

# Älykäs aikataulutusasistentti

Tekoälyn soveltaminen



Jouni Kiviperä  
Turun ammattikorkeakoulu

# Älykäs Aikataulutusassistentti

---

Tekoälyn soveltaminen

Opiskelija: Jouni Kiviperä

Oppilaitos: Turun ammattikorkeakoulu

## Sisällysluettelo

1. Johdanto
2. Projektin tavoite
3. Tekninen toteutus
  - 3.1 Käytetyt teknologiat
  - 3.2 Sovelluksen rakenne
  - 3.3 Tekoälymalli ja koulutus
  - 3.4 Google Kalenteri -integraatio
  - 3.5 Käyttöliittymä
4. Haasteet ja oppimiskokemukset
5. Tulokset
6. Yhteenveto
7. Lähteet
8. Liitteet

### 1. Johdanto

Tekoälyä voidaan hyödyntää tehokkaasti ajanhallinnan apuvälineenä. Tässä projektissa kehitettiin älykäs aikataulusassistentti, joka auttaa käyttäjää ennustamaan tehtävien keston ja sijoittamaan ne optimaalisesti kalenteriin.

### 2. Projektin tavoite

Tavoitteena oli kehittää Pythonilla toimiva graafinen sovellus, joka:

- Hyödyntää koneoppimista tehtävien ennustamisessa
- Synkronoituu Google Kalenteriin
- On helposti käytettävissä arjen aikataulujen suunnitteluun

### 3.1 Käytetyt teknologiat

- Python 3.10+
- Scikit-learn (ML)
- Pandas (data)
- Tkinter (UI)

- Google API (kalenteri)
- Pickle (mallin tallennus)

### 3.2 Sovelluksen rakenne

- gui\_ai\_predict.py: käyttöliittymä ja logiikka
- train\_model.py: mallin koulutus
- model.pkl: tallennettu malli
- opetusdata\_ai\_kalenteri.csv: opetusdata
- google\_auth.py: autentikointi Google API:lle

### 3.3 Tekoälymalli ja koulutus

Koneoppimismalli (RandomForestRegressor) koulutettiin data.csv-tiedostolla. Se sisälsi tehtävän vaikeuden, tärkeyden, deadlinen, viikonpäivän ja käyttäjän oman arvion kestosta. Tarkkuus saavutettiin keskimäärin MAE: 9.88 minuuttia.

### 3.4 Google Kalenteri -integraatio

Tehtävät siirtyvät tekoälyn arvioimalla aikavälillä automaattisesti käyttäjän Google Kalenteriin API:n ja OAuth-autentikoinnin avulla.

### 3.5 Käyttöliittymä

Tkinter-pohjainen käyttöliittymä sisältää:

- Tehtävien syötön ja muokkauksen
- Tekoälyennusteen
- Kalenterisynkronoinnin napin

## 4. Haasteet ja oppimiskokemukset

- Google API:n käyttöönotto vaati huolellisuutta
- Mallin koulutus onnistui hyvin pienelläkin datamäärällä
- Tkinterin integrointi koneoppimismalliin toi hyvää käytännön oppia

## 5. Tulokset

Sovellus toimii kokonaisuutena. Tekoäly ennustaa tehtävien keston ja ne viedään kalenteriin. Käyttöliittymä on selkeä ja helppokäyttöinen.

## 6. Yhteenveto

Projekti osoitti, miten tekoälyä voi käyttää arjen tukena. Ratkaisu on laajennettavissa mm. verkkosovellukseksi, jatkuvaksi oppimiseksi tai ryhmäkäyttöön.

