

## АЛГОРИТМІЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ІНДИВІДУАЛЬНОГО ТА ГРУПОВОГО МЕНЕДЖМЕНТУ ЗАДАЧ ЗАСОБАМИ VUE.JS

Співак І.Я.<sup>1)</sup>, Коваль В.В.<sup>2)</sup>, Крепич С.Я.<sup>3)</sup>

Західноукраїнський національний університет

<sup>1)</sup> к.т.н., доцент; <sup>2)</sup> магістр; <sup>3)</sup> к.т.н., доцент

### І. Вступ

Менеджмент задач є основою успішного функціонування будь-якої організації. Він стосується процесу визначення, планування, організації, виконання та контролю завдань для досягнення конкретних цілей, він набуває особливої актуальності, враховуючи швидкі зміни, нові технології та постійно зростаючі вимоги ринку[1-3]. Для спрощення процесу нотування власних планів потрібно розробити програмне забезпечення, для планування своєї робочої чи позаробочої діяльності, що дасть можливість користувачеві більше часу відвести на реалізацію, аніж кропіткє планування власноруч, що забирає багато часу [4,5]. Додаток буде розроблений як веб-сайт з адаптивністю для усіх девайсів, для більшого залучення користувачів.

### II. Мета роботи

Метою роботи є аналіз існуючих додатків, виділення їх основних переваг та недоліків, проектування прототипу власної системи та реалізація цього прототипу у вигляді додатку тайм-менеджменту задач для індивідуального та групового користування.

### III. Основна частина

На етапі проектування, основною метою є вибір ряду інструментів для побудови додатку та детальний опис архітектури проекту. Для додатку тайм-менеджменту було обрано мікросервісну архітектуру, задля імовірного послідовного розгортання та масштабування без особливих проблем [6,7]. На рисунку 1, наведено схему загальної архітектури проекту.

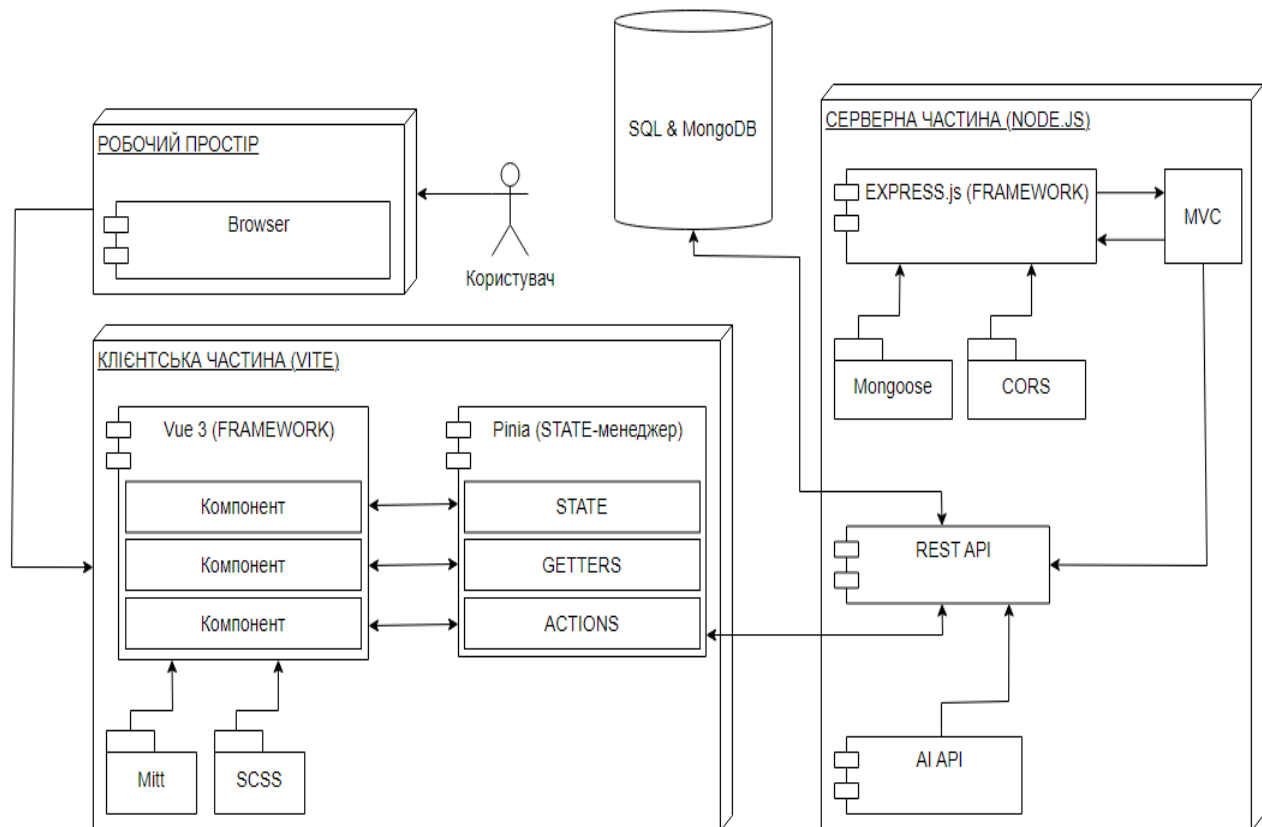


Рисунок 1 – Схема архітектури проекту

Окрім стандартизованих патернів розподілу додатку на клієнтську та серверну частини, опису обміну даними з базами даних та взаємодію окремих компонентів з їх локальними сховищами, до цієї схеми було додано окремий мікросервіс, що працює з API штучного інтелекту [8,9]. Цей компонент є важливою складовою даного прототипу, адже надає можливість використовувати сторонній штучний інтелект для частки функціоналу додатку [10].

