

Аналіз та обґрунтування вибору технології розробки застосунку

Фронтенд

На даний момент, є три найпопулярніші JavaScript-фреймворки для створення користувацьких інтерфейсів – Vue.js, React.js та Angular. Кожен з них має свої переваги та недоліки. Розглянемо їх більш детально.

JavaScript-фреймворк – це інструмент, який використовується для побудови користувацьких інтерфейсів, динамічних веб-, мобільних, десктопних додатків за допомогою мови програмування JavaScript. JavaScript – динамічна, об'єктно-орієнтована прототипна мова програмування.

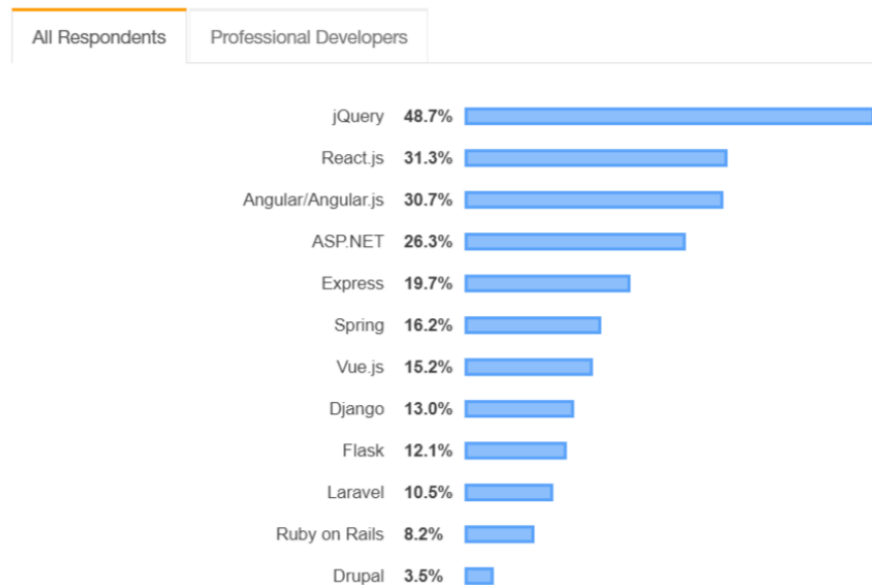
Фреймворки полегшують розробку продукту, оскільки пропонують функції та можливості, що вже реалізовані та можуть бути використані під час розробки продукту, і розробнику не потрібно додатково використовувати час на пошук та вирішення. Це лише одна з переваг використання фреймворків. Є також багато інших переваг:

- швидка розробка;
- код продукту стає більш структурованим і більш легшим в розумінні;
- код стає меншим;
- розробка продукту стає більш гнучкою;
- спрощує розробку продукту для декількох розробників одночасно.

Проте кожен фреймворк має свої переваги над іншими. Розглянемо переваги трьох найпопулярніших JavaScript-фреймворків – Vue.js, React.js та Angular, а також їх популярність серед розробників.

За даними StackOwerflow:

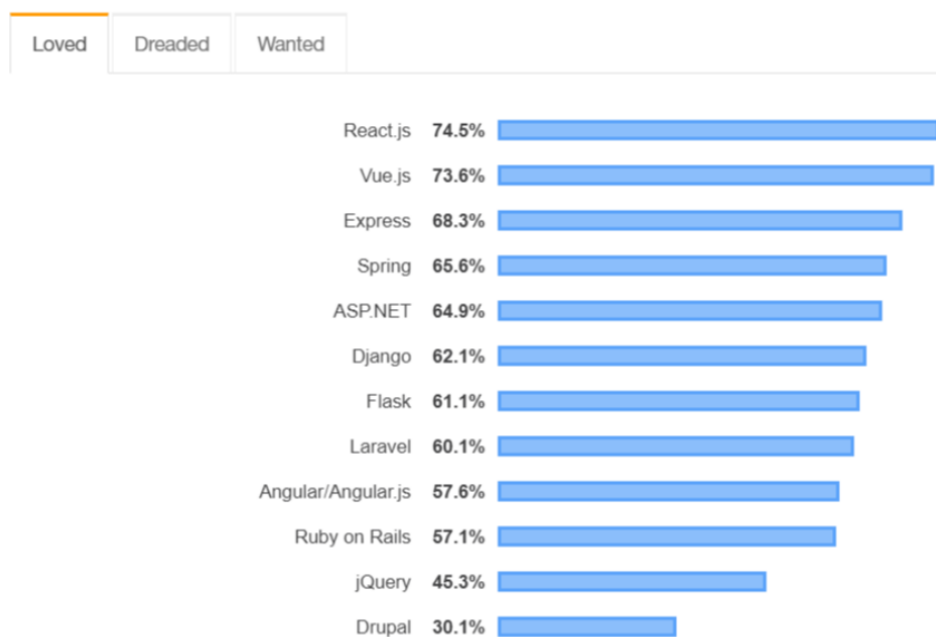
Web Frameworks



Незважаючи на стрімкий розвиток Vue.js, цей фреймворк посідає лише 7 місце в рейтингу (як серед усіх респондентів, так і серед професіоналів). React.js та Angular.js поділили 2-е та 3-е місце.

Примітно, що згідно із загальним результатом 2-е місце посідає React.js, але професійні розробники вивели на 2-е місце саме Angular.js.

Most Loved, Dreaded, and Wanted Web Frameworks



У той же час, розробникам найбільше подобається працювати саме з React.js та Vue.js, коли Angular.js виявився практично наприкінці списку. Серед тих, хто раніше не розробляв програми за допомогою цих фреймворків, найбажанішим для вивчення виявився React.js. За ним слідує Vue.js, а третє місце посідає Angular.js.

State of JavaScript - щорічний звіт, який є головним показником розвитку JavaScript в цілому та всіх інструментів, які з ним пов'язані. За його даними маємо наступну картину:



Переважно, розробники задоволені робочим процесом із React.js та Vue.js, у той час як Angular.js задовольняє їх лише на 38%.

Тепер варто розібрати окремо кожен фреймворк.

React.js вже третій рік поспіль займає перший рядок усіх рейтингів. Деякі розробники зовсім не мають нарікань у бік цього фреймворку, оскільки він швидко розвивається і стає більш стабільним.

<i>Pros</i>	<i>Cons</i>
The biggest community of developers	It's necessary to learn JSX to use it to the utmost
The support that is provided by Facebook	Low-quality documentation (developers often write it themselves)
Saturated ecosystem	Steep learning curve
Can be used for native and web apps	Too many sudden updates

React.js – дуже насичене середовище розробки. Здебільшого користувачі відгукуються про неї тільки з позитивного боку.

Однак останнім часом Facebook випускає дуже багато оновлень, через що старіють деякі інструменти. Деяким розробникам такий перебіг подій подобається. Але здебільшого, спільнота висловлює невдоволення з цього приводу. Адже в такий спосіб їм необхідно постійно освоювати нові методи.

Примітно, що сьогодні фахівці React.js є найбільш затребуваними в США. Станом на початок травня 2020 року, у LinkedIn активно понад 30000 відкритих вакансій на запит React.js, на Indeed – близько 4000.

