КПІ ім. Ігоря Сікорського

Матеріали IX Міжнародної науково-практичної конференції молодих вчених та студентів «Інженерія програмного забезпечення і передові інформаційні технології (SoftTech-2025)»

КАФЕДРА ІНФОРМАТИКИ ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ
КПІ ім. Ігоря Сікорського
2021

26-28 листопада
Україна, Київ

УДК 004.89:005.8

Бевзюк Нікіта Сергійович, здобувач вищої освіти
Науковий керівник: Мушинський Олег Юрійович, викладач кафедри інформаційного менеджменту, математики та статистики
Університет економіки та права «КРОК», Україна

ПРОЄКТУВАННЯ ІНТЕЛЕКТУАЛЬНОГО АСИСТЕНТА ДЛЯ ПРОЄКТНИХ МЕНЕДЖЕРІВ НА ОСНОВІ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

Анотація. У роботі розглянуто підхід до проєктування інтелектуального вебасистента для підтримки управлінських рішень у проєктах і програмах на основі великих мовних моделей. Метою роботи є створення архітектури та прототипу асистента, що здатний зменшити когнітивне навантаження менеджера, автоматизуючи рутинні аналітичні процеси. Запропоноване рішення поєднує методи комп'ютерних наук і управління проєктами, забезпечуючи інтеграцію з системами PMIS (Jira, Confluence, Azure DevOps) через контур retrieval-augmented generation (RAG) і використання моделі GPT-4. Архітектура побудована за принципами хмарних застосунків із використанням стеку FastAPI, React, PostgreSQL, Qdrant та LiteLLM. Представлено контекстну діаграму IDEF0, що відображає взаємодію вхідних, вихідних, керуючих та підтримувальних елементів системи.

Ключові слова: інтелектуальний асистент; великі мовні моделі; управління проєктами; штучний інтелект

Nikita Bevziuk, postgraduate student,
Academic supervisor: Oleh Mushynskyi, lecturer of the Department of Information Management, Mathematics and Statistics
«KROK» University, Ukraine

DESIGN AN INTELLIGENT ASSISTANT FOR PROJECT MANAGERS BASED ON LARGE LANGUAGE MODELS

Abstract. This paper considers an approach to designing an intelligent web assistant to support management decisions in projects and programs based on large language models. The goal of this work is to create an architecture and prototype of an assistant capable of reducing the cognitive load on managers by automating routine analytical processes. The proposed solution combines computer science and project management methods, ensuring integration with PMIS systems (Jira, Confluence, Azure DevOps) through a retrieval-augmented generation (RAG) circuit and the use of the GPT-4 model. The architecture is built on the principles of cloud applications using the FastAPI, React, PostgreSQL, Qdrant, and LiteLLM stack. A contextual IDEF0 diagram is presented, reflecting the interaction of the system's input, output, control, and support elements.

Key words: intelligent assistant; large language models; project management; artificial intelligence

Вступ. Зростання невизначеності та турбулентності в діяльності організацій підсилює складність управління проєктами та програмами. Унаслідок цього різко зростає обсяг структурованої та неструктурованої інформації, якою має оперативно оперувати керівник проєкту: від вимог і змін до ризиків, обґрунтувань рішень, звітності та комунікацій зі стейкголдерами. Перевантаження інформацією підвищує когнітивні витрати, ускладнює підтримання узгодженості артефактів і негативно впливає на якість прийняття рішень у динамічних умовах.

Паралельно стрімко зростає проникнення штучного інтелекту (ШІ) у професійні практики. Дослідження свідчать, що ШІ стає повсякденним інструментом знаннєвих працівників. Так, уже 43% працівників інтелектуальної праці у США щоденно користуються системами ШІ, тоді як наприкінці 2022 року цей показник не перевищував 10% [1]. Особливо стрімко зростає популярність генеративного ШІ, за даними McKinsey, частка компаній, які регулярно використовують такі інструменти, збільшилася з 33% до 65% лише за рік. Інвестиції у цей сектор у 2023 році сягнули рекордних 25,23 мільярда доларів США [2]. Це свідчить про те, що ШІ дедалі глибше інтегрується в управлінські процеси, створюючи передумови для нових форматів роботи менеджерів.

Основна частина. У межах цього дослідження було спроєктовано інтелектуального веб-асистента підтримки управлінських рішень у проєктах і програмах. Основою рішення є велика мовна модель (LLM) класу GPT-4, яка забезпечує діалог із користувачем природною мовою та інтерпретацію запитів у термінах процесів і артефактів управління. Асистент інтегрується з системами PMIS (напр., Jira/Confluence, Azure DevOps) і корпоративними репозиторіями знань через контур retrieval-augmented generation (RAG) з доменною онтологією, що підвищує точність та трасованість відповідей. Менеджер може формулювати запитання або описувати задачі звичними словами, а система — надавати обґрунтовані відповіді з посиланням на джерела, пропонувати варіанти дій, генерувати аналітичні висновки та короткі звіти.

Ключовою метою застосунку є зменшення когнітивного та операційного навантаження на менеджера в частині рутинних і аналітичних операцій, залишаючи відповідальність за експертне судження та стратегічні рішення за людиною. Асистент підтримуватиме підготовку стратегічних пропозицій, створення звітів, оцінку ризиків і можливостей, планування ресурсів і сценарний аналіз, а також допомагатиме забезпечувати узгодженість артефактів із організаційними політиками та стандартами. Водночас система не замінює фахівця в питаннях, що вимагають глибокого галузевого досвіду, етичної оцінки чи лідерського впливу; вона працює в парадигмі human-in-the-loop із механізмами пояснюваності, обмеженням галюцинацій (через RAG і верифікацію джерел) та контролем якості вихідних матеріалів. Наявні емпіричні дані свідчать, що моделі сімейства GPT ефективно автоматизують рутинні управлінські процеси та підсилюють прийняття рішень [3]; відповідно, запропонований асистент покриває істотну частину щоденних завдань менеджера, підвищуючи продуктивність без втрати керованості та відповідальності.

