalta tarvittavaa dataa, kuten tapahtuman myyntitapahtumat. **Projekti luodaan Spring Boot tekniikalla**

- Järjestelmän tarkoitus ja tiivis kuvaus siitä, mistä on kyse ja kenelle järjestelmä on tarkoitettu.
- Toteutus- ja toimintaympäristö lyhyesti:
 - Palvelinpuolen ratkaisut ja teknologiat (esim. palvelinteknologia, mikä tietokantajärjestelmä on käytössä)
 - Käyttöliittymäratkaisut ja teknologiat (esim. päätelaitteet: puhelin, täppäri, desktop)

Järjestelmän määrittely

Ohessa kuvattu järjestelmän käyttäjäryhmät ja heidän roolinsa yrityksessä. Käyttäjistä on luotu käyttäjätarinat, joiden pohjalta on luotu projektion työjono Trelloon sekä tarkempi toiminnallisten ominaisuuksien erittely.

Rooli 1: Lipunmyyjä

Rooli 2: Admin-käyttäjä / järjestelmänvalvoja

Rooli 3: Asiakas (Verkkokauppa)

Rooli 1: Lipunmyyjä (ja fyysinen asiakas)

Lippuvarauksen tekeminen asiakkaalle, siten että lippu tulostuu, varauksesta tulee uniikki tunnus, jonka voi merkitä ovella käytetyksi ja voi myös poistaa, jos asiakas palauttaa lipun. Varaus on nähtävissä datassa ja myyntiraportissa. Ennakkomyynnin loputtua loput liput tulostetaan ovella myytäväksi

Lipunmyyjänä haluan etsiä lipun, jota asiakas toivoo kaupassa, jotta voin myydä sen hänelle.

Lipunmyyjänä haluan tulostaa lipun asiakkaalle, jotta asiakas saa lipun ja pääsee keikalle.

Lipun myyjänä haluan nähdä, onko haluttua lipputyyppejä saatavilla, jotta tiedän, voinko myydä lipun asiakkaalle.

Lipun myyjänä haluan lisätä halutut tuotteet ostoskoriin, jotta voin laskuttaa kaikki kerralla asiakkaalta.

Lipun myyjänä haluan löytää asiakkaan ostaman lipun tietokannasta nimellä tai koodilla, jotta asiakkaan esimerkiksi hukatessa lipun voin tulostaa hänelle uuden.

Asiakkaana haluan valita lipputyypin, jotta saan mahdollisesti alennusta.

Lipunmyyjänä en halua myydä lippuja, joita ei ole enää saatavilla, sillä se olisi huonoa asiakaspalvelua

Lipunmyyjänä edustan omistajaa, enkä halua, että samalla lipulla pääsee useampi sisälle, sillä haluan maksimoida tuotot

Toiminnalliset vaatimukset

- Mahdollisuus valita lipputyyppe (Aikuinen, Lapsi, Eläkeläinen/Varusmies)
- Etsiä tietokannasta haluttu lippu
 - päivämäärällä? nimellä? tapahtumapaikka? paikkakunta? esiintyjän nimi
- Mahdollisuus valita tapahtuma, johon lippua myydään
 - siirtää ostoskoriin
 - Mahdollisuus ostaa monta lippua kerralla
- Mahdollisuus ottaa lippu pois käytöstä, mikäli asiakas palauttaa sen
 - lipun koodi deaktivoidaan palauttaessa ja yksi uusi lippu vapautetaan myyntiin? automaatio?
 - Samaa lippua ei voi käyttää useammin kuin yhden kerran
- Lippuja ei voi myydä enempää kuin määrän X
 - jos kpl määrä 0 -> ei onnistu
- Lippuja ei voi myydä ajan dd.mm.yyyy jälkeen
 - jos pvm x -> ei onnistu
- Myydyn lipun on oltava uniikki koodi tietokannassa
- Lipun uniikin koodin on pystyttävä skannata
- Myytyjen lippujen määrien (ja rahojen) on oltavissa nähtävissä datassa (ja oltava tulostettavissa)
- Sovellusta voi käyttää Windows & OSX käyttöjärjestelmillä
- Tietoturva
 - asiakkaan tietojenkäsittely, tarviiko lipunmyyjän nähdä kaikki asiakkaan tiedot? yms. gdpr jutut? salasanat..
 - muiden myyjien tiedot salassa
 - Sisäänkirjautuminen
 - Omat kirjautumistiedot
 - Henkilötiedot
 - Osoitetiedot

- Sovellus mahdollistaa usean samanaikaisen asiakkaan vierailun
- Sovellus mahdollistaa usean samanaikaisen myyntitapahtuman
- Ennakkomyynnin loputtua loput liput tulostetaan ovelle myytäviksi

