#1. Vaihe - Määrittely ja suunnittelu

## 1. Johdanto

Tämän projektin tavoitteena on suunnitella ja toteuttaa verkkosovellus Tmi Miika Mehtiölle. Sovellus on tarkoitettu julkaistavaksi vuoden 2025 loppupuolella, ja sen tavoitteena on toimia monipuolisena digitaalisen palvelun alustana. Verkkosovellus esittelee Miika Mehtiön tarjoamia palveluita, kuten cocktail- ja showbaarimestaripalveluita, koulutusta, konsultointia sekä bar catering -ratkaisuja. Sivusto mahdollistaa myös yhteydenoton, varaukset, käyttäjäprofiilien luonnin sekä materiaalien jakamisen ammattilaisille.

Tämä projektisuunnitelma kattaa kehitystyön ensimmäisen vaiheen: \*\*verkkosovelluksen määrittelyn ja suunnittelun\*\*. Tavoitteena on rakentaa tekninen perusta, johon kuuluvat muun muassa toiminnallinen rakenne, käyttöliittymän rakenne, käyttäjätarpeiden kartoittaminen, tietomalli sekä alustava backend-arkkitehtuuri. Visuaalinen viimeistely, sisällöntuotanto ja lopullinen julkaisu tapahtuvat myöhemmässä vaiheessa.

Projektin tekninen toteutus perustuu nykyaikaisiin web-teknologioihin. Kehitystyössä keskitytään modulaariseen rakenteeseen, saavutettavuuteen, käyttäjäystävällisyyteen ja jatkokehitettävyyteen.

## 2. Projektin tavoitteet

Projektin tavoitteena on toteuttaa Miika Mehtiön verkkosovelluksen tekninen perusta ja keskeisimmät toiminnallisuudet. Sovellus julkaistaan myöhemmin kesällä 2025, kotimaisten yritysten lomakauden päätyttyä. Visuaalinen viimeistely, sisällöntuotanto ja lopullinen julkaisu toteutetaan projektin myöhemmissä vaiheissa kurssin ulkopuolella.

Tämän kehitysvaiheen aikana (IT00AK35-3004 Web-kehittämisen jatkokurssi – Monimuoto) toteutetaan seuraavat toiminnot:

- Sivuston yleinen rakenne ja sivupohjien tekninen toteutus

- Navigaatio- ja siirtymätoiminnot eri näkymien välillä

- Käyttäjäprofiilin luonti ja kirjautuminen (autentikointi ja tietojen tallennus)

- Tiedustelulomake, jonka tiedot käsitellään palvelinpuolella

- Kalenteritoiminto ja varauslomake, joka estää päällekkäiset varaukset

- Portfolio-osio, jossa kuvat esitetään karusellina ja niitä voi kommentoida

Projektissa ei toteuteta muita toiminnallisuuksia kuin edellä mainitut. Mahdolliset prototyypissä olevat lisäominaisuudet ja -toiminnallisuudet toteutetaan ainoastaan, mikäli aikaa jää suunnitellun kehitystyön ulkopuolella.

## 3. Käytettävät teknologiat

\*\*Frontend:\*\*

- HTML5 – sivujen ja komponenttien rakenne

- CSS3 – käyttöliittymän tyylittely

- JavaScript – toiminnallisuuksien toteutus

- React – komponenttipohjainen käyttöliittymärakenne

\*\*Backend:\*\*

- Node.js – palvelinympäristö

- Express.js – API-reititys ja palvelinlogiikka

- PostgreSQL – tietokanta sovelluksen datalle (käyttäjät, varaukset, kommentit jne.)

- JWT (JSON Web Token) – käyttäjien kirjautumisen ja istunnon hallintaan

\*\*Muut työkalut ja ympäristöt:\*\*

- Azure – sovelluksen ja tietokannan pilvijulkaisu ja ylläpito

- Git ja GitHub – versionhallinta, tiimityöskentely ja dokumentointi

- Figma – käyttöliittymäprototyyppien suunnittelu ja testaaminen

## 4. Rajoitteet

Projektilla on ajalliset ja sisällölliset rajoitteet, jotka ohjaavat kehitystyötä tämän kurssin puitteissa. Sovelluksen lopullinen julkaisu, sisällöntuotanto ja visuaalinen viimeistely tapahtuvat myöhemmässä vaiheessa kesällä 2025, eikä niitä toteuteta tässä projektivaiheessa.

