


Курсова робота на тему:

“Технологія GraphQL, як альтернатива REST API”

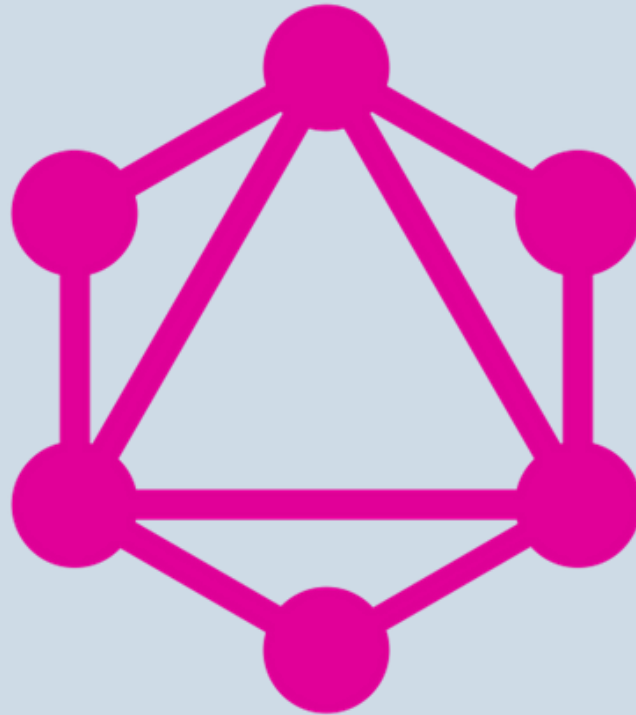
Студента 301-ї групи
Забурянного А. Г.

Науковий керівник:
доц. Матвій О. В.



Мета роботи:

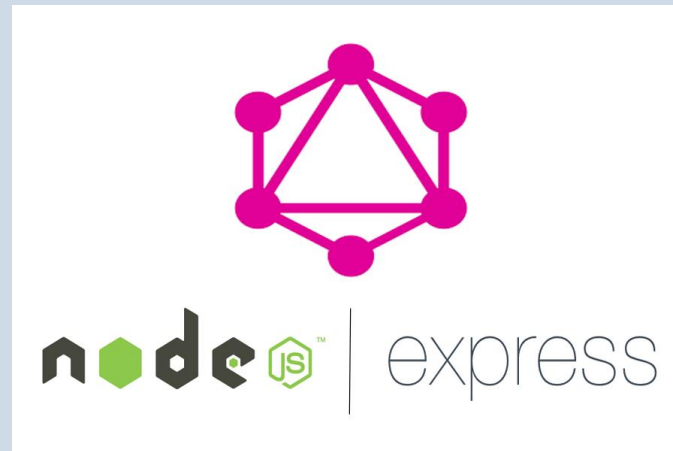
Опанувати технологію GraphQL та розробити тестовий додаток з метою демонстрації її базових можливостей.



