РОЗДІЛ 1. АНЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ, МЕТОДІВ ТА ЗАДАЧ РОЗРОБЛЕННОЇ АВТОМАТИЗОВАНОЇ СИСТЕМИ

* 1. Аналіз предметної області.

Аналіз предметної області є важливим етапом у розробці програмного забезпечення, оскільки він дозволяє збагнути специфіку та особливості середовища, в якому має функціонувати програмна система. Цей етап передбачає детальне дослідження об'єкта аналізу, включаючи його потреби, вимоги, характеристики та взаємозв'язки з іншими системами або процесами:

Аналіз предметної області є важливим етапом у розробці програмного забезпечення, оскільки він дозволяє збагнути специфіку та особливості середовища, в якому має функціонувати програмна система. Цей етап передбачає детальне дослідження об'єкта аналізу, включаючи його потреби, вимоги, характеристики та взаємозв'язки з іншими системами або процесами.

Під час аналізу предметної області визначаються такі ключові аспекти:

1. Потреби користувачів: Ретельне вивчення потреб та вимог користувачів до програмної системи. Це включає визначення функціональності, яка має бути реалізована, та інтерфейсу, який має забезпечити зручність взаємодії.

2. Характеристики середовища: Аналіз характеристик середовища, в якому буде функціонувати програмна система, включаючи технічні, організаційні та культурні аспекти. Це допомагає врахувати особливості роботи системи в конкретних умовах.

3. Вимоги до системи: Визначення функціональних та нефункціональних вимог до програмної системи на основі виявлених потреб користувачів та особливостей середовища.

4. Аналіз конкурентів та аналогів: Вивчення існуючих рішень на ринку, які вирішують подібні завдання або задовольняють подібні потреби. Це допомагає зрозуміти сильні та слабкі сторони конкурентів та забезпечити перевагу у власній розробці.

5. Аналіз ризиків та обмежень: Визначення потенційних ризиків та обмежень, які можуть вплинути на успішність реалізації програмної системи, та розробка стратегій їх управління.

Аналіз предметної області є основою для подальшого проектування, розробки та впровадження програмної системи, тому його важливість важко переоцінити.

Отже, проаналізувавши наведені ключові аспекти потреб кінцевих користувачів виокремлено наступні вимоги для веб-застосунку

1. Географічні дані та їх використання:

- Використання географічних даних є центральним аспектом предметної області. Геодезичні координати, висота точок, топографічні мапи та інші географічні атрибути використовуються для різноманітних цілей, від навігації та картографування до наукових досліджень та планування інфраструктури.

2. Технології веб-картографії:

- Застосування сучасних технологій веб-картографії, таких як бібліотека Leaflet, дозволяє створювати інтерактивні та привабливі карти в інтернет-додатках. Це дає можливість користувачам взаємодіяти з географічними даними у зручний спосіб та отримувати корисну інформацію.

3. Використання географічних API:

- Використання географічних API, таких як OpenTopoMap, надає доступ до високоякісних географічних даних та сервісів, які можна використовувати для візуалізації та аналізу географічних об'єктів. Це включає в себе отримання топографічних карт, ефектні відображення рельєфу та інші географічні аспекти.