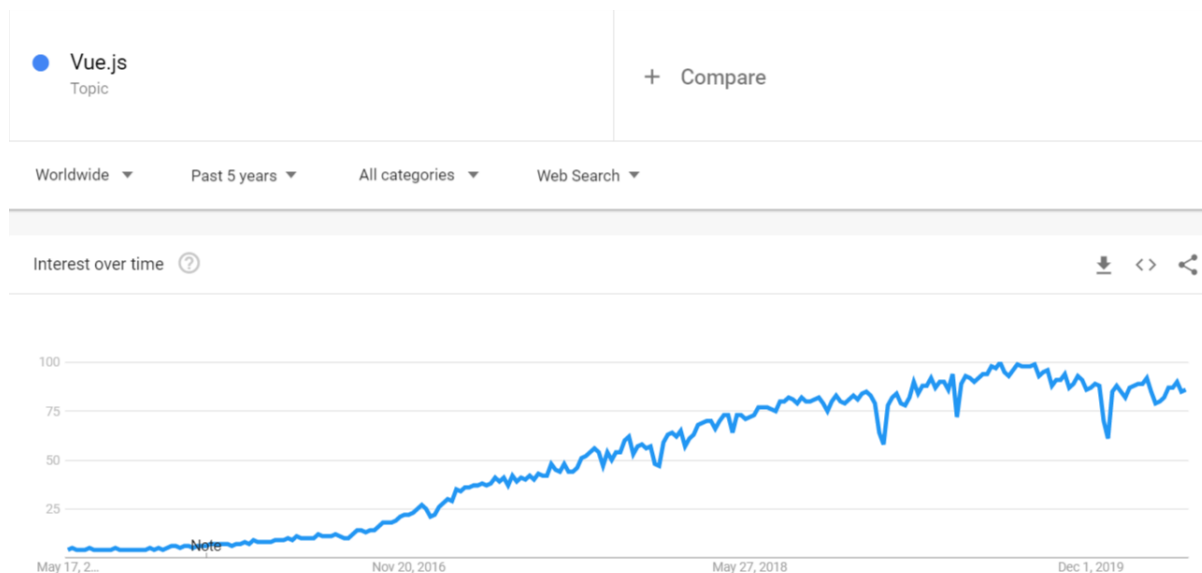
Одна з ключових причин популярності React.js – підтримка авторитетної компанії Facebook. На базі цього середовища розробки сьогодні працюють такі сервіси, як Instagram, Whatsapp та Twitter. Усі вони дуже швидкі, привабливі зовні. Це забезпечило високий рівень довіри React.js.

Незважаючи на численні переваги, є лише кілька випадків, коли це середовище розробки буде найбільш підходящим:

- швидка розробка легких програм корпоративного рівня;
- створення SPA або кросплатформових додатків;
- розширення функціоналу наявного додатка.

Це не означає, що React.js є поганим в інших випадках. Швидше альтернативні варіанти будуть кращими.

Vue.js став відкриттям останніх років. Несподівано, з рядового середовища розробки він став однією з найулюбленіших серед професіоналів.



Цікаво, що популярність зростає без підтримки великих компаній. Гіганти ринку звернули увагу на Vue.js лише після різкого стрибка. Саме тому він виявився третім фреймворком у тривалому протистоянні React.js та Angular.js.

<i>Pros</i>	<i>Cons</i>
The tiny size of the framework	Lack of support for large-scale projects
Two-way data binding	Lack of resources
Very easy to learn	Problems in the work of apps on iOS and in Safari
Circumstantial and detailed documentation	Small community

Vue.js зараз виділяється тим, що він має безліч особливостей. Деякі навіть можуть виступати як недоліки. Наприклад, Vue.js дуже гнучкий. Але у разі роботи над великими проектами у великій команді це може сприяти більшій кількості помилок.

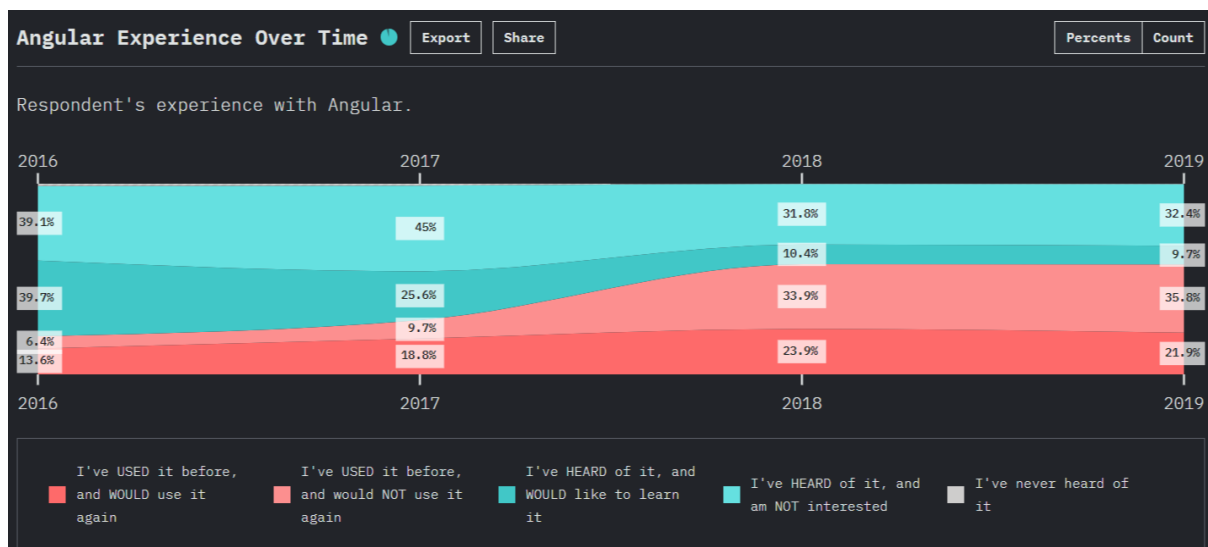
Після того, як Vue.js почав виявляти свої особливості, на нього звернули увагу багато гігантів ринку, серед яких Gitlab, WizzAir, EuroNews. Grammarly повністю створено на базі Vue, а Alibaba та Xiaomi у 2018 році оголосили про повний перехід на Vue.js.