GraphQL

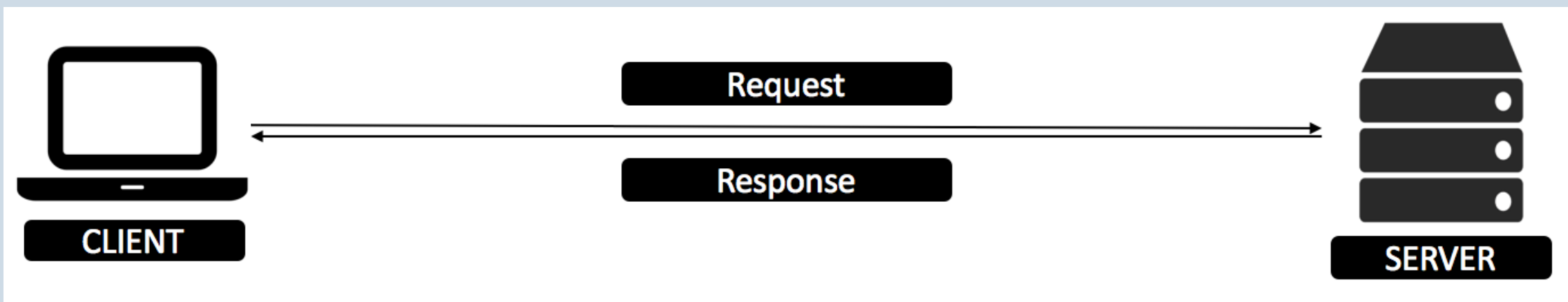
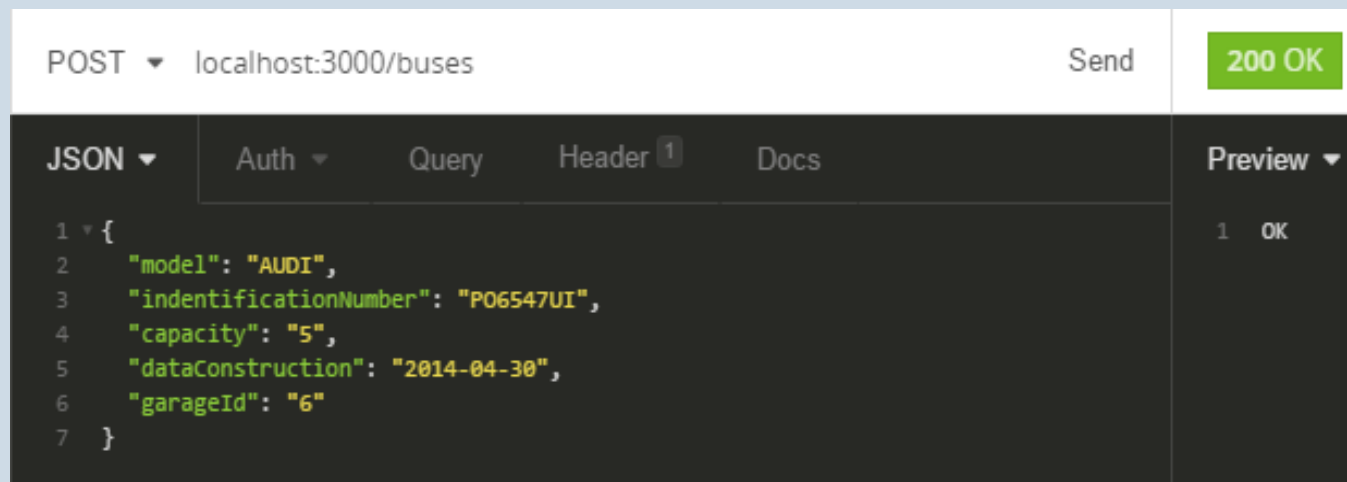
Задачі

1. Дослідити роботу архітектурного стилю REST API, його переваги та недоліки.
2. Набути базових навичок роботи з REST API.
3. Покращити навички роботи з JavaScript з використанням платформи Node.JS.
4. Розглянути основні поняття та задачі GraphQL, підходи до їх розв'язку.
5. Здійснити порівняльну характеристику REST API та GraphQL.
6. Реалізувати можливості технологій REST API та GraphQL на практиці.



Базові поняття REST API

1. Метод запиту
2. Шлях запиту
3. Тіло запиту
4. Код (статус) відповіді
5. Тіло відповіді