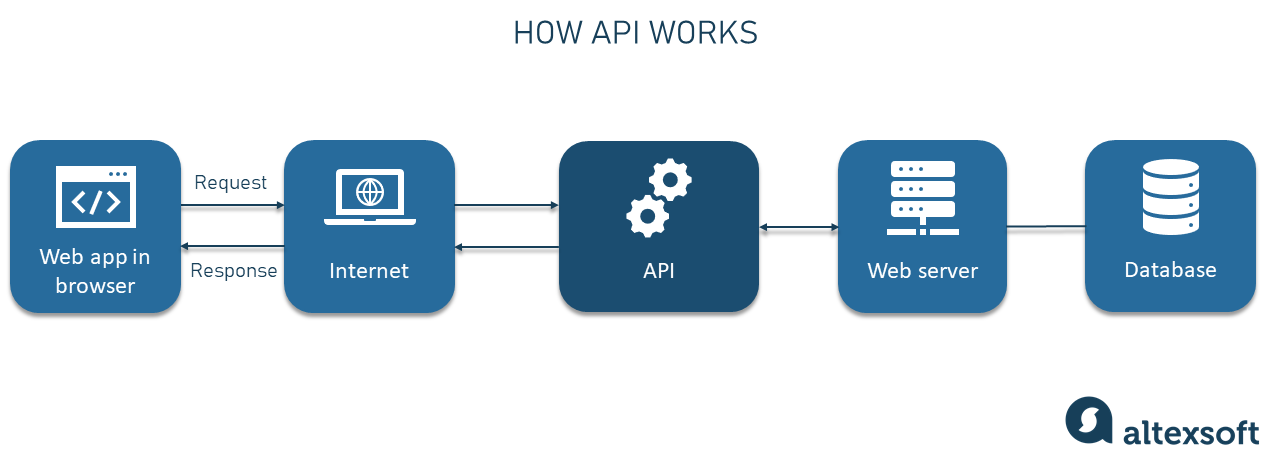


Рис. 1.1 - Використання API у застосунках.

4. Потреби користувачів:

- Важливим елементом аналізу є розуміння потреб та очікувань користувачів. Наприклад, в навігаційних застосунках може бути важливою точність та швидкодія відображення мап, тоді як у наукових дослідженнях може бути важливою доступність різноманітних географічних даних для аналізу.

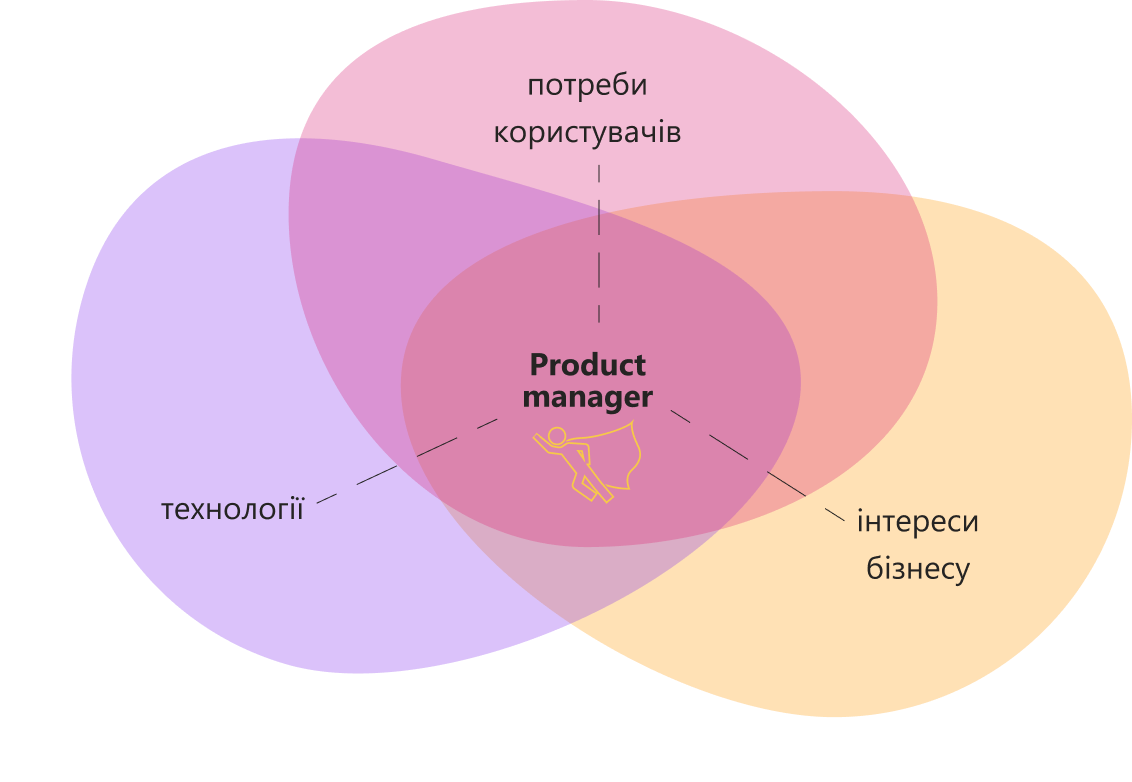


Рис. 1.2 – Проектний менеджмент.

