Lainatehtailijat

Loppuraportti

Web-ohjelmointi

INTIM22A6

Kevät 2024

Kostamo Teemu, Pernu Heikki

A picture containing text, font, screenshot, black

Description automatically generated



INTIM22A6 Tiivistelmä

Tekijä Teemu Kostamo, Heikki Pernu Vuosi 2024

Työn nimi Web-ohjelmointi / Loppuraportti

Ohjaaja Petri Kuittinen, Lea Mustonen

Tässä loppuraportissa kuvaamme Web-ohjelmointikurssiin liittyvän Django web-sovellusta. Rakensimme lainausjärjestelmän, jossa käyttäjät voivat varata ja palauttaa erilaisia työkaluja.

Avainsanat Django, web-sovellus, lainausjärjestelmä

Sivut XX sivua



INTIM22A6 Abstract

Author Teemu Kostamo, Heikki Pernu Year 2024

Subject Web-ohjelmointi / Final Report

Supervisors Petri Kuittinen, Lea Mustonen

This is a final report for a web programming course, in which we describe our Django based web application for a fictional residents' association that allows users to borrow and return tools.

Keywords Django, web application, reservation system.

Pages XX pages

Sisällys

[1 Johdanto 1](#_Toc164943735)

[2 Teknologia 2](#_Toc164943736)

[3 Projekti 2](#_Toc164943737)

[3.1 Perustoiminnallisuudet 3](#_Toc164943738)

[3.2 Tietokanta 6](#_Toc164943739)

[4 Johtopäätökset 7](#_Toc164943740)

[Lähteet 9](#_Toc164943741)

Kuvat, taulukot ja kaavat

[Kuva 1 Kirjautumissivu 3](#_Toc164943742)

[Kuva 2 Rekisteröitymissivu 4](#_Toc164943743)

[Kuva 3 Näkymä palvelussa olevista välineistä 5](#_Toc164943744)

[Kuva 4 Tuotteiden hallinta Admin-liittymän kautta 6](#_Toc164943745)

[Kuva 5 Models.py 7](#_Toc164943746)

# Johdanto

Tässä raportissa kuvataan Web-ohjelmointi-kurssin osana olevaa Django-projektin toteutusta. Olimme valinneet aiheeksi lainausjärjestelmän, jonka käyttäjä on kuvitteellinen asukasyhdistys. Valitsemassamme lainausjärjestelmässämme käyttäjä voi selata, varata ja palauttaa työkaluja. Lisäksi käyttäjä pystyy myös selaamaan vanhoja varauksiaan. Tavoitteenamme oli luoda käyttäjäystävällinen ja intuitiivinen järjestelmä, joka helpottaa asukasyhdistyksen jäsenten työkalujen lainaamista.

Mahdollisena lisätoiminnallisuutena oli lisätä tietyille käyttäjille erikoisrooli, joka mahdollistaa tuotteiden hallinnoinnin (lisääminen, tuotetietojen päivitys, aktivointi) ilman Djangon hallintasivustoa. Aikataulullisista syistä tämä jätettiin kuitenkin toteuttamatta.

# Teknologia

Rakentamamme web-sovellus pohjautuu Django-ohjelmointikehykseen (W3schools, n.d.; Django Project, n.d.). Toiminnallisuuksien ohjelmointikielenä käytetään Pythonia ja JavaScriptia (Python Software Foundation, n.d.). Sivustojen muotoiluun käytetään HTML-kieltä ja Tailwind CSS – kirjastoa (Mozilla, n.d.;Tailwind CSS, n.d.). Tietokantana käytimme MySQL-tietokantaa (MySQL, n.d.). Tietokantasuunnittelussa käytimme apuna draw.io web-sovellusta. Itse sivusto pyörii Docker-kontissa yhden projektiryhmäläisen palvelimella. Projektin suunnitteluun ja hahmotteluun käytimme Microsoftin Loop-sovellusta (Microsoft, n.d.). Sovelluskehitys tapahtui käyttäen Visual Studiota ja versionhallinta puolestaan GitHubissa.

