

УДК 004.75

DOI <https://doi.org/10.32689/maup.it.2025.1.35>

**Тетяна ЧІЛІКІНА**

кандидат фізико-математичних наук, доцент кафедри комп'ютерних наук та інформаційних технологій, Полтавський університет економіки і торгівлі, [tv.0502@ukr.net](mailto:tv.0502@ukr.net)

ORCID: 0000-0001-8234-9131

**Ганна КАРНАУХОВА**

старший викладач кафедри комп'ютерних наук та інформаційних технологій, Полтавський університет економіки і торгівлі, [karnaukhovahv@gmail.com](mailto:karnaukhovahv@gmail.com)

ORCID: 0000-0001-8089-8743

**Оксана ОРИХІВСЬКА**

старший викладач кафедри комп'ютерних наук та інформаційних технологій, Полтавський університет економіки і торгівлі, [aka.jeita@gmail.com](mailto:aka.jeita@gmail.com)

ORCID: 0000-0003-2775-0832

**Яна ЗАГОРУЙКО**

здобувач спеціальності «Комп'ютерні науки», Полтавський університет економіки і торгівлі, [yana.zahoruiko@gmail.com](mailto:yana.zahoruiko@gmail.com)

ORCID: 0009-0001-3470-532X

## РОЗРОБКА САЙТУ ДЛЯ УПРАВЛІННЯ СВОЇМ РОЗКЛАДОМ ТА ЗАВДАННЯМИ

**Анотація.** У сучасному динамічному світі ефективне управління часом є критично важливим, однак наявні цифрові планувальники часто не відповідають потребам користувачів через складність або обмежений функціонал.

**Мета роботи** – створення вебзастосунку, який поєднує інтуїтивний інтерфейс, ефективне управління цілями та завданнями й адаптацію під індивідуальні потреби користувача.

**Методологія.** Аналітичний підхід до вивчення користувацьких потреб і моделей поведінки, методи UX/UI-дизайну, застосування сучасних фреймворків і мов програмування (TypeScript, React, Next.js, NestJS), а також використання принципів адаптивної верстки, клієнт-серверної архітектури та баз даних MongoDB.

**Наукова новизна** полягає в структурованій інтеграції згаданих підходів у функціонал одного вебзастосунку: обмеження користувача до 5 активних цілей, що забезпечує фокусування; можливість аналізу досягнень через розділ статистики; адаптивність інтерфейсу з підтримкою української та англійської мов. Вебзастосунок реалізовано на основі модульної архітектури з чітким розподілом відповідальностей між модулями (auth, user, tasks, goals, statistics). Клієнтська частина побудована за принципами features based архітектури, що підвищує масштабованість проекту.

Розроблено інструмент, який може бути використаний широким колом користувачів – студентами, фахівцями, фрилансерами, творчими людьми – для ефективного управління особистим часом і досягнення стратегічних життєвих або професійних цілей.

**Висновки.** Запропонований у цій роботі цифровий планувальник має низку ключових переваг, а саме: цільова спрямованість, гнучке планування, мінімалістичний інтерфейс, аналітика прогресу, адаптивність.

**Ключові слова:** веб-планувальник, програма-асистент, управління часом, JavaScript.

## Tetyana CHILIKINA, Hanna KHARNAUKHOVA, Oksana ORIKHIVSKA, Yana ZAHORUIKO. WEBSITE DEVELOPMENT FOR MANAGING YOUR SCHEDULE AND TASKS

**Abstract.** In today's dynamic world, effective time management is critically important, but existing digital planners often do not meet user needs due to complexity or limited functionality.

**The purpose** of the work is to create a web application that combines an intuitive interface, effective management of goals and tasks, and adaptation to the individual needs of the user.

**Methodology.** Analytical approach to studying user needs and behavior patterns, UX/UI design methods, application of modern frameworks and programming languages (TypeScript, React, Next.js, NestJS), as well as use of principles of adaptive layout, client-server architecture and MongoDB databases.