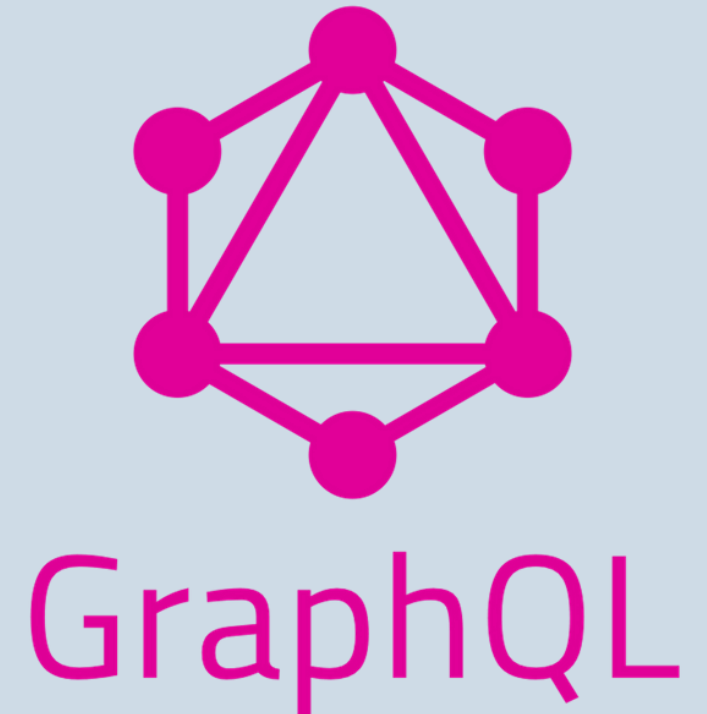


Чому ж прийшов час відмовитись від старого доброго REST API на користь GraphQL?

- До цих пір немає загального узгодження того, що таке RESTful API
- Словник REST підтримується не в повному обсязі
- RESTful API дуже складно відлагоджувати
- Як правило, RESTful API прив'язані до HTTP протоколу



vs



Базові поняття GraphQL

Запит (Query)

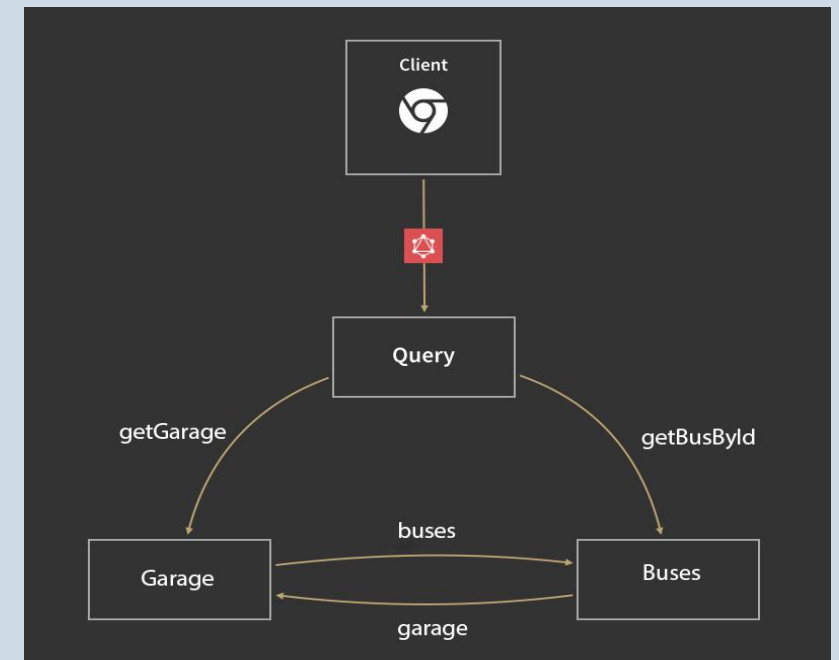
```
GraphQL ▾ Auth ▾ Query Head
1 query SpecificGarage{
2   garage(garageId:1){
3     adress
4     buses{
5       model
6       indentificationNumber
7       dataConstruction
8       capacity
9     }
10  }
11 }
```

Розпізнавач (Resolver)

```
garages: {
  type: new GraphQLList(garageType),
  resolve(root, args) {
    return new Promises(function(resolve, reject){
      mc.query('SELECT * FROM garage', function(err, result) {
        return (err ? reject(err) : resolve(result));
      });
    })
  }
}
```