# Projekti

Jaoimme projektin alussa työtehtävät ja aikataulutimme ne. Onnistuimme hyvin pitämään kiinni suunnitellusta aikataulusta. Käytimme yhteensä noin 27 tuntia projektin toteuttamiseen, mikä oli hyvin lähellä alkuperäistä arviotamme 25 tuntia. Jokainen tiimin jäsen osallistui tasapuolisesti projektin eri vaiheisiin. Alla taulukkomuodossa alustava projektin tavoitteiden aikataulutus ja resurssointi.

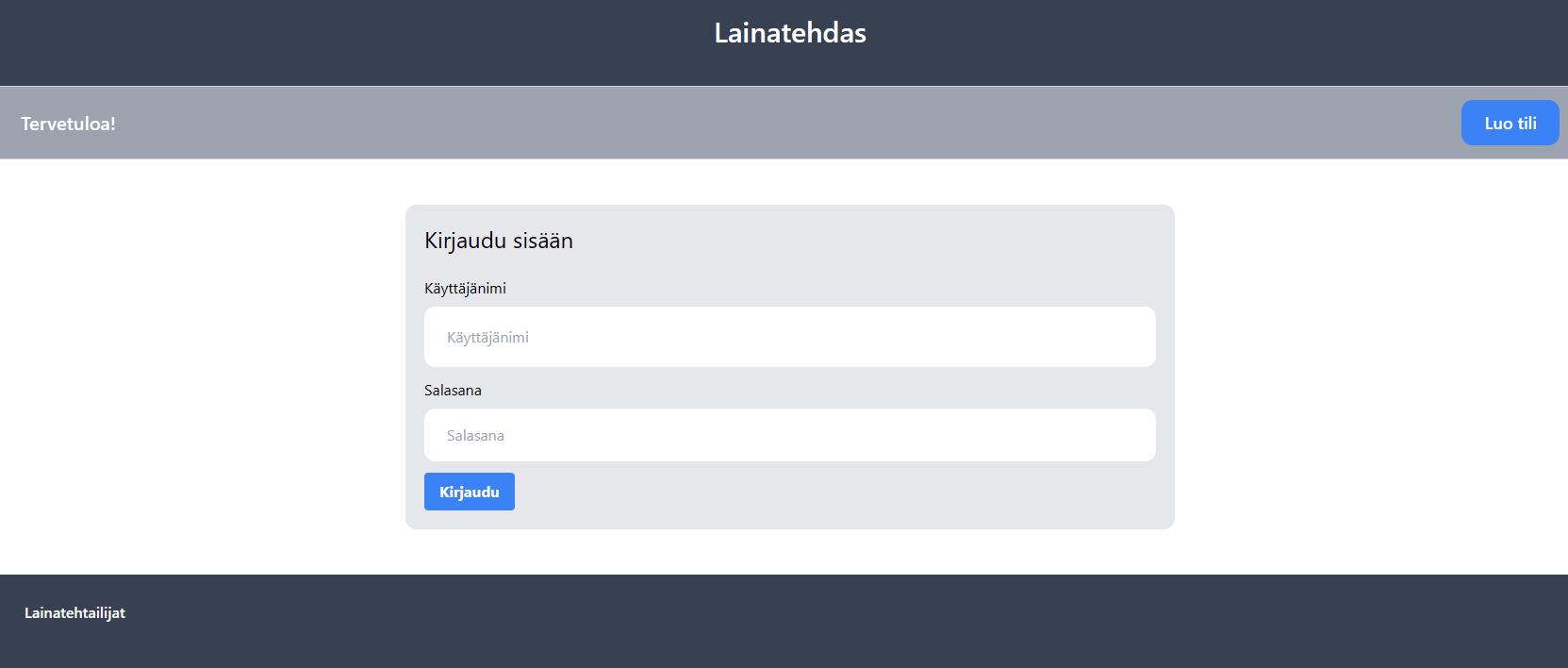
|  |  |  |  |
| --- | --- | --- | --- |
| Tehtävä | Tavoitepäivämäärä | Aika-arvio | Tekijä |
| Rekisteröitymissivu | 12.4.2024 | 2h | Heikki |
| Kirjautumissivu | 12.4.2024 | 2h | Heikki |
| Pääsivu | 12.4.2024 | 3h | Teemu |
| Väliraportti | 12.4.2024 | 1h | Heikki & Teemu |
| Varaus/palautustoiminnallisuus | 19.4.2024 | 3h | Teemu |
| Omat varaukset | 19.4.2024 | 4h | Heikki |
| Webbipalvelimen pystytys | 24.4.2024 | 4h | Teemu |
| Hakutoiminnallisuus | 24.4.2024 | 2h | Heikki |
| Testaaminen | 26.4.2024 | 2h | Heikki & Teemu |
| Loppuraportti + video valmis | 28.4.2024 | 2h | Heikki & Teemu |

Projektin tuloksena syntyi toimiva lainausjärjestelmä, joka täytti asettamamme minimivaatimukset.