5. Інтеграція з іншими системами:

- При аналізі предметної області важливо врахувати можливості інтеграції з іншими системами, які використовують географічні дані. Наприклад, інтеграція з системами навігації або сервісами доставки може покращити функціональність та цінність системи для користувачів.

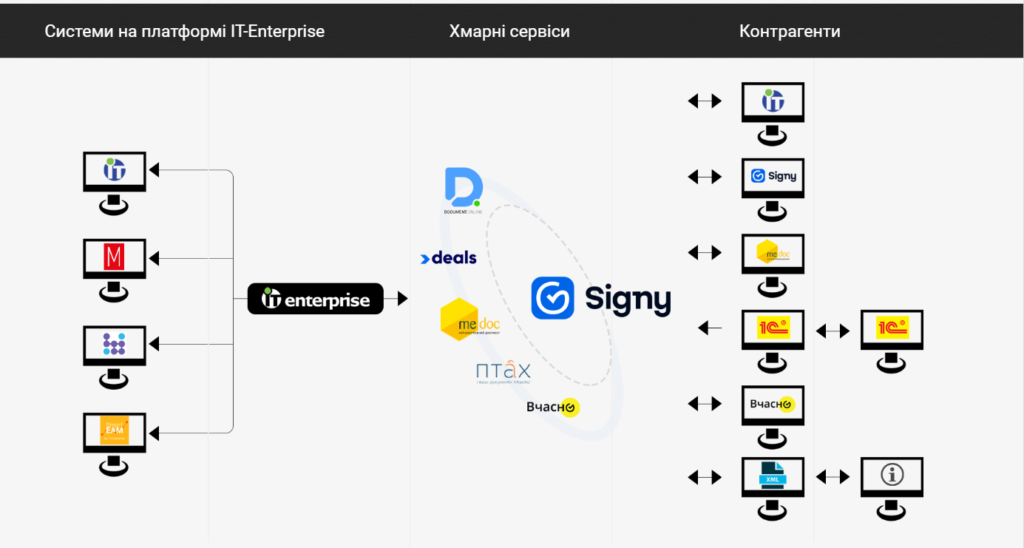


Рис. 1.3 – Інтеграція з іншими системами.

Аналіз предметної області дозволяє зрозуміти вимоги та потреби користувачів, визначити ключові аспекти та особливості середовища, в якому буде використовуватися система, та підготувати підґрунтя для подальшої розробки програмного забезпечення.

* 1. Ознайомлення з особливостями наявних систем

Ознайомлення з особливостями функціонування наявних систем є критичним етапом в розробці програмного забезпечення, оскільки дозволяє розробникам зрозуміти вже існуючі рішення та їхні можливості, а також визначити недоліки, які можна уникнути або покращити в новій системі. Це допомагає уникнути повторного винайдення колеса, використовуючи найбільш ефективні та оптимальні підходи, які вже були випробувані та визнані відповідними.

Перед тим, як розглядати приклади наявних систем, важливо розглянути деякі ключові принципи та переваги ознайомлення з ними.

По-перше, вивчення існуючих систем дозволяє розробникам отримати унікальні інсайти щодо того, як вони працюють та які проблеми вони вирішують. Це може надихнути на нові ідеї або допомогти уникнути помилок, які вже були зроблені.



Рис. 1.4 – Ознайомлення з принципами роботи наявних систем.

По-друге, ознайомлення з існуючими системами дозволяє виявити їхні переваги та недоліки. Наприклад, може виявитися, що попередні системи мають обмеження у швидкодії або в масштабованості, які можна покращити в новому проекті.



Рис. 1.5 – Ознайомлення з існуючими системами.