Tässä projektivaiheessa keskitytään ainoastaan määriteltyihin toiminnallisuuksiin, jotka on listattu kohdassa 2. Muita toimintoja ei suunnitella, toteuteta tai dokumentoida tässä vaiheessa. Mikäli projektin aikana jää ylimääräistä aikaa, voidaan harkita jonkin pienemmän lisätoiminnon toteuttamista, mutta nämä eivät kuulu projektisuunnitelman varsinaisiin tavoitteisiin.

Lisäksi projektissa ei käsitellä seuraavia osa-alueita:

- Maksuliikenteen tai verkkokaupan toteutusta

- Sisäänrakennettua koulutusmateriaaliympäristöä

- Video- tai stream-pohjaista sisältöä

- Tekoälyyn perustuvia toimintoja

- Graafista suunnittelua, animaatioita tai brändi-identiteettiä

Kaikki kehitystyö keskittyy tekniseen toteutukseen, rakenteeseen, toiminnallisuuksiin ja niiden dokumentointiin.

## 5. Käyttäjäpersoonat

---

### 🟦 1. Tapahtumajärjestäjä – Laura Lehtonen

\*\*Nimi:\*\* Laura Lehtonen

\*\*Ikä:\*\* 38

\*\*Ammatti:\*\* Ammattimainen tapahtumatuottaja

\*\*Tekninen osaaminen:\*\* Hyvä – käyttää säännöllisesti digitaalisia työkaluja tapahtumahallintaan

\*\*Laitekäyttö:\*\* Kannettava tietokone (Windows), iPhone

\*\*Tavoitteet:\*\*

- Löytää laadukkaita baaripalveluita tapahtumiin helposti

- Saada selkeät tarjoukset ja vertailla vaihtoehtoja

- Hallita yhteydenotto ja varaus keskitetysti yhdestä paikasta

\*\*Motivaatiot:\*\*

- Luoda onnistuneita ja vaikuttavia asiakastapahtumia

- Ylläpitää hyvää mainetta ammattijärjestäjänä

- Tehostaa omaa ajankäyttöään ja päätöksentekoa

\*\*Haasteet:\*\*

- Aikataulupaineet ja kiire tapahtumien lähestyessä

- Useiden palveluntarjoajien vertailun vaivalloisuus

- Tarve varmistaa palvelun luotettavuus ilman pitkiä selvityksiä

\*\*Käyttöskenaario:\*\*

Laura käyttää sovelluksen tiedustelulomaketta lähettääkseen tarjouspyynnön bar catering -palveluista tulevaa yritystapahtumaa varten. Hän odottaa saavansa vastauksen suoraan sähköpostiinsa.

---

### 🟩 2. Baarialan ammattilainen – Samu Räsänen

\*\*Nimi:\*\* Samu Räsänen

\*\*Ikä:\*\* 30

\*\*Ammatti:\*\* Baarimestari, cocktail-spesialisti

\*\*Tekninen osaaminen:\*\* Erinomainen – aktiivinen digipalveluiden käyttäjä

\*\*Laitekäyttö:\*\* Android, tablet, POS-järjestelmä

\*\*Tavoitteet:\*\*

- Kehittää omaa osaamistaan showbaarimestarina

- Löytää ammatillisesti hyödyllisiä sisältöjä ja kontakteja

- Saada näkyvyyttä ja verkostoitua muiden alan osaajien kanssa

\*\*Motivaatiot:\*\*

- Erottua muista baarialan toimijoista

- Päästä esiintymään tapahtumiin tai kilpailuihin

- Kehittää omaa uraansa kansainvälisesti

\*\*Haasteet:\*\*

- Koulutusten ja työn yhteensovittaminen

- Rajallinen tieto siitä, missä omaa osaamista voi kehittää

- Pienet budjetit ja henkilökohtaisten resurssien niukkuus

\*\*Käyttöskenaario:\*\*

Samu avaa sovelluksen, luo itselleen käyttäjätunnuksen ja kirjautuu ensimmäistä kertaa järjestelmään. Hän tarkistaa omat tietonsa ja tutkii profiilinäkymää.