## Perustoiminnallisuudet

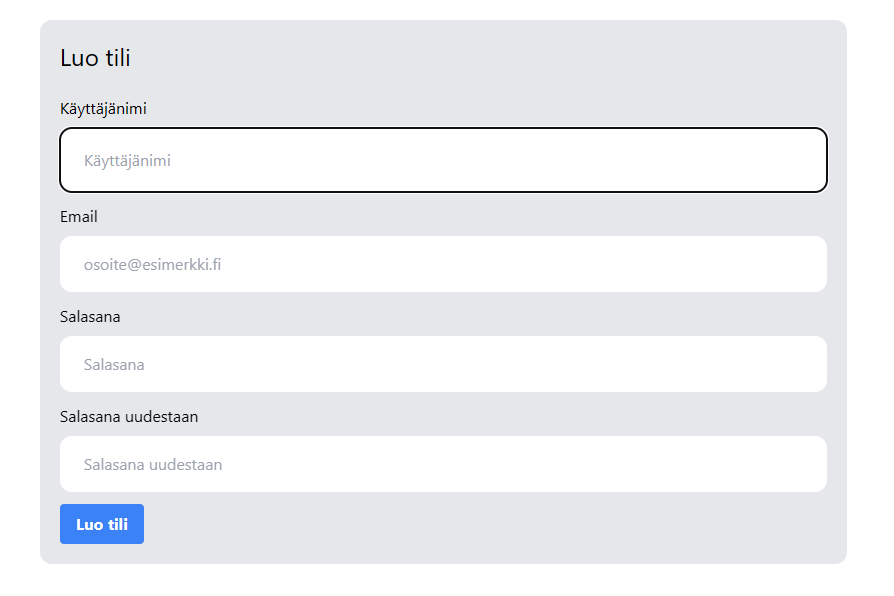
Sovellus on julkaistu osoitteessa <https://lainatehdas.duckdns.org>. Sovelluksen sisäänkirjautumissivulla on normaalin sisäänkirjautumismahdollisuuden lisäksi mahdollisuus luoda tili.