По-третє, аналіз існуючих рішень допомагає зрозуміти потреби користувачів та ринкові вимоги. Це дозволяє зорієнтуватися на конкретні проблеми, які потрібно вирішити, та спрямовує розробку в тому напрямку, що максимально відповідає потребам користувачів.



Рис. 1.6 – Аналіз існуючих рішень.

Наприклад, якщо ми розробляємо систему управління проектами, ознайомлення з існуючими рішеннями, такими як Asana, Trello або Jira, дозволить нам зрозуміти, як ці системи вирішують завдання управління завданнями, спілкування з командою та відстеження прогресу проектів. Такий аналіз допоможе визначити переваги та недоліки кожної системи, а також виявити можливості для покращення в новому проекті.

Обширний огляд особливостей функціонування деяких наявних систем:

1. Google Maps:
   * є однією з найпопулярніших систем картографування та навігації, вона має широкий функціонал та величезну базу користувачів.
   * надає користувачам можливість швидко знаходити маршрути, переглядати зображення з супутників, оцінювати місця та додавати свої відгуки.



Рис. 1.7 – Google Maps.

1. OpenStreetMap (OSM):
   * є вільно доступною глобальною мапою, яка розробляється спільнотою користувачів. Вона має відкритий код та дозволяє редагувати та покращувати карту всім бажаючим.
   * зазвичай використовується в проектах з відкритим кодом, а також у веб-сервісах, де потрібна географічна інформація, яка може бути редагована та оновлювана спільнотою.