Схема (Schema)

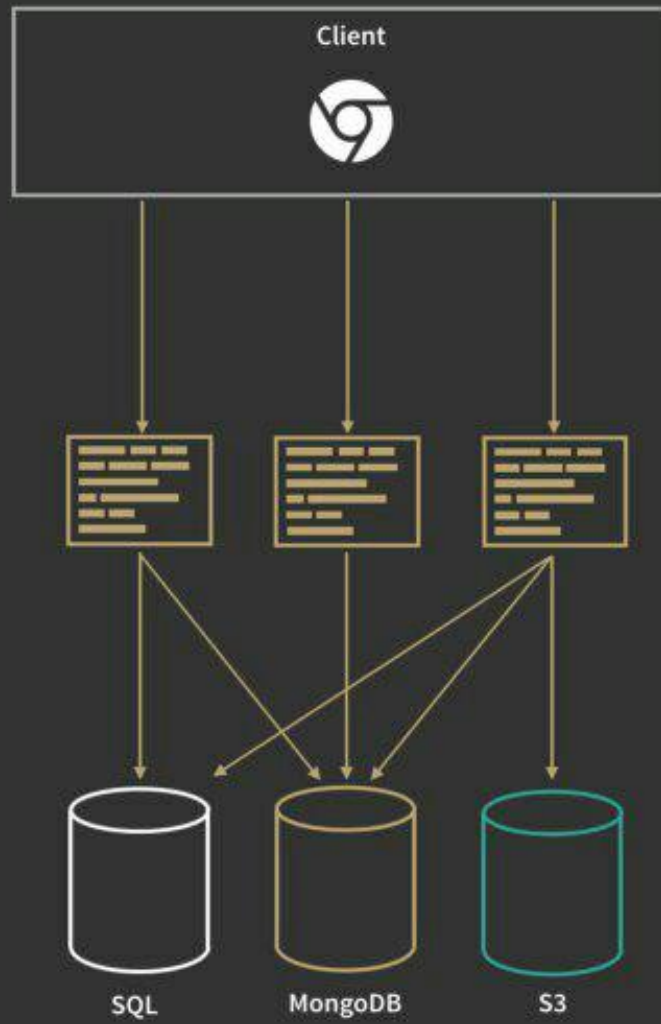
Оскільки GraphQL – строго типізований, то потрібно визначати схему кожного типу окремо, а потім поєднувати описані типи у ієрархічну структуру.



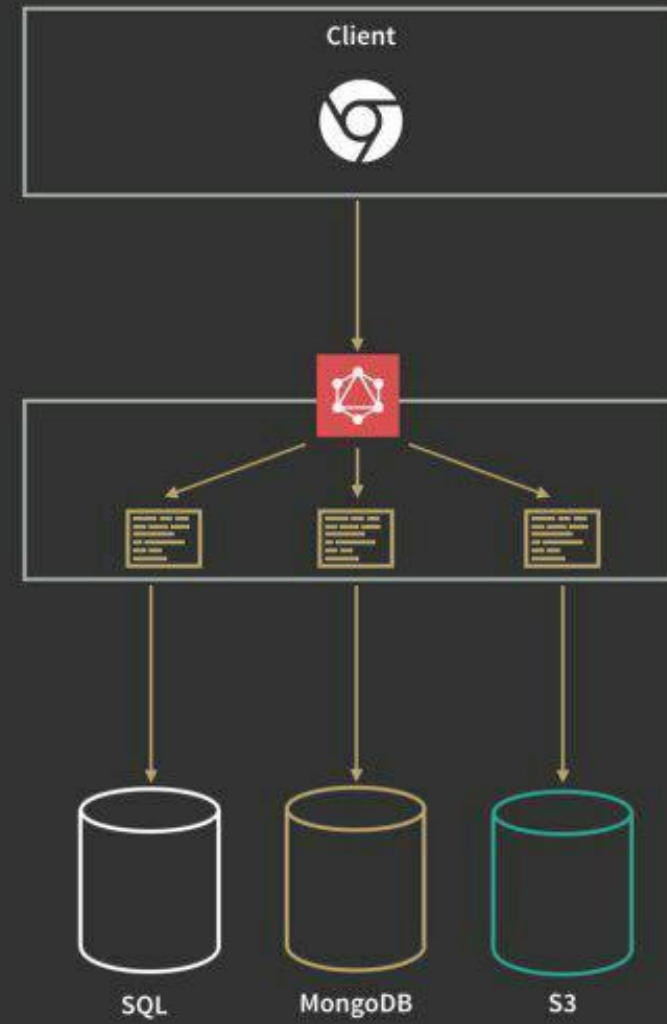
Порівняльна характеристика REST API та GraphQL