Крім того, сьогодні фреймворк має великий попит на азіатському ринку.

У яких випадках Vue.js буде кращим вибором?

- для розробки «розумних» та високопродуктивних додатків;
- для раннього виходу програми на ринок;
- створення невеликих та легких додатків на зразок Grammarly.

Angular.js довгий час був найкращим вибором для розробки інтерфейсу користувача. Так він став частиною популярного стека MEAN.



Проте за останні кілька років розробники все більше скаржилися на неповноцінність фреймворку порівняно з конкурентами. Як результат, багато компаній поступово відмовляються від використання Angular.js.

Pros	Cons
Two-way data binding	It's bad for creating the SPA, which are trending today
Directives that permit creating dynamic and rich content with HTML	Steep learning curve
Dependency injection	Constrained functionality for SEO
Big community	Poor documentation compared to the competitors

Багато в чому фреймворк втратив свою популярність через появу нових трендів веб-розробки. Команда Angular.js не реалізувала необхідний

функціонал у нових версіях середовища розробки. Тому сьогодні ми спостерігаємо, що Vue.js і React.js стають кращим вибір. Якщо виходити за рамки цих трьох фреймворків, то angular іноді поступається іншим інструментам (наприклад, Svelte).

Тим не менш, Angular.js все ще використовується для підтримки багатьох популярних сайтів та веб-застосунків. Серед них опинилися The Guardian, UpWork, PayPal та Sony. Всі вони є великими сайтами, в яких Angular.js виявляє себе досить добре.

У яких випадках слід звернути увагу на Angular.js?

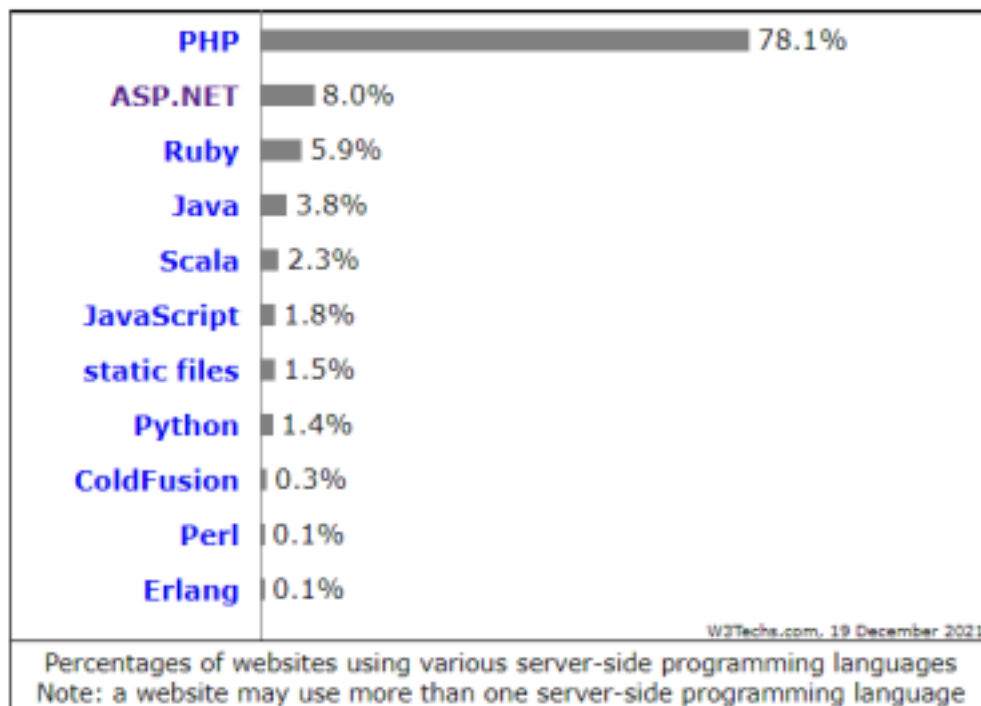
- створення масштабних програм;
- необхідність наявності легко-масштабованої архітектури;
- створення месенджерів та інших програм «в реальному часі»;