**The scientific novelty** lies in the structured integration of the aforementioned approaches into the functionality of one web application: limiting the user to 5 active goals, which ensures focus; the ability to analyze achievements through the statistics section; adaptability of the interface with support for Ukrainian and English languages. The web application is implemented based on a modular architecture with a clear division of responsibilities between modules (auth, user, tasks, goals, statistics). The client part is built according to the principles of features based architecture, which increases the scalability of the project.

A tool has been developed that can be used by a wide range of users – students, professionals, freelancers, creative people – to effectively manage personal time and achieve strategic life or professional goals.

**Conclusions.** The digital planner proposed in this work has a number of key advantages, namely: goal-oriented, flexible planning, minimalist interface, progress analytics, and adaptability.

**Key words:** web planner, assistant program, time management, JavaScript.

**Постановка проблеми в загальному вигляді та її зв'язок з важливими науковими чи практичними завданнями.** У сучасному світі, де темп життя постійно зростає, а кількість повсякденних завдань невіддільно збільшується, багато людей стикаються з труднощами ефективного розподілу часу. Нестача чіткої структури дня, надлишок інформаційних подразників і нечітко сформульовані цілі призводять до зниження концентрації, зростання рівня стресу та відчуття розгубленості в потоці справ.

Попри наявність великої кількості цифрових планувальників, багато з них або надто примітивні та не забезпечують системності, або ж мають надмірно складний функціонал, що відволікає від головного. Це створює потребу в інструменті, який допомагає зосередитися на важливому, зберігаючи простоту й логічність інтерфейсу.

Зростання попиту на прості та візуально привабливі планувальники, орієнтовані на досягнення цілей, підтверджує актуальність цієї розробки. Вона поєднує в собі зручність користування з чіткою структурою, що допомагає не лише планувати день, а й рухатися до довгострокових стратегічних результатів. Таким чином, створення подібного цифрового інструменту є своєчасною та практичною відповіддю на запити сучасного користувача.

**Аналіз останніх досліджень і публікацій.** В [1] представлено веб-додаток для організації завдань, що включає функції створення, категоризації, пріоритезації, планування та відстеження прогресу. Дослідження оптимізації розкладу завдань у мобільних хмарних обчисленнях для IoT-додатків, що може бути корисним для розробки систем управління завданнями, представлено в роботі [10]. Стаття [12] аналізує методи планування розкладу програмних проєктів, включаючи використання мережевих діаграм для візуалізації залежностей між завданнями. Серед найбільш популярних платформ для управління завданнями та організації часу варто виокремити Trello [3] та Todoist [4]. Обидва інструменти демонструють високий рівень ефективності, але орієнтовані на різні типи користувачів і стилі роботи. Todoist краще підходить для користувачів, які шукають простоту, швидкість і мінімалістичний дизайн для особистого планування. Натомість Trello забезпечує більше можливостей для візуалізації та командної взаємодії, що є важливим при реалізації багатоступеневих проєктів.

**Постановка завдання.** Для розробки власного цифрового планувальника доцільно враховувати найкращі практики цих сервісів, зокрема: інтуїтивність інтерфейсу, гнучкість налаштувань, можливість аналітики. Це дозволить створити конкурентоспроможний продукт, що відповідає сучасним очікуванням користувачів [9].

**Виклад основного матеріалу дослідження.** Основним матеріалом для розробки планувальника стали концепції з таких галузей знань:

Теорія goal-setting theory: дослідження Едвіна Локка та Гаррі Латема доводять, що постановка конкретних, обмежених за кількістю і складністю цілей підвищує продуктивність та мотивацію. У планувальнику цю ідею реалізовано через ліміт у 5 активних цілей, що дозволяє уникнути розфокусу та зберегти чіткість у плануванні [6].

Методи тайм-менеджменту: теорії Девіда Аллена (метод GTD), техніка Pomodoro Франческо Чірілло, принцип Парето (80/20), вказують на необхідність структурування задач і фокусування на пріоритетних діях [2]. У розробленому додатку завдання структуруються навколо цілей, які формують фокус користувача на найважливішому.

UX/UI-дизайн: сучасні принципи дизайну (керованість, контрастність, доступність) вимагають адаптації інтерфейсів до потреб користувачів, включаючи темну та світлу тему, а також багатомовність [7]. У планувальнику реалізовано підтримку української та англійської мов, що розширює аудиторію проєкту.