Kuva 1 Kirjautumissivu



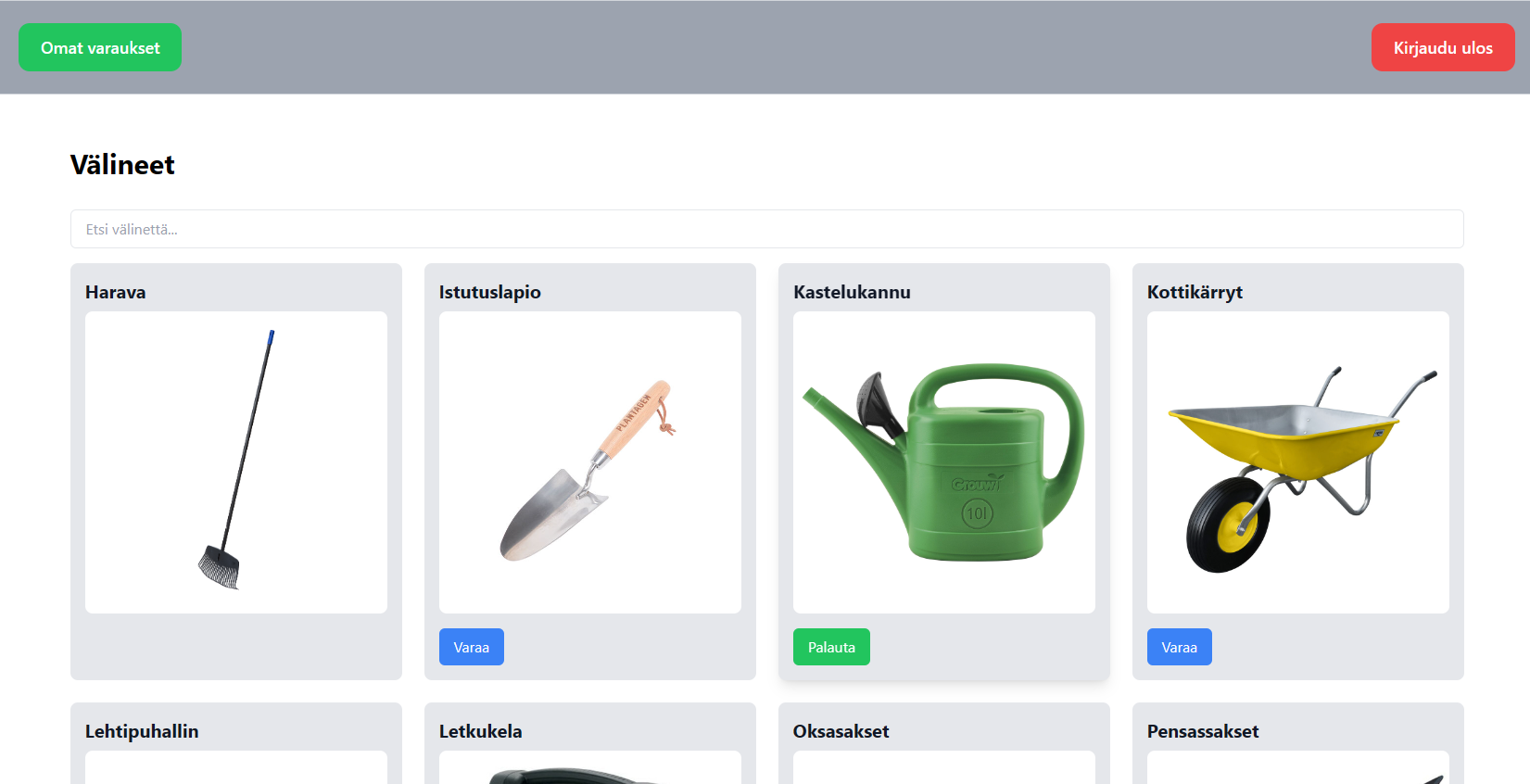
Klikkaamalla ”Luo tili” käyttäjälle näytettään sivu, jossa käyttäjä voi rekisteröityä. Rekisteröitymiseen tarvitaan vain käyttäjänimi, sähköpostiosoite ja salasana. Käyttäjä tallennetaan Djangon käyttämään tietokantaan.

