**Зміст**

1. Стек технологій, необхідний для реалізації додатку.
2. Написання серверної частини та підключення до БД.
3. Написання клієнтської частини додатку.
4. З’єднання клієнтської та серверної частин.
5. Висновок.
6. **Стек технологій, необхідний для реалізації додатку.**

Для того, щоб написати додаток, який буде виконувати CRUD операції з певним набором даних нам необхідні настутні технології:

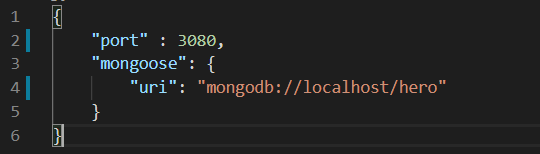
* MongoDB – для зберігання даних
* Node JS + Express.js – API, за допомогою якого ми будемо обробляти запити
* Angular – клієнтська частина, інтерфейс через який коричтувач буде виконувати необхідні дії

1. **Написання серверної частини та підключення до БД.**

У вибраній вами директорії потрібно створити файл *server.js* та відкрити його у любому зручному редакторі коду. Наступний кроком буде відкрити командну строку вашого комп’ютера і ввести в ній *cd path* , де *path* – це шлях до папки з вашим проектом, після чого написати *npm install express* – ця команда установить необхідні модулі для додатку.

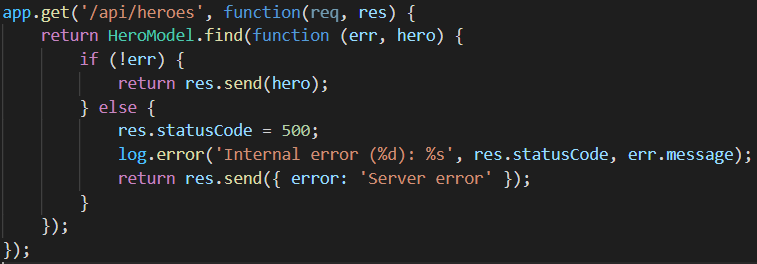
1. **З’єднання клієнтської та серверної частин.**

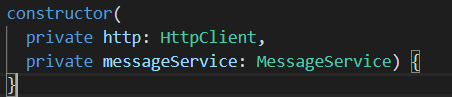
У проекті серверної частини в файлі *config.js* потрібно вказати порт за яким, буде працювати сервер.



В нашому випадку він буде 3080, це означає, що всі запити які починаютсья з <http://localhost:3080/> будуть оброблятися нашим локальним сервером.

Наступним кроком буде розписання всіх ендпоінтів для обробки запитів у файлі *server.js .* Приклад одного з ендпоінтів:



Для взаємодії з сервером з клієнтської частини та відправки запитів по протоколу http використовується клас **HttpClient.** Щоб використовувати цей клас в проекті потрібно установити пакет "@angular/common", після чого в у файлі *app.module.ts* імпортувати **HttpClientModule** із пакета "@angular/common/http". У відповідному файлі з сервісом в його конструктор потрібно передати об’єкт **HttpClient** 

і з його допомогою в методах відправляти запити на сервер. 