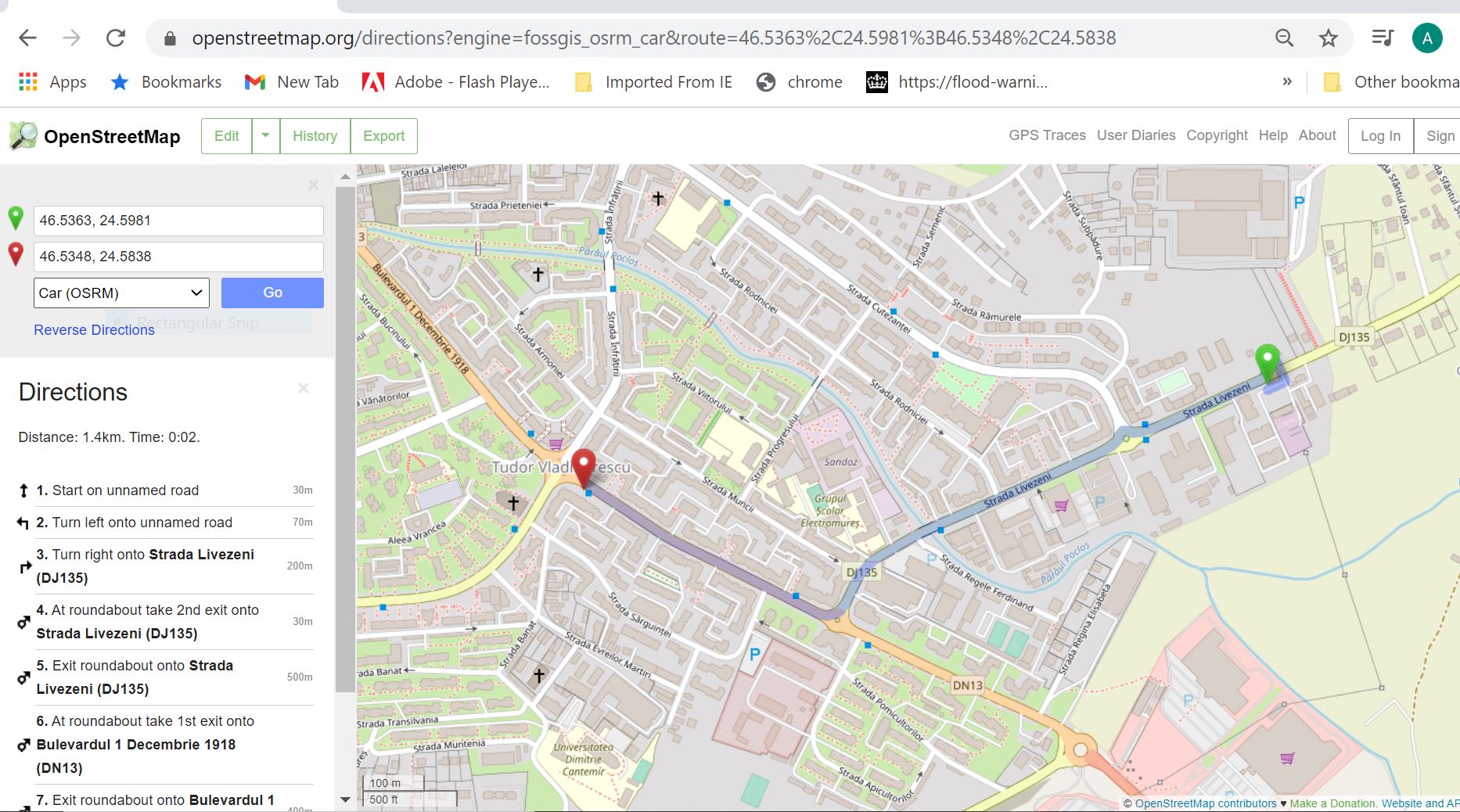


Рис. 1.8 – OpenStreetMap (OSM).

1. HERE Maps:
   * відома своєю великою кількістю функцій та високою точністю геоданих. Вона широко використовується в автомобільних системах навігації та геолокації.
   * використовується в автомобільних GPS-навігаторах, мобільних додатках для навігації та в інших застосунках, де потрібна точна географічна інформація.

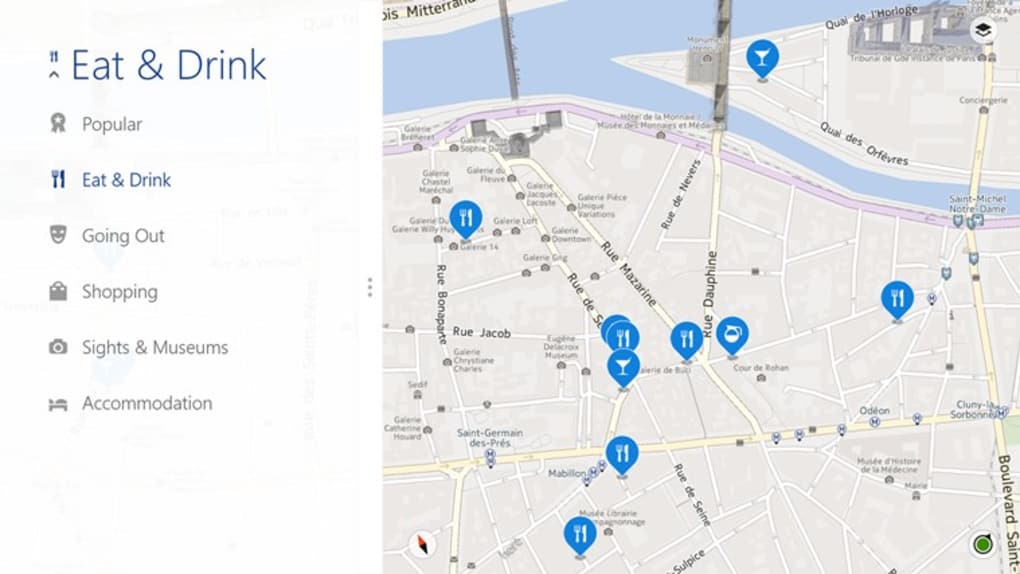


Рис. 1.9 – HERE Maps.

Важливість ознайомлення з функціональними можливостями та особливостями існуючих систем полягає у тому, що це надає розробникам важливі вказівки щодо того, як побудувати свій власний продукт. Із цього розуміння можна взяти приклади кращих практик, уникнути помилок та забезпечити оптимальну функціональність та користувацький досвід у власному програмному забезпеченні.