---

### 🟨 3. Yksityishenkilö – Mari Vuorinen

\*\*Nimi:\*\* Mari Vuorinen

\*\*Ikä:\*\* 42

\*\*Ammatti:\*\* Opettaja

\*\*Tekninen osaaminen:\*\* Riittävä – käyttää peruspalveluita ja verkkokauppoja

\*\*Laitekäyttö:\*\* iPad, kotikäyttöön tarkoitettu kannettava tietokone

\*\*Tavoitteet:\*\*

- Löytää helposti persoonallinen ohjelmanumero juhliin

- Pystyä ottamaan yhteyttä ilman kirjautumista

- Ymmärtää palvelun sisältö selkeästi jo etusivulta

\*\*Motivaatiot:\*\*

- Järjestää läheisilleen unohtumattomia hetkiä

- Tarjota vierailleen jotain poikkeuksellista ja mieleenpainuvaa

- Hoitaa varausasiat nopeasti ja sujuvasti

\*\*Haasteet:\*\*

- Vähäinen aika palveluiden etsimiseen

- Epävarmuus siitä, mikä palvelu sopii juhlaan

- Tarve yksinkertaiselle, helposti lähestyttävälle käyttöliittymälle

\*\*Käyttöskenaario:\*\*

Mari selaa sivustoa käyttäen hampurilaisvalikkoa ja alanavigaatiopalkkia. Hän löytää kiinnostavan palvelun ja lähettää tiedustelun sähköpostitse sovelluksessa olevan yhteystietolinkin kautta.

---

### 🟥 4. Ravintoloitsija – Petri Kettunen

\*\*Nimi:\*\* Petri Kettunen

\*\*Ikä:\*\* 47

\*\*Ammatti:\*\* Cocktailbaarin ja ravintolan omistaja

\*\*Tekninen osaaminen:\*\* Hyvä – käyttää POS-järjestelmiä, CRM- ja koulutusjärjestelmiä

\*\*Laitekäyttö:\*\* iPhone, iPad, työasema

\*\*Tavoitteet:\*\*

- Parantaa henkilökunnan osaamista koulutuksilla

- Löytää nopeasti kouluttaja ilman pitkiä kilpailutuksia

- Varmistaa, että koulutuksen voi sovittaa ravintolan aikatauluihin

\*\*Motivaatiot:\*\*

- Nostaa asiakaskokemuksen tasoa

- Pitää ravintolan ajan tasalla alan trendeistä

- Vahvistaa henkilöstön ammattiylpeyttä ja sitoutumista

\*\*Haasteet:\*\*

- Henkilöstön kouluttaminen ilman liiketoiminnan häiriöitä

- Sopivien koulutusten löytäminen lyhyellä varoitusajalla

- Rajalliset resurssit pienessä yrityksessä

\*\*Käyttöskenaario:\*\*

Petri käyttää sovelluksen varaustoimintoa varatakseen yritykselleen koulutus- ja konsultointipalvelun tietylle ajankohdalle, joka sopii hänen henkilökunnalleen.

---

### 🟪 5. Media-alan ammattilainen – Jenni Manner

\*\*Nimi:\*\* Jenni Manner

\*\*Ikä:\*\* 35

\*\*Ammatti:\*\* TV-tuottaja

\*\*Tekninen osaaminen:\*\* Erittäin hyvä – tottunut käyttämään tuotanto- ja suunnittelutyökaluja