Виходячи зі статистики, результатів опитувань та різноманітних звітів можна зробити висновок, що найбільш кращий вибір у 2022 – це **React.js**. Друге місце займає Vue.js, а Angular.js залишиться на третьому із великим відставанням.

Бекенд

Оскільки я маю створити надійну систему для роботи модельного агентства, варто використовувати систему ASP.Net. Вона має як переваги, так і недоліки. Почну з недоліків:

По-перше, це не найпопулярніша технологія. Надаю рейтинг найкращих мов програмування для створення серверу:



Джерело:

https://w3techs.com/technologies/overview/programming_language

Перше місце належить PHP. На мою думку, цілком заслужена позиція. Але якщо говорити про лендінги, блоги та соціальні мережі. По-друге, вартість серверів не тішить. Як ми вже вияснили, C# не така поширена мова для бекенду, через це купівля хостингу буде коштувати дорого. І останнє, четверте: мова не достатньо оптимізована, тому варто звернути на це увагу та застосовувати всі можливі методи для покращення роботи.

Тепер я хочу розповісти про переваги ASP.

Перше: коли я згадувала про недоліки мови з приводу популярності, я згадувала, що PHP гарно підходить для невеликих сайтів. А C# зазвичай використовується для написання складних та гарно захищених систем, що потребують постійної підтримки. Приклади: <https://www.microsoft.com/uk-ua/>, <https://www.dell.com/ru-ua>. У випадку з сайтом для модельного агенства, краще не ризикувати. Навіть якщо я оберу PHP, щоб забезпечити сайт необхідним захистом, потрібно буде витратити багато часу.

Друге: сам по собі код C# можна легко зрозуміти. Тому у мене не буде зайвих запитань з приводу тих чи інших структур, що знову ж таки прискорить час реалізації. Типізація мови та об'єктно-орієнтованість в даному випадку грає на руку.

Третій плюс: широкі можливості. З ASP.Net Core я зможу реалізувати майже будь-яку забаганку замовника. Microsoft подбали про те, щоб технологія була гнучкою та легко розширюваною.

І останній, четвертий плюс, ASP.Net Core надає гнучкі бібліотеки для обробки з даними, так як Identity Core (для ведення облікових записів) та Entity Framework (для керування базою даних).

Середовища розробки для обраних технологій:

Технологія	Середовище розробки
React.js	Visual Studio Code
ASP.Net Core	Visual Studio
MSSQL	MSSQLStudio

