

Univerzitet u Beogradu

Fakultet organizacionih nauka

Katedra za elektronsko poslovanje

Pametne kancelarije

Seminarski rad iz Elektronskog poslovanja

Nastavnik: Asistent Tamara Naumović

Student:Ivan Lukić 2024/0240

Aleksandra Marinković 2024/0572

SADRŽAJ

1. UVOD	3
2. Korisnički zahtevi	4
<i>Opis razvoja</i>	4
3. Izgled aplikacije	7
4. Zaključak	9
5. Literatura	10

1. UVOD

Pametne kancelarije predstavljaju savremeni koncept radnog okruženja koji integriše napredne tehnologije radi poboljšanja produktivnosti, efikasnosti i udobnosti zaposlenih. Ovaj koncept obuhvata primenu softverskih i hardverskih rešenja, uključujući automatizaciju, upravljanje resursima, pametne uređaje i sisteme za obaveštavanje, kako bi se optimizovao rad u kancelarijskom okruženju. U kontekstu brzog tehnološkog napretka i sve veće potrebe za digitalizacijom radnih procesa, pametne kancelarije postaju ključni element modernih poslovnih sistema. One ne samo da olakšavaju svakodnevne zadatke, već i doprinose boljoj organizaciji, smanjenju troškova i povećanju zadovoljstva zaposlenih.

Razlog izbora ove teme leži u želji da se istraži kako softverska rešenja, poput aplikacije Smart Office Planner, mogu unaprediti upravljanje vremenom i zadacima u kancelarijskom okruženju. Ova aplikacija, razvijena kao seminarski rad, predstavlja praktičan primer implementacije tehnologije za podršku pametnim kancelarijama, demonstrirajući mogućnosti organizovanja događaja, slanja obaveštenja i integracije sa postojećim sistemima.

Cilj ovog uvoda je da čitaoca upozna sa konceptom pametnih kancelarija, istakne značaj primene tehnologije u radnom okruženju i predstavi strukturu rada. U nastavku, rad je organizovan u nekoliko ključnih poglavlja, od kojih svako pokriva određeni aspekt teme:

Analiza zahteva za pametne kancelarije: Ovde se razmatraju specifični zahtevi za softverska rešenja u pametnim kancelarijama, uključujući potrebu za upravljanjem događajima, obaveštenjima i integracijom sa drugim sistemima.

Razvoj aplikacije Smart Office Planner: Ovo poglavlje detaljno opisuje proces razvoja aplikacije, uključujući njenu arhitekturu, funkcionalnosti i tehnološke aspekte. Poseban naglasak je na implementaciji upravljanja događajima, obaveštenja i korisničkog interfejsa.

Zaključak: Završno poglavlje sumira ključne nalaze, ističe doprinos aplikacije konceptu pametnih kancelarija i ukazuje na potencijalne pravce za buduća istraživanja.

Kroz ovaj rad, cilj je ne samo da se prikaže praktična implementacija softverskog rešenja, već i da se podstakne interesovanje za dalji razvoj tehnologija koje podržavaju pametne kancelarije. Uvod je osmišljen tako da čitaocu pruži jasan uvid u značaj teme i pripremi ga za detaljnu analizu u narednim poglavljima, uz naglasak na relevantnost i primenljivost u stvarnom poslovnom okruženju.

2. Korisnički zahtevi

Korisnički zahtevi definišu šta ova aplikacija treba da pruži: jednostavno dodavanje uređivanje i brisanje događaja; desktop obaveštenja za nadolazeće događaje, filteri po danu sedmici ili svim događajima. Ujedno mora persistirati podatke u JSON formatu uz intuitivni pristup korisničkom interfejsu. Aplikacija takođe mora biti laka za navigaciju, besplatna i fokusirana na produktivnost u pametnim kancelrijama.