Візуальна аналітика: з точки зору користувацької мотивації важливо мати змогу відстежувати власні досягнення. Тому в планувальнику реалізовано щомісячну та річну статистику, де видно кількість виконаних завдань у розрізі кожної цілі. Це сприяє як самоаналізу, так і покращенню особистої ефективності [5].

Отже, у межах теоретичного дослідження було враховано сучасні підходи до планування, психології продуктивності, користувацького досвіду та аналізу даних. Це дозволило сформувати основу для створення цифрового планувальника, який поєднує простоту, функціональність і гнучкість.

*Опис структури сайту.*

Сайт-планувальник побудований за принципами мінімалізму, фокусуючись на продуктивності та зручності користувача. Його структура складається з таких основних розділів:

1. Головна сторінка сайту.

- Коротка інформація про проєкт.
- Кнопка для авторизації користувача.
- Контактна інформація та посилання на соцмережі або додаткові ресурси.

## 2. Сторінка авторизації.

- Можливість входу за допомогою електронної пошти або Google-акаунту.
- Форма реєстрації нового користувача.

## 3. Головна сторінка планувальника.

- Можливість створення до 5 активних цілей.
- Додавання необмеженої кількості завдань до кожної цілі.
- Відображення поточних завдань та короткої тижневої статистики виконання.

## 4. Тижневе планування.

- Інтерфейс для планування завдань на поточний тиждень.
- Можливість перетягування завдань між днями тижня (drag-and-drop).
- Кожне завдання пов'язане з конкретною ціллю, що дозволяє зберігати фокус.

## 5. Статистика.

- Доступна щомісячна та річна статистика.
- Виводиться кількість виконаних завдань у розрізі цілей.
- Можна простежити активність користувача, продуктивні дні та прогрес.

## 6. Налаштування.

- Зміна мови інтерфейсу (українська / англійська).
- Перемикання між темною і світлою темами.
- Оновлення електронної пошти, паролю, аватару користувача.

На UML-діаграмі класів змодельовано взаємозв'язки між користувачем, цілями, завданнями та статистикою у планувальнику завдань (рис. 1).