В цілому, подібні сервіси, що надають доступ до експлуатації власними розробками, працюють за одним і тим самим алгоритмом взаємодії бота та користувача. Загальна схема описана на рисунку 2.

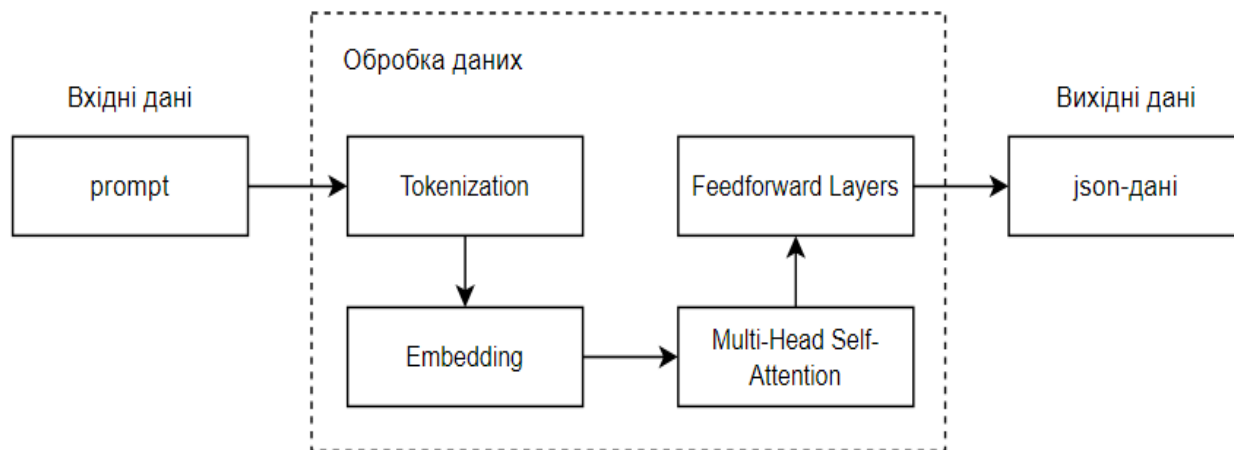


Рисунок 2 – Алгоритм роботи AI API

Описуючи схему, послідовно можна прослідкувати логіку імплементації даної розробки до проекту. Процес поділяється на відправку запиту до API та його відклику з обробленими даними, що повертаються назад до серверної частини. Взаємодія будується на відправці користувачем конкретних вказівок для ШІ (prompts), за якими він зможе власним алгоритмом віддати бажаний результат. За робочий приклад, можна обрати частково функціонуючий алгоритм розподілу справ, методом AI API, який візьме до уваги речення користувача, про відносно масштабну подію чи завдання і розіб'є його на список завдань, що виявиться для рядового користувача більш читабельним та простим у виконанні.

### Висновок

У роботі досліджено підхід до проектування та реалізації додатку тайм-менеджменту задач для індивідуального та групового користування. Виокремлено підхід реалізації AI складової, оформлені діаграми для наочної візуалізації процесів в додатку.

### Список використаних джерел

1. Джулі Абельсон, Structure and Interpretation of Computer Programs (SICP) : навч. посібник : The MIT Press, 2011. 679 с.
2. Річард Гельм, Design Patterns: Elements of Reusable Object-Oriented Software : навч. посібник : Addison-Wesley Professional, 2018. 491 с.
3. I.Spivak, S.Krepuch, S.Spivak, O.Fedorov, "Approach to estimate the level of influence of motivation on the effectiveness of employees9 depending on their needs", in 3<sup>rd</sup> International Conference on Advanced Information and Communications Technologies (AICT), 2019, pp.46-49"
4. Мартін Фолвер, Refactoring: Improving the Design of Existing Code : навч. посібник : Addison-Wesley Professional, 2019. 843 с.
5. Дональд Кеммі, The Art of Computer Programming : навч. посібник : Addison-Wesley Professional, 2019. 452 с.
6. I. Spivak, S.Krepuch, M. Litvynchuk and S. Spivak (2021), "Validation and data processing in JSON format", in 19<sup>th</sup> IEEE International Conference on Smart Technologies, Proceedings (EUROCON2021), pp.326-330
7. Крепич С.Я., Співак І.Я., Пойдич В.С. Оптимізація пам'яті при передачі даних у форматі JSON. Матеріали школи-семінару молодих вчених і студентів, СІТ'2020. Тернопіль, 2022. – с. 25
8. S.Krepuch, I.Spivak, S.Spivak, "Methodology of formation of the individual study plan of the student based on the graph model of the dependence of disciplines", CEUR Workshop Proceedings, 2023, 3426, pp.298-307
9. Craig Larman, Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, 2006, - 736p.
10. Крепич С.Я., Співак І.Я. та Літвинчук М.В. Підхід до роботи із даними у формат JSON. Матеріали школи-семінару молодих вчених і студентів, СІТ'2020. Тернопіль, 2020. – с. 31