Kuva 2 Rekisteröitymissivu

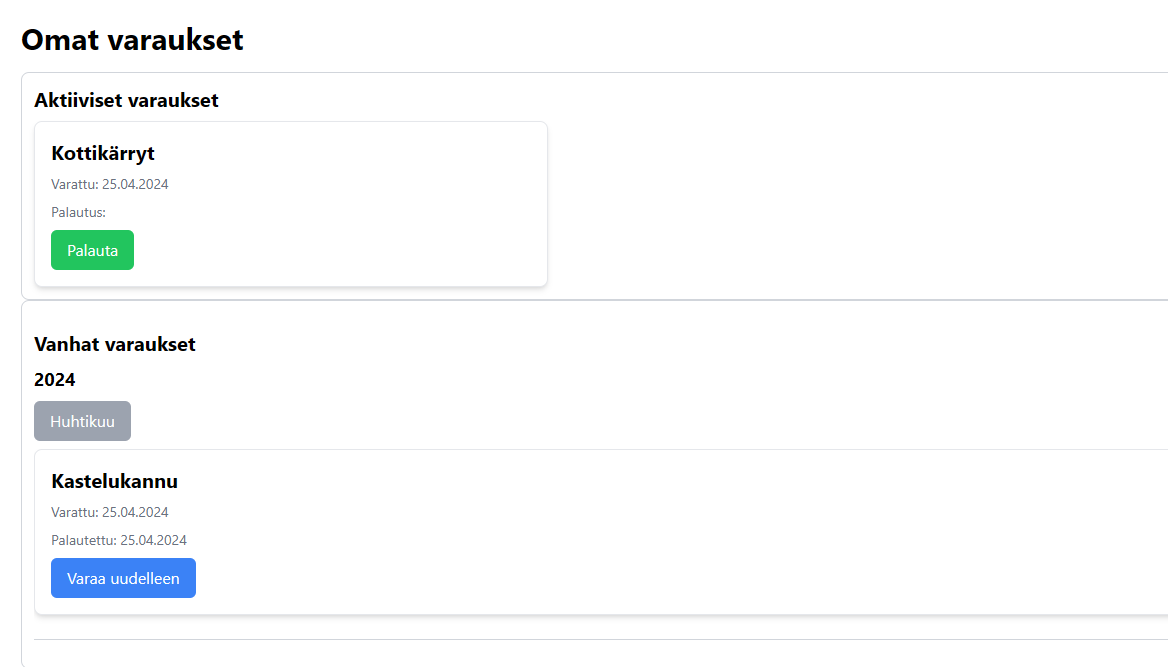


Sisäänkirjautumisen jälkeen käyttäjä näkee palvelussa olevat tuotteet. Tuotteita voi hakea käyttäen hakukenttää. Jos tuote ei ole varattuna toiselle käyttäjälle, tuotteen yhteydessä näkyy "Varaa" -painike, jota painamalla käyttäjä voi helposti varata itselleen tuotteen. Jos tuote on varattuna kyseiselle käyttäjälle, käyttäjä voi myös palauttaa tuotteen painamalla tuotteen yhteydessä näkyvää "Palauta"-painiketta. Klikkaamalla tuotetta, käyttäjä näkee lisätietoja tuotteesta.

Kuva 3 Näkymä palvelussa olevista välineistä

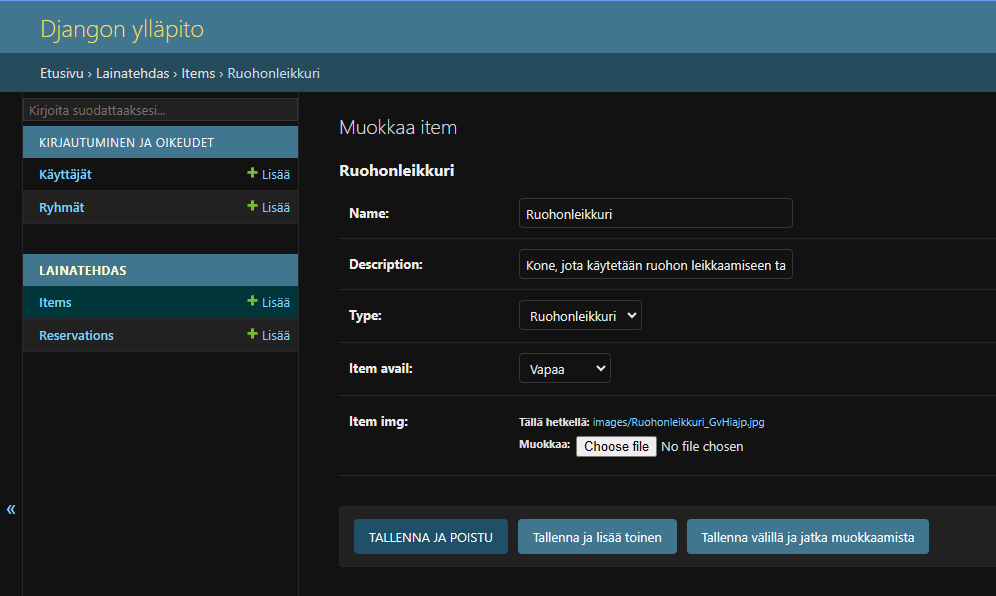


Valitsemalla ”Omat varaukset”, käyttäjä voi tarkistella omia varauksiaan. Aktiivisten varausten osalta käyttäjä voi merkata tuotteen palautetuksi. Vanhat varaukset on ryhmitelty vuoden ja kuukauden mukaan, ja käyttäjä voi helposti uudelleen varata aiemmin varaamansa tuotteen.



Varattavien tuotteiden hallinta tapahtuu Djangon Admin-hallintaliittymän kautta. Mahdollisena lisätoiminnallisuutena oli tarkoitus lisätä tuotteiden hallinta mahdollisuus normaalille sivustolle, riippuen käyttäjän roolista. Aikataulullisista syistä tämä jätettiin kuitenkin toteuttamatta.