\*\*Laitekäyttö:\*\* MacBook Pro, iPhone

\*\*Tavoitteet:\*\*

- Löytää karismaattisia henkilöitä mediaan

- Saada nopea yleiskuva palveluntarjoajan tyylistä

- Tehdä alustava arvio ilman yhteydenottoa

\*\*Motivaatiot:\*\*

- Tuottaa näyttävää ja myyvää sisältöä

- Löytää uusia kykyjä ennen kilpailijoita

- Rakentaa ainutlaatuisia ohjelmakonsepteja

\*\*Haasteet:\*\*

- Sopivien ehdokkaiden löytäminen nopeasti

- Varmistaa, että persoona ja tyyli sopivat formaattiin

- Tiukat tuotantoaikataulut ja nopeat päätökset

\*\*Käyttöskenaario:\*\*

Jenni selaa portfoliokuvia ja referenssejä sovelluksessa. Hän löytää kiinnostavan kuvan ja kommentoi sitä suoraan järjestelmän kommentointitoiminnolla osana ensikontaktia.

## 6. Käyttötapaukset ja -tilanteet

Alla on määritelty viisi keskeistä käyttötapausta, jotka pohjautuvat käyttäjäpersoonien yksilöllisiin tavoitteisiin ja skenaarioihin. Jokainen käyttötapaus on kuvattu systemaattisesti seuraavassa muodossa:

- \*\*Käyttötapaus\*\*

- \*\*Käyttäjärooli\*\*

- \*\*Edellytykset\*\*

- \*\*Peruspolku (tapahtumakulku)\*\*

- \*\*Lopputulos\*\*

- \*\*Huomiot / rajaukset\*\*

---

### 🟦 Käyttötapaus 1: Tarjouspyynnön lähettäminen tiedustelulomakkeella

\*\*Käyttäjärooli:\*\* Tapahtumajärjestäjä (Laura)

\*\*Edellytykset:\*\*

- Käyttäjä on saapunut sovelluksen etusivulle

- Tiedustelulomake on toiminnassa ja saavutettavissa

\*\*Peruspolku:\*\*

1. Käyttäjä selaa sivustoa ja siirtyy lomakesivulle

2. Hän täyttää lomakkeeseen nimensä, sähköpostiosoitteensa ja viestinsä

3. Käyttäjä lähettää lomakkeen painikkeella

4. Lomakkeen tiedot toimitetaan palvelimelle

5. Palvelin lähettää tiedot eteenpäin sähköpostiin tai tallentaa ne tietokantaan

\*\*Lopputulos:\*\*

Käyttäjä saa vahvistusilmoituksen onnistuneesta lähetyksestä. Miika saa tarjouspyynnön tarkasteltavaksi.

\*\*Huomiot / rajaukset:\*\*

Lomake ei vaadi kirjautumista. Lomake voi sisältää CAPTCHA:n tai muun roskasuodatuksen.

---

### 🟩 Käyttötapaus 2: Käyttäjätunnuksen luominen ja kirjautuminen

\*\*Käyttäjärooli:\*\* Baarialan ammattilainen (Samu)

\*\*Edellytykset:\*\*

- Käyttäjä on saapunut sovelluksen kirjautumissivulle

- Rekisteröitymistoiminto ja tietokantayhteys toimivat

\*\*Peruspolku:\*\*

1. Käyttäjä valitsee "Luo tunnus" -toiminnon

2. Hän täyttää rekisteröitymislomakkeen (nimi, sähköposti, salasana)

3. Lomake validoidaan ja tiedot tallennetaan tietokantaan

4. Käyttäjälle luodaan tunnus ja JWT-tunniste (token) palautetaan

5. Käyttäjä kirjautuu sisään ja ohjataan profiilinäkymään

\*\*Lopputulos:\*\*

Käyttäjä on kirjautuneena järjestelmään ja voi käyttää profiilitoimintoja.

\*\*Huomiot / rajaukset:\*\*

Salasanat tallennetaan suojatusti (bcrypt). Rekisteröityminen voidaan tarvittaessa vahvistaa sähköpostitse.