На ринку існують різні підходи до створення ШІ-асистентів, від локальних програм до хмарних платформ. Для реалізації даного продукту було обрано хмарну архітектуру, оскільки вона забезпечує стабільний доступ до потужних обчислювальних ресурсів, актуальних моделей. Водночас з міркувань сумісності та портовності вся система контейнеризована; інфраструктура підіймається через Docker Compose, що спрощує відтворюваність середовищ

розробки, тестування та експлуатації. Базові налаштування інфраструктури (секрети, ключі доступу) зберігаються у файлі змінних середовища та інжектуються через механізми конфігурації застосунку, що знижує ризики помилок конфігурації та полегшує CI/CD. Для локальної розробки передбачено сценарії швидкого розгортання та зупинки середовищ.

Технологічна основа бекенду базується на FastAPI (Python) як високопродуктивний асинхронний веб-фреймворк для REST-сервісів; застосунок може працювати як REST API так і у вигляді інтерактивного CLI на базі Typer для службових сценаріїв, зокрема індексації, діагностики та експериментів. Керування залежностями і середовищами здійснюється через UV, а допоміжні задачі через Justfile. Такий підхід мінімізує зовнішні обгортки над LLM-SDK та залишає розробнику гнучкість взаємодії з API постачальників моделей.

Фронтенд реалізовано на React, що забезпечує гнучкий та масштабований інтерфейс, який легко адаптувати під корпоративні сценарії. React-клієнт спілкується з FastAPI через REST-ендпоїнти та вебсокети.

Доступ до моделей організовано через LiteLLM — проксі, який уніфікує виклики до 100+ провайдерів LLM і дозволяє задавати зрозумілі псевдоніми моделей та політики маршрутизації у конфігураційному файлі. Така архітектура зменшує прив’язку до окремого вендора, спрощує контрольовані A/B-експерименти і знижує експлуатаційні ризики, оскільки відмову або зміну одного провайдера можна компенсувати перенаправленням на інший без суттєвих змін у верхніх шарах системи.

RAG-контур реалізовано з використанням векторного сховища для семантичного пошуку: документи з корпоративних джерел індексуються після розбиття на контрольовані фрагменти, а пошук виконується над векторними представленнями. RAG-шар забезпечує прив’язку результатів до джерел (ідентифікатори документів, офсети фрагментів, метадані), що служить механізмом трасованості та дозволяє зменшити ймовірність «галюцинацій» шляхом підкріплення генеративних відповідей релевантними витягами. Для підвищення відтворюваності відповіді слід фіксувати версії індексу та параметри ретривера (наприклад, розмір контексту, метрики дистанції), що робить процес відтворюваним у повторних експериментах.

Сховище даних поділено за ролями: транзакційні об’єкти — PostgreSQL (історія діалогів, акаунти, політики доступу, журнали подій), а Qdrant — для векторних представлень та пошуку. За вимоги єдиної СУБД можна використати pgvector як альтернативу для векторного індексу в PostgreSQL; однак у виробничих навантаженнях окремий векторний рушій часто забезпечує кращу латентність та масштабованість на великих корпусах [4].

Основними викликами під час створення є забезпечення швидкодії, стабільності API-з’єднання з OpenAI, а також контроль якості відповідей. Для цього передбачено внутрішній шар логіки — middleware, який перевіряє адекватність результатів і фільтрує неприпустимі відповіді. Такий підхід дозволяє підтримувати високий рівень довіри до системи з боку користувачів.

Для системного представлення функціональної архітектури рішення розроблено контекстну діаграму IDEF0 (рис. 1), яка відображає вхідні, вихідні, керуючі та підтримувальні елементи процесу роботи інтелектуального асистента.

Рис.1. Контекстна діаграма інтелектуального асистенту для керівників проєктів

Висновки. Підсумовуючи, можна стверджувати, що впровадження інтелектуальних асистентів у менеджмент є логічним етапом розвитку цифрових технологій. Дослідження свідчать, що використання GPT-рішень у бізнесі сприяє підвищенню точності управлінських рішень, скороченню витрат часу та оптимізації ресурсів [3]. Наш вебасистент створює новий рівень інтерактивності між менеджером і даними, дозволяючи швидше переходити від аналітики до дії.

Література

1. Open AI. ChatGPT usage and adoption patterns at work URL: https://cdn.openai.com/pdf/3c7f7e1b-36c4-446b-916c-11183e4266b7/chatgpt-usage-and-adoption-patterns-at-work.pdf (дата звернення: 07.11.2025.)

2. Singla A., Sukharevsky A., Yee L., Chui M., Hall B. *The state of AI in early 2024: Gen AI adoption spikes and starts to generate value*. McKinsey. 2024.

3. Kot, S., Khalid, B. GPT Chat Support for Management Practice – Literature Review. *Global Journal of Entrepreneurship and Management.* 2025. DOI: https://doi.org/10.57585/GJEM.025.001

4. Tak, K. How should a beginner choose a database for an AI agent? DEV Community. 2024. URL: https://dev.to/tak089/how-should-a-beginner-choose-a-database-for-an-ai-agent-3l9m (дата звернення: 07.11.2025.)