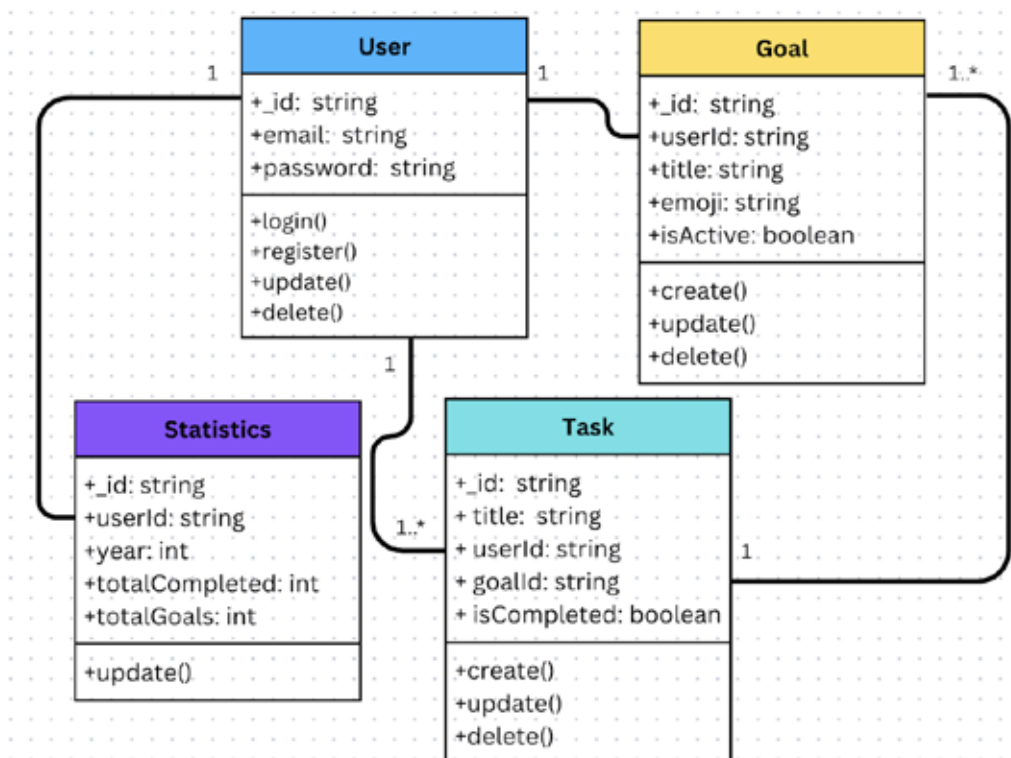


Рис. 1. UML-діаграма класів планувальника

- Зв'язок між User та Goal Тип зв'язку: 1 → 0\*. Один користувач може створювати багато цілей. Кожна ціль належить лише одному користувачу. Таким чином, між сутностями User і Goal реалізовано зв'язок типу "один до багатьох".

- Зв'язок між User та Task Тип зв'язку: 1 → 0\*. Один користувач може мати багато завдань. Кожне завдання належить одному користувачу. Це відображає зв'язок "один до багатьох" між користувачем та його завданнями.

- Зв'язок між User та Statistics Тип зв'язку:  $1 \rightarrow 0$ .\* Один користувач може мати декілька записів статистики – за різні роки або періоди відстеження. Таким чином, між User і Statistics існує зв'язок типу "один до багатьох".

- Зв'язок між Goal та Task Тип зв'язку:  $1 \rightarrow 0$ .\* Кожна ціль може включати багато завдань, які сприяють її досягненню. Завдання належить одній конкретній цілі. Це зв'язок "один до багатьох" між Goal і Task.

Проект організовано за модульним принципом. Виділено основні модулі: auth, user, tasks, statistics, goals, date.

Кожен модуль містить три основні частини:

- Controller – відповідає за обробку HTTP-запитів (GET, POST, PUT, DELETE) та викликає відповідні методи сервісу.

- Service – містить основну бізнес-логіку модуля (створення, оновлення, валідація, зв'язки з базою даних).

- Module – слугує як конфігураційний елемент, який реєструє контролери та сервіси модуля та забезпечує їхню взаємодію.

#### 1. Модуль авторизації auth.

Модуль авторизації відповідає за:

- реєстрацію нового користувача,
- вхід у систему (login) з перевіркою облікових даних,
- оновлення та валідацію токенів доступу (JWT),
- інтеграцію з Google OAuth.

Модуль включає:

- Auth Controller – приймає запити від клієнта (/register, /login, /refresh-token, тощо),
- Auth Service – обробляє логіку перевірки користувача, генерації токенів, хешування паролів,
- Auth Module – підключає залежності, реєструє потрібні провайдери та імпортує інші модулі (наприклад, UserModule для доступу до користувачів).

#### 2. Модуль користувача user.

Модуль користувача відповідає за:

- отримання інформації про користувача,
- оновлення особистих даних (ім'я, email, аватар, тощо),
- видалення облікового запису.

Модуль включає:

- User Controller – приймає запити від клієнта,
- User Service – обробляє логіку додавання та зміни даних користувача,
- User Module – підключає залежності.

#### 3. Модуль створення та редагування завдань tasks.

Модуль відповідає за створення, оновлення, видалення завдань.

Модуль включає:

- Tasks Controller – приймає запити від клієнта для додавання, оновлення завдань,
- Tasks Service – обробляє логіку додавання та зміни даних завдань,
- Tasks Module – підключає залежності.

#### 4. Модуль створення та редагування цілей goals.

Модуль відповідає за отримання, створення, оновлення, видалення цілей.

Модуль включає:

- Goals Controller – приймає запити від клієнта для отримання, додавання, оновлення цілей,
- Goals Service – обробляє логіку отримання цілей, додавання та зміни цілей,
- Goals Module – підключає залежності.

#### 5. Модуль статистики statistics.

Модуль відповідає за отримання, створення статистики.

Модуль включає:

- Statistics Controller – приймає запити від клієнта для отримання статистики,
- Statistics Service – обробляє логіку отримання, розрахунку та оновлення статистики,
- Statistics Module – підключає залежності.