GraphQL	REST API
Поняття ресурс та способу його отримання повністю розділені одне від одного.	Ресурс та спосіб його отримання поєднані в одне ціле.
Має одну “розумну” кінцеву точку (endpoint).	Має безліч різних кінцевих точок (endpoint).
Сервер визначає лише набір доступних ресурсів, а клієнт вказує потрібні йому дані в запиті.	Структура і об’єм ресурсу визначаються сервером.
Для запису даних змінюється ключове слово в тілі GraphQL запиту. (Наприклад з query на mutation)	Запис даних визначається зміною імені HTTP-методу запиту. (Наприклад з GET на POST)
Можливість отримати всі необхідні дані за один запит за рахунок вкладеності полів.	Через те, що об’єм ресурсу визначається на стороні сервера клієнт може отримати, як неповні дані, так і надлишкову інформацію.
Відповідь на запит може повертати дані у форматі: JSON.	Відповідь на запит може повертати дані в різних форматах: JSON, HTML, XML.

REST



GRAPHQL



= Arbitrary code

Створення нового елементу бази за допомогою REST API

POST localhost:3000/buses Send 200 OK TIME 125 ms

JSON Auth Query Header 1 Docs Preview Header 8

```
1 {
2   "model": "AUDI",
3   "identificationNumber": "P06547UI",
4   "capacity": "5",
5   "dataConstruction": "2014-04-30",
6   "garageId": "6"
7 }
```

1 OK

Створення нового елементу бази даних за допомогою GraphQL

POST localhost:3000/graphql Send 200 OK TIME 125 ms SIZE 169 B

GraphQL Auth Query Header 1 Docs Preview Header 5 Cookie Tim

```
1 mutation CreateBus{
2   createNewBus
3   (
4     model:"LEXUS",
5     identificationNumber:"MP1234XZ",
6     capacity:5,
7     dataConstruction:"2017-12-21",
8     garageId:8
9   )
10 {
11   busId
12   identificationNumber
13   dataConstruction
14   garageId
15 }
16 }
```

```
1 {
2   "data": {
3     "createNewBus": {
4       "busId": 54,
5       "identificationNumber": "MP1234XZ",
6       "dataConstruction": "2017-12-20",
7       "garageId": 8
8     }
9   }
10 }
```