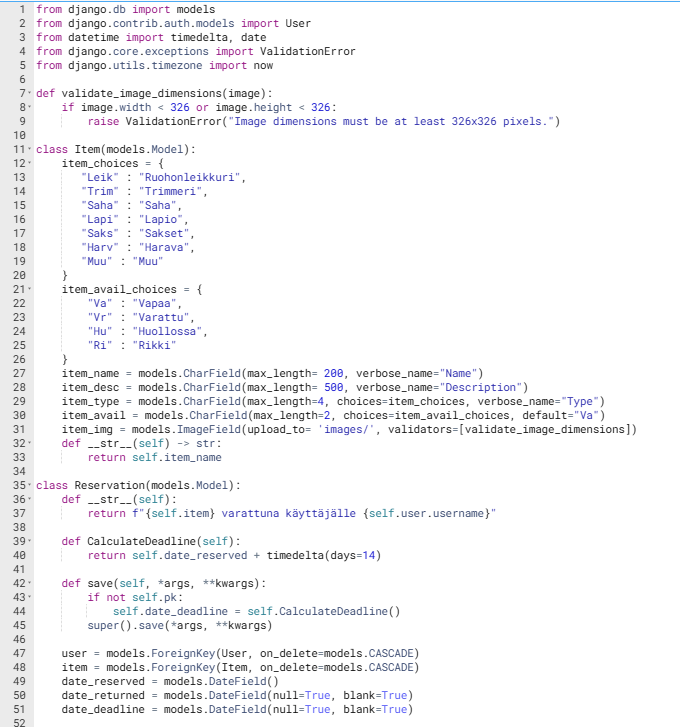
Kuva 4 Tuotteiden hallinta Admin-liittymän kautta



## Tietokanta

Tietokannaksi valitsimme ulkoisen MySQL-tietokannan Djangon oletuksen, SQLite-tietokannan, sijaan. Tietokantamalli koostuu kahdesta pääluokasta: Item ja Reservation. Item-taulu edustaa työkalua, jota voidaan lainata, ja Reservation-taulu edustaa työkalun varausta. Item-taulussa tuotteelle määritellään tuotteen luokka, tila, nimi, kuvaus sekä näytettävä kuva. Reservation-taulussa puolestaan linkitetään tuote ja käyttäjä sekä tallennetaan varaus- ja palautuspäivämäärä.

Kuva 5 Models.py



# Johtopäätökset

Projektin aikana kohtasimme vain pieniä teknisiä haasteita, mutta onnistuimme ratkaisemaan ne ja saavuttamaan tavoitteemme. Yksi suurimmista haasteista oli ajanhallinta, koska meillä oli rajoitetusti aikaa projektin toteuttamiseen. Kuitenkin hyvällä suunnittelulla ja tiimityöllä onnistuimme pitämään kiinni aikataulusta. Olemme tyytyväisiä lopputulokseen ja uskomme, että olemme ansainneet arvosanan 5. Kaikki tiimin jäsenet osallistuivat aktiivisesti projektin eri vaiheisiin ja tekivät tasapuolisesti töitä. Projektin aikaansaannoksena on täysin toimiva sovellus.

Lähteet

Django Project. (n.d.). *Django: The web framework for perfectionists with deadlines*. Haettu 12.4.2024 sivulta <https://www.djangoproject.com/>

Python Software Foundation. (n.d.). *Welcome to Python.org*. Haettu 12.4.2024 sivulta <https://www.python.org/>

Microsoft. (n.d.). *Microsoft Loop. Think, plan, and create together*. Haettu 12.4.2024 sivulta https://www.microsoft.com/en-us/microsoft-loop

Mozilla. (n.d.). *HTML basics*. Haettu 12.4.2024 sivulta <https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/HTML_basics>

MySQL. (n.d.). *MySQL Documentation.* Haettu 12.4.2024 sivulta https://dev.mysql.com/doc/

Tailwind CSS. (n.d.). *Get started with Tailwind CSS.* <https://tailwindcss.com/docs/installation/play-cdn>

W3schools. (n.d.). *Django Tutorial*. Haettu 12.4.2024 sivulta <https://www.w3schools.com/django/>