Flight Price Comparison Päättödemo

(Jani, Jenni, Miikka, Erno)

December 17, 2024

Projektin perusidea - Lentojen hintavertailu - Jani

- ► TOIVE: Käyttäjä kykenee löytämään itselleen mieluisia lentoja, vertailemaan niitä ja tallentamaan tiedot toivelistalle
- Syötteet
 - Lähtö- ja saapumislentokentät
 - Lähtöpäivä- ja aika näytetään sen jälkeiset lennot
 - Tarkempi haku: Aikuisten määrä, suoranlennon toive, maksimihinta sekä toivottu lentoluokka
- Tuloslistan järjestely ja filtteröinti
- Yksittäisen lennon yksityiskohtaiset tiedot
- ► Kahden lennon vertailu "Compare" napin avulla
- Sisäänkirjautuminen
 - Suosikkilentokentän tallentaminen
 - ► Toivelista
 - Omien tietojen muuttaminen
- Laajennettavuus: Muut liikennevaihtoehdot, kohteen hotellit, nähtävyydet, sää



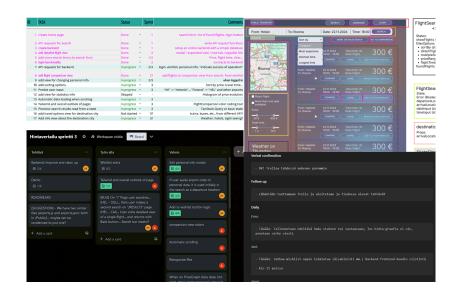
Käytetyt teknologiat - Jani

- Dev-osina: NPM, Vite, Visual Studio Code
- ► Versionhallinta: Git ja GitLab
- Ohjelmointikieli: JavaScript (React, React Router)
- ► Tyylittely: Tailwind
- APIt: Amadeus (lennot ja hintahistoria), Weather API (kohdekaupungin sää)
- Oma Backend: Node.js, Express, Render-hostauspalvelu
- ▶ Backendin testaus: Postman
- Toimintamallina Scrum

Vastuualueet - Jenni

- Erno: Amadeuksen API + React (yksittäisen lennon tiedot) + dokumentointi
- ▶ Jani: Back end (Front endissä osat, jotka keskittyivät Back endin kanssa kommunikointiin) + tekninen tieto + formatointi
- Jenni: React (lentojen vertailu, sivun käytettävyys) + Tailwind
- ▶ Miikka: React (peruskomponentit) + dokumentointi

Dokumentaatio - Jenni



Toiminnallisuudet (DEMO) - Miikka

- ► #1 Haku (TYO MEL):
 - From... To... Syötteen ehdotukset
 - Submit Vaatii syötteet toimiakseen
 - ► Haettujen lentojen listan rullaus
 - ► Filtterit/Order By/Hintagraafi
 - Yksittäinen lento tarkasteluun
 - Compare nappi
- Sisäänkirjautuminen
 - ► Virheelliset syötteet (User 1)
 - Kirjaudutaan ulos
 - Rekisteröinti
 - Henkilötietojen esittely/asetus + tyhjä-toivelista kotipaikaksi
 HFI
- ► #2 Haku (HEL NYC):
 - Advanced Search esittely, ja asetus (hinta ja luokka?)
 - Näytetään rajauksen toimivan
 - Yksi lento Vertailuun, näytetään lentojen erot
 - Yksi lento wishlistiin ja sen esittely + saman lennon uudelleen lisääminen (TOAST viesti)

<ロ > < 部 > < き > くき > き の < で

Haasteet - Erno

- Routetus ja tiedon lukeminen useasta paikasta (URL sekä hakupalkki)...
 - "Single source of truth"
 - Kahden pään voimin ratkesi useimmiten
- Yksittäisen lennon jakaminen URL:n avulla
 - Amadeus ei anna yksittäiselle lennolle uniikkia ID:tä lentoja ei voi jälkeenpäin erottaa toisistaan. Kuitenkin jos URLiin otettaisiin paljon tietoja mukaan (myös segmentit!) niin URL olisi käytännössä uniikki
 - Ongelma: lentotiedot voivat muuttua, joten parempaa olisi luoda oma database → "Share" napin jälkeen lento tallentuisi sinne
 - Ylittää projektin scopen → Harkittua karsintaa!
- ▶ Render ei tallenna dataa pysyvästi ei voitu tehdä omaa kunnollista tietokantaa

Onnistumiset - Erno

- Yleinen aikataulutus sopivasti tekemistä aina, ja yleinen skaala ei ollut ylilyöty
- Kattava kirjanpito
- ► Kaikille omia suuntautumisia, mutta kommunikaatio toimi silti
- Vahvaa jälkeä oman Back endin tuottamisessa
- Projektin päätavoitteet saavutettiin

Tärkeimmät opit - Erno

- Sprint Backlog hyvä pilkkoa, jotta tekemistä löytyy aina
- Dokumentoinnin merkitys (kuvittele tilanne, että katsot työtä muutaman kuukauden jälkeen)
- ▶ Merge requesteja tällaisessa projektissa 1-2 päivässä per hlö
- Ruudunjakoa kannattaa käyttää rohkeasti
- ► Monipuolista Git-soveltamista

Kiitos!