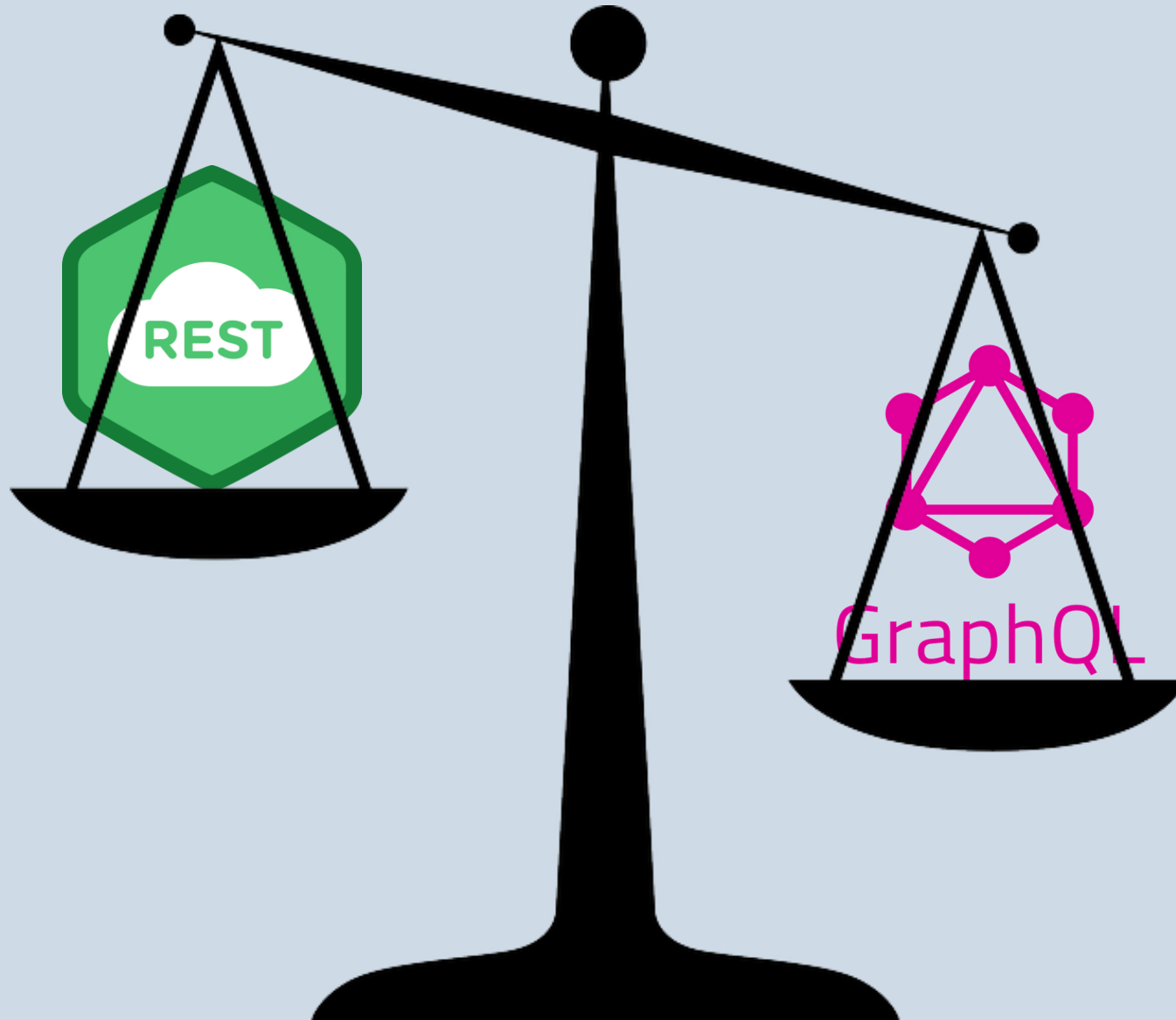
Висновки

Деякі з відмінностей між технологіями REST API та GraphQL – говорять на користь GraphQL. Досить зручно мати власний API, як набір невеликих функцій-розпізнавачів, які передбачуваним способом дістають всю необхідну інформацію за один запит.

Було виконано:

- Розроблено додаток з використанням технології REST.
- Створено HTTP-запити: GET, POST, PUT, PATCH, DELETE.
- Розроблено додаток з використанням технології GraphQL.
- Створено запити на отримання інформації та GraphQL мутації для створення, оновлення та видалення інформації з БД.
- Опановано фреймворки: express, graphql, bluebird.
- Досліджено роботу технологій REST API та GraphQL, проведено порівняльну характеристику.

Що ж краще використовувати на вашу думку REST API чи GraphQL?



Дякую за увагу!