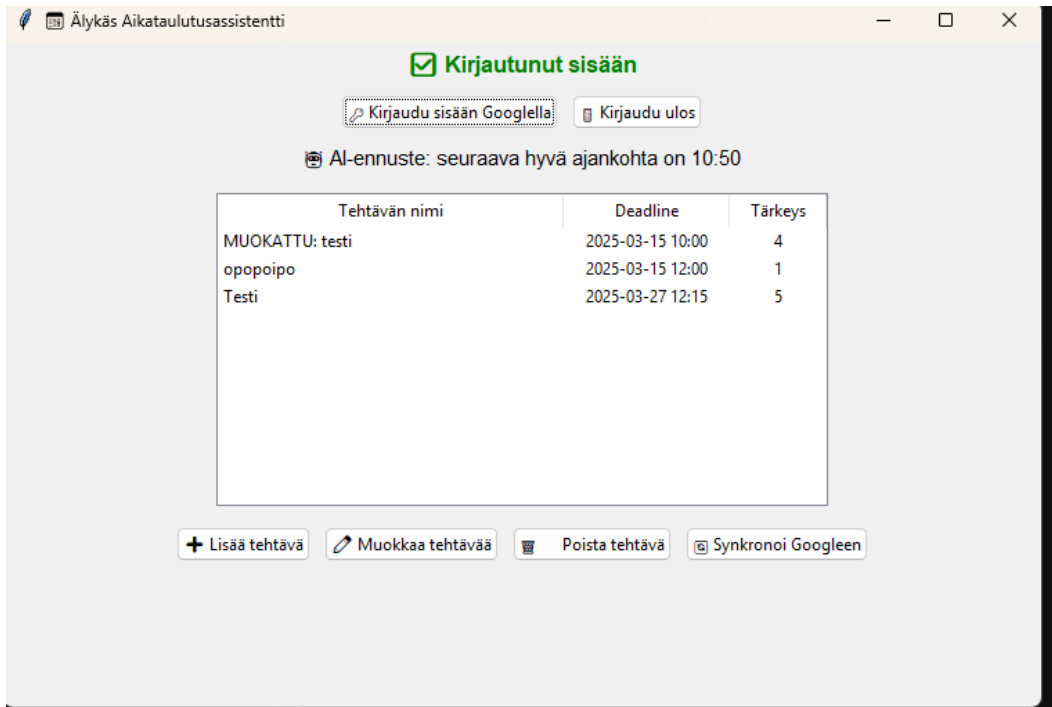
## 7. Lähteet

- <https://github.com/jounikivi/aikataulutusassistentti>
- Google Developers – Calendar API
- Scikit-learn – Random Forest
- Tkinter – Python GUI

## 8. Liitteet

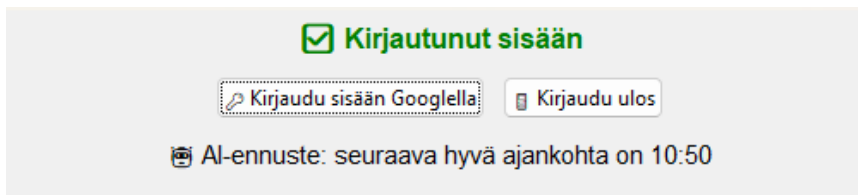
### Liite 1: Sovelluksen käyttöliittymä

Käyttöliittymä sisältää tehtävälistan, tekoälyn ennusteen ja Google-kalenteriin synkronoinnin.



### Liite 2: Tekoälyennusteen näkymä

Sovellus näyttää tekoälyn suositteleman aloitusajan.



### Liite 3: Tehtävät Google Kalenterissa

Synkronoidut tehtävät näkyvät kalenterinäkymässä automaattisesti.

9	10 <ul style="list-style-type: none"><li>• 2ip Opiskelu-projekt</li><li>• 2ip Opiskelu-projekt</li></ul>	11	12	13	14	15 <ul style="list-style-type: none"><li>• 10ap MUOKATTU:</li><li>• 10ap testi</li><li>• 10ap MUOKATTU:</li><li>• 10ap MUOKATTU:</li><li>• 12ip opopolpo</li><li>• 12ip opopolpo</li><li>• 12ip opopolpo</li></ul>
16	17	18	19	20	21	22
23	24	25	26	27 <ul style="list-style-type: none"><li>• 12:15ip Testi</li><li>• 12:15ip Testi</li></ul>	28	29