---

### 🟨 Käyttötapaus 3: Navigointi ja tiedustelun lähettäminen sähköpostitse

\*\*Käyttäjärooli:\*\* Yksityishenkilö (Mari)

\*\*Edellytykset:\*\*

- Sivustolla on toimiva navigaatiorakenne

- Yhteystiedot ovat näkyvillä alanavigaatiossa

\*\*Peruspolku:\*\*

1. Käyttäjä avaa sivuston mobiililaitteella tai tietokoneella

2. Hän käyttää hampurilaisvalikkoa ja alanavigaatiota löytääkseen yhteystiedot

3. Yhteystiedoissa oleva sähköpostiosoite avaa käyttäjän sähköpostiohjelman

4. Käyttäjä kirjoittaa oman viestinsä ja lähettää sen sähköpostilla

\*\*Lopputulos:\*\*

Käyttäjän viesti toimitetaan Miikalle hänen sähköpostiosoitteeseensa.

\*\*Huomiot / rajaukset:\*\*

Ei vaadi kirjautumista. Sovellus ei käsittele tai tallenna viestiä teknisesti – sähköposti lähtee käyttäjän omasta sähköpostiohjelmasta.

---

### 🟥 Käyttötapaus 4: Varaus koulutus- ja konsultointipalvelulle

\*\*Käyttäjärooli:\*\* Ravintoloitsija (Petri)

\*\*Edellytykset:\*\*

- Käyttäjä on kirjautunut sisään

- Kalenteri- ja varaustoiminnot ovat käytettävissä

\*\*Peruspolku:\*\*

1. Käyttäjä siirtyy varausosioon

2. Hän valitsee kalenterista sopivan päivämäärän

3. Täyttää varauslomakkeen (palvelu, ajankohta, lisätiedot)

4. Järjestelmä tarkistaa saatavuuden ja tallentaa varauksen

5. Käyttäjä saa vahvistuksen onnistuneesta varauksesta

\*\*Lopputulos:\*\*

Varaus kirjautuu tietokantaan ja näkyy kalenterissa varattuna ajankohtana.

\*\*Huomiot / rajaukset:\*\*

Käyttäjä ei voi tehdä päällekkäisiä varauksia. Kalenteri tukee reaaliaikaista tilanäyttöä.

---

### 🟪 Käyttötapaus 5: Portfolion selaaminen ja kuvan kommentointi

\*\*Käyttäjärooli:\*\* Media-alan ammattilainen (Jenni)

\*\*Edellytykset:\*\*

- Portfolio on saavutettavissa ilman kirjautumista

- Kommentointiominaisuus on käytössä

\*\*Peruspolku:\*\*

1. Käyttäjä selaa portfolio-osuutta

2. Hän klikkaa kuvaa tarkastellakseen sitä

3. Kommentointikenttä avautuu kuvan alle

4. Käyttäjä kirjoittaa kommentin ja painaa "Lähetä"

5. Kommentti tallentuu ja näkyy julkisesti tai moderoidusti

\*\*Lopputulos:\*\*

Kuvan yhteyteen liitetään käyttäjän kommentti, joka voi toimia yhteydenoton ensiaskeleena.

\*\*Huomiot / rajaukset:\*\*

Kommentointi voidaan rajata vain rekisteröityneille käyttäjille. Kommentteja voidaan moderoida.

## 7. Käyttöliittymän suunnittelu ja prototyypit

Verkkosovelluksen käyttöliittymä on suunniteltu visuaaliseksi, yksinkertaiseksi ja saavutettavaksi. Suunnittelussa pyritään tukemaan mahdollisimman sujuvaa ja intuitiivista käyttäjäkokemusta, riippumatta käyttäjän laitteesta tai teknisestä osaamisesta.

### Prototyyppi

Käyttöliittymän prototyyppi on toteutettu \*\*Figma-suunnittelutyökalulla\*\*. Prototyyppi on \*\*keskeneräinen\*\* ja \*\*jatkuvan kehityksen alla\*\*, ja sen sisältö kehittyy rinnakkain teknisen toteutuksen kanssa. Se sisältää muun muassa seuraavat näkymät:

- Etusivu

- Portfolio ja kuvien kommentointinäkymä

- Tiedustelulomake

- Kalenterinäkymä ja varaustoiminto

Prototyypissä korostuvat erityisesti:

- Selkeä \*\*navigaatiorakenne\*\*: yläpalkki ja mobiililaitteilla hampurilaisvalikko

- Käyttäjäystävällinen \*\*lomakerakenne\*\*, jossa on riittävän isot kentät ja ohjeistukset

- \*\*Toimintopainikkeet\*\* (CTA) sijoiteltu loogisesti (esim. “Lähetä tarjouspyyntö”, “Varaa aika”, “Kommentoi kuvaa”)

- Mahdollisimman \*\*selkeä hierarkia ja visuaalinen kontrasti\*\*

\*\*Huomio:\*\* Prototyyppi ei vielä toistaiseksi ole responsiivinen. Prototyyppiä tulee tarkastella \*\*Figma-näkymällä "iPhone 16 Pro Max"\*\*, jossa kaikki elementit ja asettelut on suunniteltu toimimaan oikein.

### Prototyypin käyttö kehityksessä

Prototyyppi toimii projektissa \*\*yksin työskentelevän kehittäjän työkaluna\*\* visuaalisen suunnittelun ohjaamiseen ja palautteen keräämiseen. Sitä käytetään erityisesti \*\*sidosryhmän – Miika Mehtiön – kanssa tapahtuvaan esittelyyn ja kehitysehdotusten vastaanottamiseen\*\*. Miika antaa palautetta visuaalisesta ilmeestä, toiminnallisuuksista ja rakenteesta, ja ohjaa näin verkkosovelluksen muotoa lopullista käyttöönottoa varten.

\*\*Linkki Figman protoon:\*\* <https://www.figma.com/proto/E75SE748NU9vYsLa2Rzipd/Project-Miika-Mehti%C3%B6-Tmi?page-id=0%3A1&node-id=433-4397&viewport=-4230%2C18019%2C1.46&t=V4RygzNmPYu7gMua-1&scaling=scale-down&content-scaling=fixed&starting-point-node-id=433%3A4397>  
\*\*Salasana:\*\* VilleOpe!

## 8. Tietoarkkitehtuuri ja tekninen suunnittelu

Verkkosovellus toteutetaan modulaarisesti erottaen käyttöliittymä (frontend), palvelinlogiikka (backend) ja tietokantakerros (data persistence). Arkkitehtuuri tukee skaalautuvuutta, laajennettavuutta ja selkeää koodin hallintaa.

---

### Yleiskuva järjestelmästä

Sovellus koostuu kolmesta kerroksesta:

1. \*\*Frontend (React)\*\*

2. \*\*Backend (Node.js + Express)\*\*

3. \*\*Tietokanta (PostgreSQL)\*\*

Tiedonsiirto tapahtuu RESTful API -rajapinnan kautta. Lomakkeet, kirjautuminen, kommentointi ja varaukset lähettävät pyynnöt backendille, joka käsittelee ne ja kommunikoi tietokannan kanssa.

---

### Käyttötapauskohtainen tekninen toteutus

#### 🟦 1. Tiedustelulomakkeen lähetys

- \*\*Frontend:\*\* Lomake toteutetaan Reactilla, sisältää kenttävalidoinnit.

- \*\*Backend:\*\* POST /api/inquiry vastaanottaa lomakkeen ja lähettää tiedot eteenpäin (sähköposti tai tietokanta).

- Ei vaadi kirjautumista.

- Lomake voidaan suojata CAPTCHAlla ja sisältää palautteen käyttäjälle (onnistuminen/virhe).

#### 🟩 2. Käyttäjätunnuksen luominen ja kirjautuminen

- \*\*Frontend:\*\* Rekisteröinti- ja kirjautumislomakkeet toteutetaan Reactilla.

- \*\*Backend:\*\*

- POST /api/register – tallentaa käyttäjän bcrypt-hashatulla salasanalla.

- POST /api/login – tarkistaa tunnuksen, palauttaa JWT-tokenin.

- Tiedot tallennetaan PostgreSQL:iin tauluun `users`.

- JWT tallennetaan selaimen localStorageen ja käytetään jatkossa API-kutsuissa.

#### 🟨 3. Tiedustelu sähköpostilinkin kautta

- \*\*Frontend:\*\* Alanavigaatiossa on näkyvissä mailto-linkki.

- Klikkaus avaa käyttäjän oman sähköpostiohjelman → ei siirry backendille.