*verkkokauppa

Ei-toiminnalliset vaatimukset (ei koodattavissa olevat vaatimukset)

- Sovelluksen on oltava asiakkaan saatavilla 24/7
- Sovellusta on mahdollista käyttää mobiilisti

Rooli 2 Admin-käyttäjä

Admin-käyttäjä ylläpitää TicketGuru-järjestelmää, Admin-käyttäjän on pystyttävä kirjautumaan järjestelmään sisään admin-tunnuksilla. Admin-käyttäjän on pystyttävä tarkastelemaan kaikkia tehtyjä varauksia. Hänen on pystyttävä poistamaan, muokkaamaan ja lisäämään tapahtumia ja kaikkia niiden tietoja. Ennakkomyynnin loputtua loput liput tulostetaan ovelle myytäviksi

Admin käyttäjänä haluan lisätä myyjiä, jotta lippujen myynti onnistuu.

Admin käyttäjänä haluan lisätä tapahtuman, jotta se saadaan myyntiin.

Admin käyttäjänä haluan muokata ja poistaa tapahtumia, jotta lipunmyyjillä on ajantasaiset tiedot.

Admin käyttäjänä haluan saada kaikki myymättä olevat liput tulostettua helposti, jotta ne saadaan ovelle myyntiin.

Toiminnalliset vaatimukset

- Mahdollisuus valita lipputyyppe (Aikuinen, Lapsi, Eläkeläinen/Varusmies)
- Mahdollisuus luoda tapahtuma
- Mahdollisuus valita tapahtuma, johon lippua myydään
- Mahdollisuus ostaa monta lippua kerralla
- Mahdollisuus ottaa lippu pois käytöstä, mikäli asiakas palauttaa sen
- Lippuja ei voi myydä enempää kuin määrän X
- Lippuja ei voi myydä ajan dd.mm.yyyy jälkeen
- Myydyn lipun on luotava uniikki koodi
- Lipun uniikin koodin on pystyttävä skannata
- Samaa lippua ei voi käyttää useammin kuin yhden kerran

- Myytyjen lippujen määrien(ja rahojen) on oltavissa nähtävissä datassa (ja oltava tulostettavissa)
- Tietoturva
- Sisäänkirjautuminen
- Kirjautumistiedot
 - Mahdollisuus nollata/vaihtaa myyjän salasanaa
- Sovellus mahdollistaa usean samanaikaisen asiakkaan vierailun
- Sovellus mahdollistaa usean samanaikaisen myyntitapahtuman
- Ennakkomyyntin loputtua loput liput tulostetaan ovella myytäviksi