#### 6. Модуль статистики date.

Модуль відповідає за отримання даних про поточний або минулий тиждень.

Модуль включає:

- Statistics Service – приймає запити від інших модулів про необхідні дати,
- Statistics Module – підключає залежності.

До загальних принципів можна віднести наступні:

- Кожен модуль ізольований, відповідає за одну область відповідальності (SRP).
- Взаємодія між модулями відбувається через сервіси, які експортуються через `@Module({ exports: [...] })`.
- Дані зберігаються в MongoDB [11], моделі створюються за допомогою Typegoose схем.
- Усі змінні оточення зберігаються у файлі `env`.

Також для забезпечення стабільності, безпеки й зрозумілості API в проєкті реалізовані додаткові технічні шари: DTO, валідація та guard-и.

*Реалізація клієнтської частини.*

Проектна структура реалізована за принципами features based архітектури з акцентом на модульність, повторне використання компонентів та простоту масштабування. Features based підхід передбачає розділення логіки за функціональними частинами застосунку, де кожна фіча організована у вигляді окремого модуля, що включає всі необхідні елементи для її роботи. Уся логіка проєкту розділена на окремі функціональні блоки, які відповідають модулям бекенду: auth, user, tasks, goals, statistics.

Оскільки проєкт включає авторизацію та використання HTTP токенів, для коректної роботи з авторизованими користувачами створюємо interceptor, який автоматично додає токени до запитів та оновляє їх, а також middleware у Next.js, що працює на серверній стороні фронтенду й перевіряє автентичність користувача через HTTP cookie перед обробкою запитів.

Окрім цього, у проєкті реалізовано багатомовність інтерфейсу через систему перекладів, що дозволяє адаптувати застосунок для користувачів з різних мовних середовищ. Повідомлення про успішні або помилкові дії виводяться через модуль сповіщень, що забезпечує зворотний зв'язок у взаємодії з користувачем. Для уніфікації зовнішнього вигляду застосунку та забезпечення консистентності дизайну створено окремі файли зі стилями, які охоплюють як глобальні налаштування, так і стилі для окремих компонентів.

Розроблений сайт дозволяє користувачеві створювати, переглядати та редагувати свій щоденний розклад із фокусом до 5 цілей. Користувач взаємодіє з інтуїтивним інтерфейсом, у якому всі елементи взаємопов'язані: завдання можна створювати, сортувати, редагувати, перетягувати, позначати як виконані та зв'язувати з довгостроковими цілями (див. рис. 2).