- Sovellus ei käsittele viestiä teknisesti.

#### 🟥 4. Kalenterin käyttö ja varauksen tekeminen

- \*\*Frontend:\*\* Kalenterikomponentti näyttää varattavat ajat.

- Käyttäjä valitsee ajan ja täyttää lomakkeen.

- \*\*Backend:\*\*

- GET /api/availability – tarkistaa vapaat ajat

- POST /api/bookings – luo uuden varauksen

- PostgreSQL: `bookings`-taulu (id, user\_id, date, service, status)

- Varaustoiminto estää tuplabuukkaukset vertaamalla ajan saatavuutta ennen tallennusta.

#### 🟪 5. Portfolio ja kommentointi

- \*\*Frontend:\*\* Kuvakaruselli ja lomake kommentin lisäämiseksi.

- \*\*Backend:\*\*

- GET /api/images – hakee kuvat ja kommentit

- POST /api/comments – lisää kommentin kuvaan

- PostgreSQL: `comments`-taulu (id, image\_id, user\_id, content, timestamp)

- Kommentointi voi vaatia kirjautumisen.

- Tarvittaessa: moderointilogiikka (kommentti hidden by default, admin hyväksyy)

---

### Teknologiat, kirjastot ja rakenteet

| Kerros | Teknologiat / työkalut |

|---------------|-------------------------------------------------------|

| Frontend | React, HTML, CSS, JavaScript (ES6+), React Hook Form |

| Backend | Node.js, Express, JWT, bcrypt, dotenv, cors |

| Tietokanta | PostgreSQL (Azure-hostattu tai muu pilvi-instanssi) |

| Versionhallinta | Git, GitHub |

| Muut | Figma (UI-protot), mahdollisesti Axios tai fetch |

---

### Esimerkki tiedonkulusta: Kommentin lisääminen

1. Käyttäjä kirjoittaa kommentin kuvaan → painaa "Lähetä"

2. React-komponentti lähettää POST-pyynnön backendille

3. Backend validoi tiedot ja lisää rivin `comments`-tauluun

4. Palauttaa uuden kommentin frontendille → näytetään käyttöliittymässä

---

### Rakenne ja laajennettavuus

- Sovellus tukee jatkossa uusia toiminnallisuuksia (esim. koulutusmoduulit, maksuliikenne) ilman rakenteen muuttamista.

- Tietokanta normalisoitu ja suunniteltu roolipohjaista käyttöä varten.

- Komponentit ovat uudelleenkäytettäviä ja frontendissä jaoteltu selkeästi (views, forms, nav, etc).

## 9. Projektinhallinta ja käyttäjätestaus

### Projektinhallinta

Projektin toteutuksesta vastaa yksi henkilö, joka toimii sekä suunnittelijana että kehittäjänä. Työn etenemistä ohjaa ja tukee selkeä vaiheistus sekä jatkuva dokumentointi.

\*\*Projektinhallinnan periaatteet:\*\*

- Projektia hallitaan \*\*iteratiivisesti\*\*: jokainen uusi toiminnallisuus suunnitellaan, toteutetaan ja testataan ennen seuraavaan siirtymistä.

- Kehitystyötä dokumentoidaan vaiheittain ja selkeästi: jokaiselle osalle on oma kuvauksensa projektidokumentaatiossa.

- \*\*Tuntikirjanpitoa\*\* pidetään erillisessä dokumentissa, joka liitetään osaksi palautusta.

- \*\*GitHub\*\* toimii versionhallintana sekä tehtävien ja muutosten seurannassa (committien viestit kuvaavat työn etenemistä).

\*\*Työvälineet:\*\*

- Git & GitHub – versiointi, kehityshistorian hallinta

- VS Code – koodieditori

- Figma – prototyypit ja käyttöliittymän suunnittelu

- Microsoft Word / Markdown – dokumentointi

\*\*Työn vaiheistus:\*\*

1. Määrittely ja suunnittelu (tämä dokumentti)

2. Tekninen toteutus vaiheittain (frontend → backend → integraatiot)

3. Käyttöliittymän ja toiminnallisuuksien testaus

4. Dokumentointi ja palautus

---

### Käyttäjätestaus

Käyttäjätestaus suoritetaan ennen lopullista palautusta, keskittyen käyttöliittymän toimivuuteen, ymmärrettävyyteen ja teknisiin toiminnallisuuksiin.