Opis razvoja

Razvoj je započeo definisanjem strukture: Event klasa za događaje, EventManager za operacije sa skladištenjem podataka, NotificationService za obaveštenja. Implementirano je u Pythonu 3.12 sa Tkinter-om za GUI i Pleyer-om za notifikacije. Podaci se spremaju u events.json. Dodane su niti za praćenje obaveštenja bez blokiranja interfejsa.

Programski kod sa objašnjenjem:

```
import json
import os

from datetime import datetime, timedelta
from typing import List, Dict
```

```
class Event:

    def __init__(self, title: str, description: str, date_time: datetime,
notification_minutes: int = 15):

        self.title = title

        self.description = description

        self.date_time = date_time

        self.notification_minutes = notification_minutes

        self.notified = False
```

Programski kod konstruktora Event klase

```
class NotificationService:

    def __init__(self, event_manager: EventManager):
        self.event_manager = event_manager
        self.running = False
        self.check_thread = None

    def start_monitoring(self) -> None:
        if self.running:
            return
        self.running = True
        self.check_thread = threading.Thread(target=self._monitor_loop,
daemon=True)
        self.check_thread.start()
```

NotificationService klasa i njena funkcija startovanja demon procesa u threadu

```
[

{
    "title": "kafica sa menadzerom",
    "description": "u sali 15",
    "date_time": "2025-09-11T09:57:00",
```

```
"notification_minutes": 30,  
"notified": false  
,  
{  
    "title": "probni event",  
    "description": "test",  
    "date_time": "2025-09-12T10:25:00",  
    "notification_minutes": 15,  
    "notified": false  
}  
]
```

Format događaji.json fajla u kom se čuvaju sami događaji

3. Izgled aplikacije

Pametne Kancelarije

Pametne Kancelarije

Kontrole

- Dodaj novi događaj
- Izmeni izabrani
- Obriši izabrani
- Osveži događaje
- Test obaveštenja

Prikaži događaje:

- Sve
- Danas
- Ova nedelja
- Predstojeći

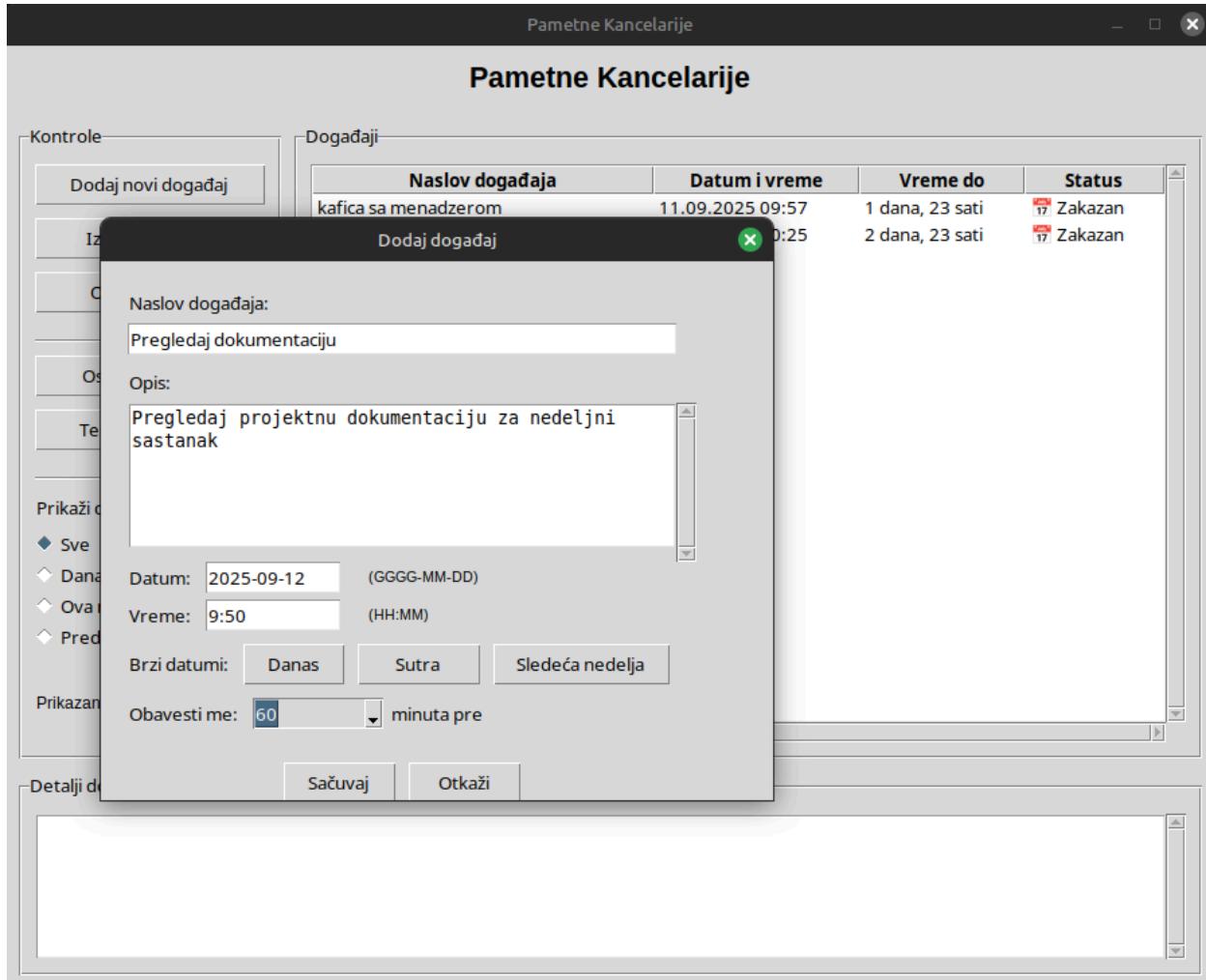
Prikazano 2 događaja (Sve)

Događaji

Naslov događaja	Datum i vreme	Vreme do	Status
kafica sa menadzerom	11.09.2025 09:57	1 dana, 23 sati	 Zakazan
probni event	12.09.2025 10:25	2 dana, 23 sati	 Zakazan

Detalji događaja

Početna stranica same aplikacije



Prozor za lako dodavanje ili izmenu nekog događaja

Prikaži događaje:

- ◆ Sve
- ◆ Danas
- ◆ **Ova nedelja**
- ◆ Predstojeći

Prikazano 3 događaja (Ova nedelja)

Filteri događaja

Detalji događaja

Naslov: sastanak sa profesorom

Datum i vreme: subota, 13. septembar 2025 u 12:00

Vreme do događaja: 4 dana, 1 sati

Više informacija kad selektujemo neki događaj

4. Zaključak

Ovaj seminarski rad obrađuje razvoj "Pametne kancelarije" u formi aplikacije, koja poboljšava produktivnost kroz upravljanje događajima i obaveštenjima. Implementirane su ključne funkcije poput persistencije i filtriranja, koristeći Python tehnologije. Sopstveno mišljenje: Aplikacija ima veliki potencijal za integraciju sa IoT-om i cloud-om za budući razvoj. Prednosti uključuju jednostavnost, ali nedostaci su nedostatak multi-korisničke podrške. Dalji rad bi trebao uključiti sinhronizaciju sa kalendarima i analitiku, kao i povezanost drugih zaposlenih unutar jednog eventa.

5. Literatura

- [1] Plyer Developers, *Plyer Documentation*, [Online]. Available: <https://plyer.readthedocs.io/en/latest/>
- [2] Python Software Foundation, *Python 3 Documentation*, [Online]. Available: <https://docs.python.org/3/index.html>
- [3] Python Software Foundation, *tkinter — Python Interface to Tcl/Tk*, [Online]. Available: <https://docs.python.org/3/library/tkinter.html>