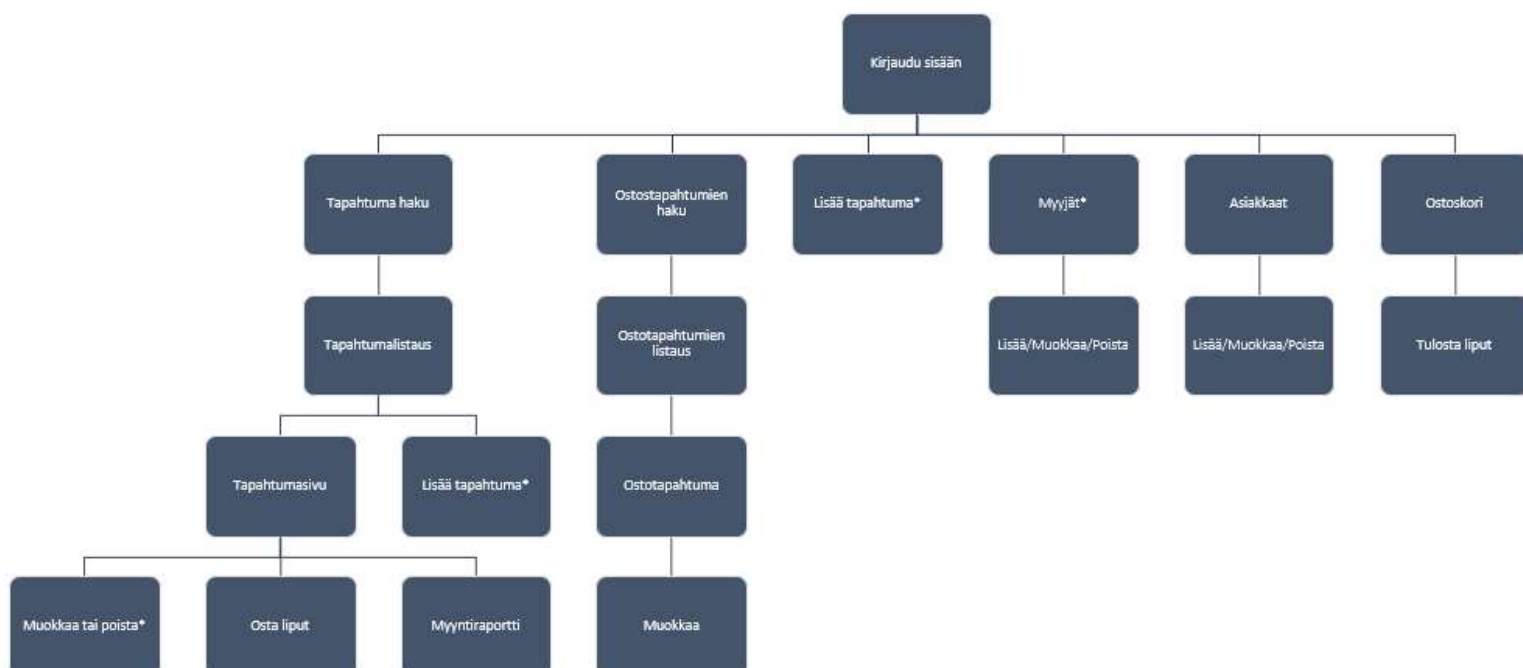
*verkkokauppa

Ei-toiminnalliset vaatimukset(ei koodattavissa olevat vaatimukset)

- Sovelluksen on oltava asiakkaan saatavilla 24/7
- Sovellusta on mahdollista käyttää mobiilisti

Käyttöliittymä

* admin-oikeuksilla



Kenttä	Tyyppi	Kuvaus
id	int PK	Tilin id
nimimerkki	varchar(30)	Tilin nimimerkki
avatar	int FK	Tilin avatar, viittaus avatar -tauluun
kayttaja	int FK	Viittaus käyttäjään käyttäjä -taulussa

Tekninen kuvaus

Teknisessä kuvauksessa esitetään järjestelmän toteutuksen suunnittelussa tehdyt tekniset ratkaisut, esim.

- Missä mikäkin järjestelmän komponentti ajetaan (tietokone, palvelinohjelma) ja komponenttien väliset yhteydet (vaikkapa tähän tyyliin: <https://security.ufl.edu/it-workers/risk-assessment/creating-an-information-systemdata-flow-diagram/>)
- Palvelintoteutuksen yleiskuvaus: teknologiat, deployment-ratkaisut yms.
- Keskeisten rajapintojen kuvaukset, esimerkit REST-rajapinta. Tarvittaessa voidaan rajapinnan käyttöä täsmentää UML-sekvenssikaavioilla.
- Toteutuksen yleisiä ratkaisuja, esim. turvallisuus.

Tämän lisäksi

- ohjelmakoodin tulee olla kommentoitua
- luokkien, metodien ja muuttujien tulee olla kuvaavasti nimettyjä ja noudattaa johdonmukaisia nimeämiskäytäntöjä
- ohjelmiston pitää olla organisoitu komponentteihin niin, että turhalta toistolta vältytään

Testaus

Tässä kohdin selvitetään, miten ohjelmiston oikea toiminta varmistetaan testaamalla projektin aikana: millaisia testauksia tehdään ja missä vaiheessa. Testauksen tarkemmat sisällöt ja testisuoritusten tulosten raportit kirjataan erillisiin dokumentteihin.

Tänne kirjataan myös lopuksi järjestelmän tunnetut ongelmat, joita ei ole korjattu.

Asennustiedot

Järjestelmän asennus on syytä dokumentoida kahdesta näkökulmasta:

- järjestelmän kehitysympäristö: miten järjestelmän kehitysympäristön saisi rakennettua johonkin toiseen koneeseen
- järjestelmän asentaminen tuotantoympäristöön: miten järjestelmän saisi asennettua johonkin uuteen ympäristöön.

Asennusohjeesta tulisi ainakin käydä ilmi, miten käytettävä tietokanta ja käyttäjät tulee ohjelmistoa asentaessa määritellä (käytettävä tietokanta, käyttäjätunnus, salasana, tietokannan luonti yms.).

Käynnistys- ja käyttöohje

Tyypillisesti tässä riittää kertoa ohjelman käynnistykseen tarvittava URL sekä mahdolliset kirjautumiseen tarvittavat tunnukset. Jos järjestelmän käynnistämiseen tai käyttöön liittyy joitain muita toimenpiteitä tai toimintajärjestykseen liittyviä asioita, nekin kerrotaan tässä yhteydessä.

Usko tai älä, tulet tarvitsemaan tätä itsekin, kun tauon jälkeen palaat järjestelmän pariin !