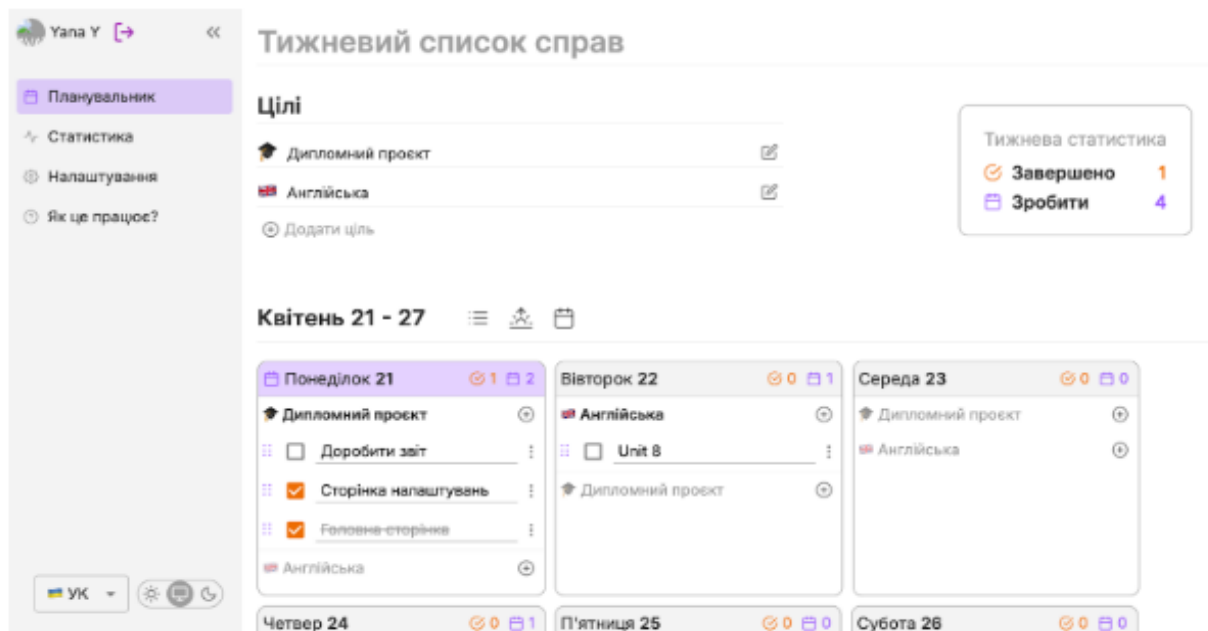


Рис. 2. Головна сторінка планувальника

**Висновки з даного дослідження та перспективи подальших розвідок у даному напрямі.** Створений вебзастосунок демонструє гнучкість, адаптивність до потреб користувача та можливість інтеграції нових функцій, таких як аналітика звичок, а також підтримка мобільної версії. Це закладає ґрунт для подальшого вдосконалення продукту й масштабування його функціональності для ширшої аудиторії.

**Список використаних джерел:**

1. Adebayo A., Alawode O. Web-Based Task Organizer for Enhanced Productivity. *ResearchGate*. 2023. URL: [https://www.researchgate.net/publication/384666926\\_Web-Based\\_Task\\_Organizer\\_for\\_Enhanced\\_Productivity](https://www.researchgate.net/publication/384666926_Web-Based_Task_Organizer_for_Enhanced_Productivity)
2. Allen D. Getting Things Done: The Art of Stress-Free Productivity. Penguin Books. Publisher: Penguin Books. 2002. 288 p
3. Atlassian компанія. Trello – візуальний інструмент для управління завданнями та проектами. URL: <https://trello.com>.
4. Doist компанія. Todoist – це популярний додаток для управління завданнями. URL: <https://todoist.com>.
5. Few S. Information Dashboard Design: The Effective Visual Communication of Data. O'Reilly Media. Publisher: O'Reilly. 2006. 224 p
6. Locke E. A., Latham G. P. Building a practically useful theory of goal setting and task motivation: A 35-year odyssey. *American Psychologist*, 2002. 57(9), 705–717. <https://doi.org/10.1037/0003-066X.57.9.705>
7. Norman D. A. The Design of Everyday Things. MIT Press. Publisher: The MIT Press. 2013. 268 p.
8. Ольховська О. В., Кошова О. П., Черненко О. О., Тур В. М. РОЗРОБКА МОБІЛЬНОГО АВТОМАТИЗОВАНОГО АГРЕГАТОРА ВАКАНСІЙ В ІТ-ГАЛУЗІ. *Таврійський науковий вісник. Серія: Технічні науки*, 2024. (6), 72–80. <https://doi.org/10.32782/tnv-tech.2024.6.8>
9. Peter A. Heslin, Jay B. Carson, Don Vandewalle Practical Applications of Goal Setting Theory to Performance Management URL: [https://www.researchgate.net/publication/228211042\\_Practical\\_Applications\\_of\\_Goal\\_Setting\\_Theory\\_to\\_Performance\\_Management](https://www.researchgate.net/publication/228211042_Practical_Applications_of_Goal_Setting_Theory_to_Performance_Management) (дата звернення: 30.01.2025).
10. Ramalingam V., Dhavachelvan P., Sundararajan E. An Efficient Dynamic Decision-Based Task Optimization and Scheduling in Mobile Cloud Computing for IoT Applications. *Procedia Computer Science*, 2023. 222, 134–140. Elsevier. <https://doi.org/10.1016/j.procs.2023.01.023>
11. Ranjeet Jha, MongoDB Essentials: Build Faster, Perform Better, Store Smarter, Scale Bigger, and Master Document Database, 2025, 380 p.
12. Zhao J. Research on Software Project Schedule Planning Technology. *Procedia Computer Science*, 2023. 222, 225–232. Elsevier. <https://doi.org/10.1016/j.procs.2023.01.030>