\*\*Testauksen tavoitteet:\*\*

- Varmistaa, että sovellus on looginen ja selkeä käyttäjän näkökulmasta

- Löytää käyttöliittymävirheitä tai epäselvyyksiä (esim. lomakkeiden logiikka)

- Testata kalenterin ja varaustoimintojen teknistä luotettavuutta

- Saada palautetta sidosryhmältä (Miika Mehtiö)

\*\*Testaajat:\*\*

- \*\*Sidosryhmän edustaja (Miika Mehtiö)\*\* toimii ensisijaisena testikäyttäjänä

- Lisäksi vähintään yksi ulkopuolinen testaaja (esim. ystävä tai kollega) käyttöliittymän käytettävyyden arviointiin

\*\*Testauksen menetelmät:\*\*

- \*\*Käytettävyystestit\*\*: käyttäjä suorittaa tehtäviä (esim. “lähetä lomake”, “kommentoi kuva”)

- \*\*Havaintopohjainen palaute\*\*: kirjataan mahdolliset ongelmat tai epäselvyydet

- \*\*Vapaa palaute\*\*: sidosryhmän suorat kommentit ja kehitysehdotukset

\*\*Testauksen dokumentointi:\*\*

- Testit kirjataan muistiin, ja niistä laaditaan lyhyt yhteenveto

- Olennaiset havainnot dokumentoidaan ja mahdollisesti korjataan ennen palautusta

- \*\*Yksityiskohtainen käyttäjätestaussuunnitelma laaditaan myöhemmin, lähempänä varsinaista testausta\*\*

## 10. Projektin vaiheiden dokumentointi

Projektissa panostetaan vaiheiden selkeään ja ajantasalla pysyvään dokumentointiin. Dokumentaatio toimii sekä projektin sisäisenä työvälineenä että kurssin palautusmateriaalina. Kaikki dokumentaatio säilytetään projektin \*\*GitHub-repositoriossa\*\*, joka toimii sekä versionhallintana että julkaisualustana.

### Tuntikirjanpito

- Projektista pidetään \*\*yksityiskohtaista tuntikirjanpitoa\*\*, joka kirjataan erilliseen dokumenttiin.

- Kirjanpidossa seurataan käytetty aika per toiminto/kehitysvaihe (esim. suunnittelu, frontend, backend, testaus, dokumentointi).

- Tuntikirjanpito toimitetaan osana lopullista palautusta, liitteenä tai linkitettynä.

### Dokumentaation rakenne

Projektin dokumentaatio koostuu seuraavista osista:

- Projektisuunnitelma (tämä tiedosto)

- Prototyyppi (Figma-linkki)

- Käyttäjätestauksen palaute ja yhteenveto

- Tuntikirjanpito (erillinen tiedosto)

- \*\*Versionhallintadokumentti\*\*, jossa kuvataan yksityiskohtaisesti mitä työtä on tehty ja mitä muutoksia on tehty aina, kun GitHubiin tehdään commit (esim. commit-viestit, ominaisuuspäivitykset, refaktoroinnit)

### Dokumentointitapa

- Dokumentaatio kirjoitetaan \*\*Markdown-muodossa\*\*, josta voidaan tarvittaessa konvertoida PDF/Word-versiot.

- Jokaiselle kehitysvaiheelle tai toiminnolle on oma osionsa: mitä tehtiin, miksi tehtiin, miten ratkaistiin ongelmat.

- Muutokset ja päätökset (esim. toiminnallisuuksien lisäykset/poistot) kirjataan näkyvästi ja perustellen.

- Dokumentaatio toimii samalla \*\*kehittyvänä työkaluna\*\*, ei pelkästään lopullisena raporttina.

- \*\*Kaikki dokumentaatio säilytetään GitHub-repositoriossa\*\*, versionhallinnan piirissä.