


WinTelemetryService.exe

Gizli AI Sınav Asistanı (SEB-benzeri Ortamlar için)

Bu proje, Windows tabanlı bir sistemde çalışan **tamamen görünmez bir yapay zekâ sınav asistanı** geliştirmek amacıyla hazırlanmıştır. Uygulama, SEB'e %99 benzeyen sınav ortamlarında çalışmak üzere tasarlanmıştır. Her şey arka planda yürür — **arayüz, uyarı, bildirim, ikon yoktur**.

 **Önemli:** Bu proje sadece **akademik senaryo simülasyonu ve sistem kısıtları üzerine araştırma** amacıyla geliştirilmiştir. Gerçek sınav ortamlarında etik dışı kullanımı asla önerilmez.

Ne Yapar?

- Windows açıldığında otomatik başlar
- Başlangıçta pasiftir
- Ctrl + 2 + 3** tuşlarına aynı anda basılınca aktif olur, tekrar basılınca devre dışı kalır
- Aktifken:
 - Ctrl + Alt** → panodaki soruyu AI modeline gönderir
 - Cevap panoya yazılır
 - Ctrl + Shift** → cevabı yapıştırır (**Ctrl+V**)
 - Yanıt hazır olduğunda **NumLock LED durumu ters çevrilir** (gizli sinyal)
- Windows kapanınca kendisi kapanır

Kullanılan Teknolojiler

- Python 3.12+
- `pywin32`, `keyboard`, `requests`, `pyinstaller`
- `OpenAI API` (veya başka bir LLM servisi)
- `winreg` ile sistem başlangıcına ekleme
- `ctypes` ile NumLock LED kontrolü

Kurulum Aşamaları (9 + 1 adım)

1. Ortamı Hazırla

```
bash
```

```
pip install pywin32 keyboard requests pyinstaller
```

2. Klavye Kısayolları (Hotkey Toggle)

- `ctrl + 2 + 3` → uygulamayı aktif/pasif yapar
- `keyboard.add_hotkey()` ile gerçekleştirilir

3. Clipboard Erişimi

`win32clipboard` ile doğrudan Windows panosuna erişilir:

- `GetClipboardData()` → soru çek
- `SetClipboardData()` → cevap yaz

4. Yapay Zekâ ile İletişim

`requests` ile OpenAI API gibi bir servise bağlanılır `query_ai(question)` fonksiyonu panodaki soruyu gönderip cevabı alır

Örnek (OpenAI için):

```
python

import requests

def query_ai(question):
    headers = {"Authorization": "Bearer sk-APIKEY"}
    data = {
        "model": "gpt-3.5-turbo",
        "messages": [{"role": "user", "content": question}]
    }
    r = requests.post("https://api.openai.com/v1/chat/completions", json=data, headers=headers)
    return r.json()["choices"][0]["message"]["content"]
```

5. Başlangıçta Otomatik Açılma

```
python

import winreg, os

def add_to_startup():
    key = winreg.OpenKey(winreg.HKEY_CURRENT_USER,
        r"SOFTWARE\Microsoft\Windows\CurrentVersion\Run", 0, winreg.KEY_SET_VALUE)
    winreg.SetValueEx(key, "WinTelemetryService", 0, winreg.REG_SZ, os.path.abspath("assistant.exe"))
    winreg.CloseKey(key)
```

6. Penceresiz (Gizli) Çalışma

EXE üretirken `--windowed` parametresi kullan:

```
bash
```

```
pyinstaller --noconfirm --onefile --windowed assistant.py
```

Alternatif olarak ilk satıra:

```
python
```

```
import ctypes
```

```
ctypes.windll.user32.ShowWindow(ctypes.windll.kernel32.GetConsoleWindow(), 0)
```

7. Ana Döngü (Loop)

```
python
```

```
import time
```

```
while True:
```

```
    if active:
```

```
        # Klavye olaylarını dinle, AI çağrısı yap
```

```
        pass
```

```
    time.sleep(0.1)
```

7.5 NumLock LED ile Gizli Geri Bildirim

```
python
```

```
import ctypes
```

```
VK_NUMLOCK = 0x90
```

```
KEYEVENTF_EXTENDEDKEY = 0x1
```

```
KEYEVENTF_KEYUP = 0x2
```

```
def is_numlock_on():
```

```
    return bool(ctypes.windll.user32.GetKeyState(VK_NUMLOCK) & 0x0001)
```

```
def toggle_numlock():
```

```
    ctypes.windll.user32.keybd_event(VK_NUMLOCK, 0x45, KEYEVENTF_EXTENDEDKEY, 0)
```

```
    ctypes.windll.user32.keybd_event(VK_NUMLOCK, 0x45, KEYEVENTF_EXTENDEDKEY | KEYEVENTF_KEYUP, 0)
```

```
def signal_answer_ready():
```

```
    try:
```

```
        toggle_numlock()
```

```
    except Exception:
```

```
        pass
```

8. Ctrl+Alt → Soru Gönder

```
python
```

```
def on_ctrl_alt():  
    question = get_clipboard()  
    answer = query_ai(question)  
    set_clipboard(answer)  
    signal_answer_ready()
```

9. Ctrl+Shift → Cevabı Yapıştır

```
python
```

```
keyboard.add_hotkey('ctrl+shift', lambda: keyboard.write(get_clipboard()))
```



Çalıştırmak İçin

1. Tüm adımları uyguladıktan sonra:

```
bash
```

```
pyinstaller --noconfirm --onefile --windowed assistant.py
```

2. `assistant.exe`'yi çalıştır
3. Bilgisayarı yeniden başlat → uygulama sessizce çalışır
4. `Ctrl+2+3` → aktif et
5. `Ctrl+Alt` → soru al
6. NumLock ışığı değiştiyse → cevap hazır
7. `Ctrl+Shift` → yapıştır



Notlar ve Uyarılar

- Uygulama hiçbir arayüz veya bildirim vermez
- NumLock LED'i olmayan cihazlarda durum değişmeyebilir (hata vermez)
- Bu proje gerçek sınav ortamlarında değil, kurgusal sistem kısıtlarını test etmek için hazırlanmıştır



ETİK UYARI

Bu yazılım yalnızca:

- Akademik araştırma amaçları
- Sistem güvenliđi testleri
- Eğitim materyali olarak

kullanılmalıdır. Gerçek sınav ortamlarında kullanımı akademik dürüstlük kurallarını ihlal eder ve ciddi sonuçları olabilir.