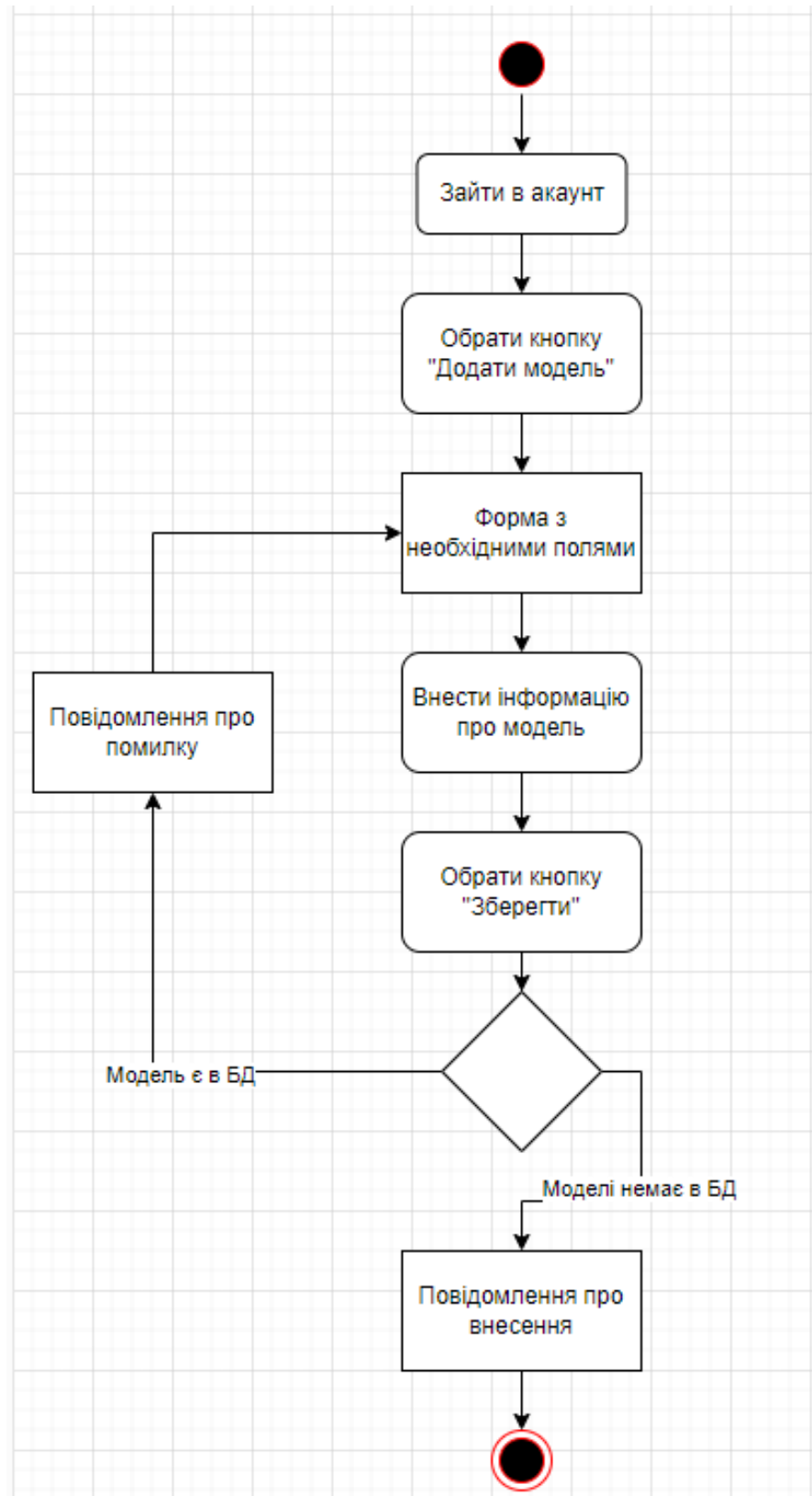
Структура веб-запиту складається з трьох частин: стартовий рядок, заголовки, тіло повідомлення, що містить дані запиту, запитуваний ресурс або опис проблеми, якщо запит не виконано.

В моєму додатку запити формуються на фронтенд частині за допомогою бібліотеки axios та мають такий загальний вигляд:

```
get: <T>(url: string) => axios.get<T>(url).then(responseBody)
post: <T>(url: string, body: {}) => axios.post<T>(url,
body).then(responseBody)
put: <T>(url: string, body: {}) => axios.put<T>(url,
body).then(responseBody)
delete: <T>(url: string) =>
axios.delete<T>(url).then(responseBody)
```

Розробка структури інформаційної системи

Діаграма діяльності:



Діаграма класів:

