

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

ПАРАСОЧКІН ВІТАЛІЙ ВОЛОДИМИРОВИЧ

Допускається до захисту:

завідувач кафедри
інформаційних технологій,
доктор технічних наук, доцент

_____ Т. В. Нескородева

« ____ » _____ 20__ р.

РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ ВИВЧЕННЯ АНГЛІЙСЬКОЇ МОВИ

Спеціальність 122 «Комп'ютерні науки»

Кваліфікаційна (бакалаврська) робота

Керівник:

Антонов Ю.С., доцент кафедри
інформаційних технологій,
к.ф—м.н., доцент

Оцінка:

____ / ____ / ____

(бали за шкалою ЄКТС/за національною шкалою)

Голова

ЕК:

(підпис)

АНОТАЦІЯ

Парасочкін В.В. Розробка веб-додатку для вивчення англійської мови.
Спеціальність 122 «Комп'ютерні науки», освітня програма «Сучасні інформаційні технології та програмування». Донецький національний університет імені Василя Стуса, Вінниця, 2022.

Кваліфікаційна (бакалаврська) робота присвячена розробці веб-додатку для вивчення англійської мови. За допомогою технологій HTML та CSS, мови програмування JavaScript, бібліотеки React та бібліотеки Redux розроблено веб-додаток, який допоможе користувачам вивчати англійську мову.

Ключові слова: веб-додаток, англійська мова, веб-розробка, HTML, CSS, мова програмування JavaScript, бібліотека React, бібліотека Redux.

Рис. 25, Табл. 2, Джерел 39.

Parasochkin V.V. Development of a web application for learning English.
Specialty 122 "Computer Science", educational program "Modern Information Technology and Programming". Vasyl Stus Donetsk National University, Vinnytsia, 2022.

Qualifying (bachelor's) work is devoted to the development of a web application for learning English. Using HTML and CSS technologies, JavaScript programming languages, the React library and the Redux library, a web application has been developed to help users learn English.

Keywords: web application, English language, web development, HTML, CSS, JavaScript programming language, React library, Redux library.

Fig. 25, Table. 2, Sources 39.

ЗМІСТ

ВСТУП.....	4
РОЗДІЛ I	7
ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ	7
1.1 Аналіз обраної предметної області.....	7
1.2 Розгляд існуючих аналогів.	8
Висновок до розділу 1.....	14
РОЗДІЛ II	15
ОСНОВНІ ТЕОРЕТИЧНІ ПОЛОЖЕННЯ ТА ПОСТАНОВКА ЗАДАЧІ.....	15
2.1 Визначення загальних процесів роботи додатку.....	15
2.2 Визначення ключових моментів	16
2.3 Постановка задачі.....	17
Висновок до розділу 2.....	18
РОЗДІЛ III	19
РЕАЛІЗАЦІЯ ВЕБ ДОДАТКУ ТА ОПИС ВИКОРИСТАНИХ ТЕХНОЛОГІЙ	19
3.1 Вибір мови програмування.....	19
3.2 Вибір веб-фреймворку	23
3.3 Вибір середовища розробки IDE	32
3.4 Засіб для хостингу проекту.	33
3.5 Реалізація веб-додатку.	35
Висновок до розділу 3.....	46
ВИСНОВКИ.....	47
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	48

ВСТУП

Ні для кого не секрет, що англійська мова в даний момент розвитку світу є найпопулярнішою та найпотрібнішою мовою в світі. Англійська мова є офіційною мовою 59 суверенних країн, що робить її найпопулярнішою офіційною мовою у світі. Такі країни як Канада, США, Великобританія і Австралія є англomовними.

Англійською мовою говорять більше 1-го мільярду населення всієї земною кулі [1] . А це приблизно кожна восьма людина на Землі. Підприємці по всьому світу використовують її для підписання певних договорів та угод. Світові змагання такі як чемпіонат Європи з футболу або ж Олімпійські ігри вибрали офіційною мовою саме англійську. Та що там говорити, саме англійська мова використовується представниками різних країн в міжнародних перемовинах та підписанні документів.

Також не можна сперечатись з тим фактом, що англійська мова найпотрібніша для вивчення різноманітної інформації в всесвітній павутині. Найбільше корисної інформації в інтернеті знаходиться саме на англійській мові. На англійській мові викладають в найпрестижніших вузах світу, таких як Гарвардський університет, Оксфордський університет тощо. Тому завдяки знанням цієї мови, можна поїхати навчатись саме в ці вузи світу.

Оскільки більшість країн світу говорить англійською мовою, то для подорожей не тільки країнами Європи, а й по всьому світу, то з'являється ще одна причина знати англійську мову.

В цьому плані, є велика кількість людей в нашій країні, яка на доволі хорошому рівні знає та спілкується англійською мовою. Тому англійська мова в Україні популярна: вона викладається в школах і в деяких спеціальностей в університетах та коледжах, інколи навіть в дитячих садочках.

На сьогоднішній день, в реаліях нашого світу і з огляду на те що було сказано вище – знати англійську мову є звичайною необхідністю для кожної

грамотної людини та хорошого фахівця в деяких галузях роботи. Однак більшість людей, які вивчали англійську, можуть з впевненістю сказати, що процес вивчення мови зовсім не простий. Цей процес вивчення мови потребує багато сил, енергії, зусиль, терпіння, а здебільшого і коштів. Це довгий і повільний процес, але так необхідний людині.

Тому постає логічне питання, як людині в сучасному світі вивчити англійську?

Але в нашій час, в час розвитку технологій кожна людина, яка справді хоче вивчити англійську, мову може це запросто зробити навіть вдома. Для цієї мети наразі є різні онлайн сервіси і способи, такі як, спеціальні сайти, онлайн-репетитори, додатки на мобільний телефон та інші. Але кожен з цих сервісів має певну кількість недоліків, що робить процес вивчення англійської мови не продуктивним, повільним та нудним. Іноді деякі з цих причин та чинників взагалі ставлять хрест на вивченні англійської мови.

Для людей, які в вивченні англійської мови потребують живого спілкування з викладачами або іншими студентами існує безліч різноманітних курсів, вебінарів та тренінгів. Але нажаль, більшість дійсно хороших таких способів вивчення є достатньо дорогими, що може сіяти відсторонення від цієї справи для деяких учнів.

Можна сказати, що пощастило, що дуже швидко зростає потреба в знанні англійської мови в нашій країні, і через це вивчення англійської вже є обов'язковим предметом навіть в університетах та школах. Однак переважно велика кількість викладачів та і студентів погодиться з тим, що важко викладати, а більшість студентів погодиться, що важко розуміти коли в групі є 15-20 або і більше студентів.

В такому випадку в студента може взагалі зникнути бажання вивчати англійську мову, оскільки в викладача просто немає можливості донести матеріал до кожного студента групи.

Саме через таку велику необхідність та проблеми вивчення іноземної мови мною було поставлена задача розробити веб-додаток, тобто сайт, який допоможе всім охочим в вивченні англійської мови. Також це вирішить проблему складного, нудного та не індивідуального підходу до вивчення в університетах та інших навчальних закладах. Також за допомогою цього веб-додатку можна буде вивчати англійську вдома, що є необхідним в нашій час.

Мета бакалаврської роботи: вивчення методики створення веб додатків за допомогою мови програмування JavaScript та бібліотеки React.

Завдання дослідження:

1. Огляд існуючих веб-додатків за даною тематикою. Виявлення їх недоліків та переваг.
2. Знайомство з інструментами розробки за допомогою яких можливе створення веб додатку.
3. Розробка веб-додатку для вивчення англійською мови.

Об'єктом дослідження є проектування та розробка веб-додатку за допомогою мови програмування JavaScript та бібліотеки React.

Предметом дослідження є процес розробки веб-додатку для вивчення англійської мови, з використанням мови програмування JavaScript, бібліотеки React та інших потрібних інструментів.

Зв'язок роботи з науковими програмами, планами, темами. Наведені у бакалаврській роботі дослідження пов'язані з фундаментальною науково-дослідною роботою «Дослідження та комп'ютерно-математичне моделювання складних систем та процесів у науці, освіті та інформаційно-комунікаційній діяльності підприємств» (№ держреєстрації 0116U002394, 2018-2022 рр.).

Робота складається зі змісту, вступу, 3-х розділів зі своїми підрозділами, які в свою чергу мають 25 рисунків та 2 таблиці і висновки, а також із загального висновку до курсової роботи та списку використаних джерел в кількості 39 найменувань. Загальний об'єм роботи 53 сторінок.

РОЗДІЛ І

ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз обраної предметної області

Для початку, щоб розуміти що таке веб-додаток, слід дати визначення цьому терміну. Веб-додаток – це будь-яка комп’ютерна програма, яка може виконувати певні функції, використовуючи безпосередньо веб-браузер як свого клієнта [2].

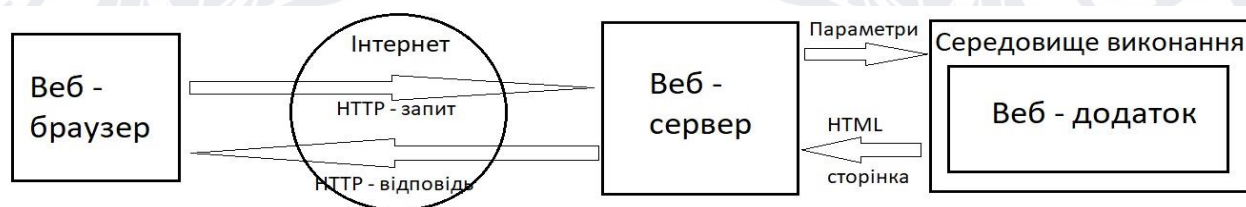


Рисунок 1.1 – Загальна схема взаємодії користувача з веб-додатком

Додаток може бути зовсім простим як наприклад дошка оголошень або програма для списку задач, в той час додаток може бути досить складним, як наприклад мобільна ігрова програма, яку кожна людина може завантажити на свій телефон.

Варто розуміти, що на даному етапі розвитку технологій, вони беруть активну участь в нашому повсякденному житті. Веб-додаток може бути вбудовано в веб-сайт, або сам веб-сайт може бути веб-додатком. Наприклад, Facebook [3] та Instagram [4] це веб-сайти зі дуже значними реалізаціями веб-додатків.

Веб-додаток може звільняти розробника безпосередньо від відповідальності за створення клієнта для певного типу операційної системи, тому кожен може користуватися додатком, якщо користувач має доступ до звичайного інтернету. Оскільки клієнт безпосередньо працює у веб-браузері, виходячи з цього, користувач використовує персональний комп’ютер чи мобільний телефон. Вони можуть використовувати такі веб-браузери як Internet

Explorer [5], Chrome [6] або Firefox [7]. У веб-додатках зазвичай використовується комбінація сценарію на стороні сервера та сценарію на стороні клієнта. Клієнтський сценарій займається поданням інформації, тоді як серверний сценарій займається всіма складними справами, такими як зберігання та отримання інформації.

В більшості веб-додатків є можливість переходити між вкладками без перезавантаження сторінки, що прискорює роботу веб-додатку в цілому.

На перший погляд можна легко сплутати між собою веб-додаток та веб-сайт. Однак насправді виділяють такі розбіжності між ними:

- Веб-сайт – це просто зв’язана між собою певна кількість HTML файлів, доступних через мережу. В той час як веб-додаток – це комп’ютерний додаток що постачається через мережу.
- Веб-додаток може бути частиною веб-сайту або навіть окремим додатком.
- Веб-сайти служать для користувача лише для доставки певної інформації, як музика, відео або звичайний текст. В той час як веб-додаток може взаємодіяти з користувачем і давати певні відповіді на основі дій користувача.
- Веб-додаток взаємодія з користувачем, а веб-сайт – ні.
- Веб-додаток здебільшого підключається до бази даних.

1.2 Розгляд існуючих аналогів.

Оскільки, тема веб-додатків дуже сильно розвивається в наш час, і в сукупності з сильною потребою вивчати англійську мову, мені не знадобилось багато зусиль, щоб знайти будь-які веб-додатки для вивчення англійської мови.

Відвідавши посилання [8] , де зібрано ТОП-20 безкоштовних та найпопулярніших веб-сервісів, для вивчення англійської мови, я можу з легкістю виокремити, які на мою думку є найкращими та найбільше підходять для людей з різними рівнями знань англійської:

Один з найпопулярніших веб-сервісів, та той яким я особисто користувався певний період часу має назву **Duolingo** [9].

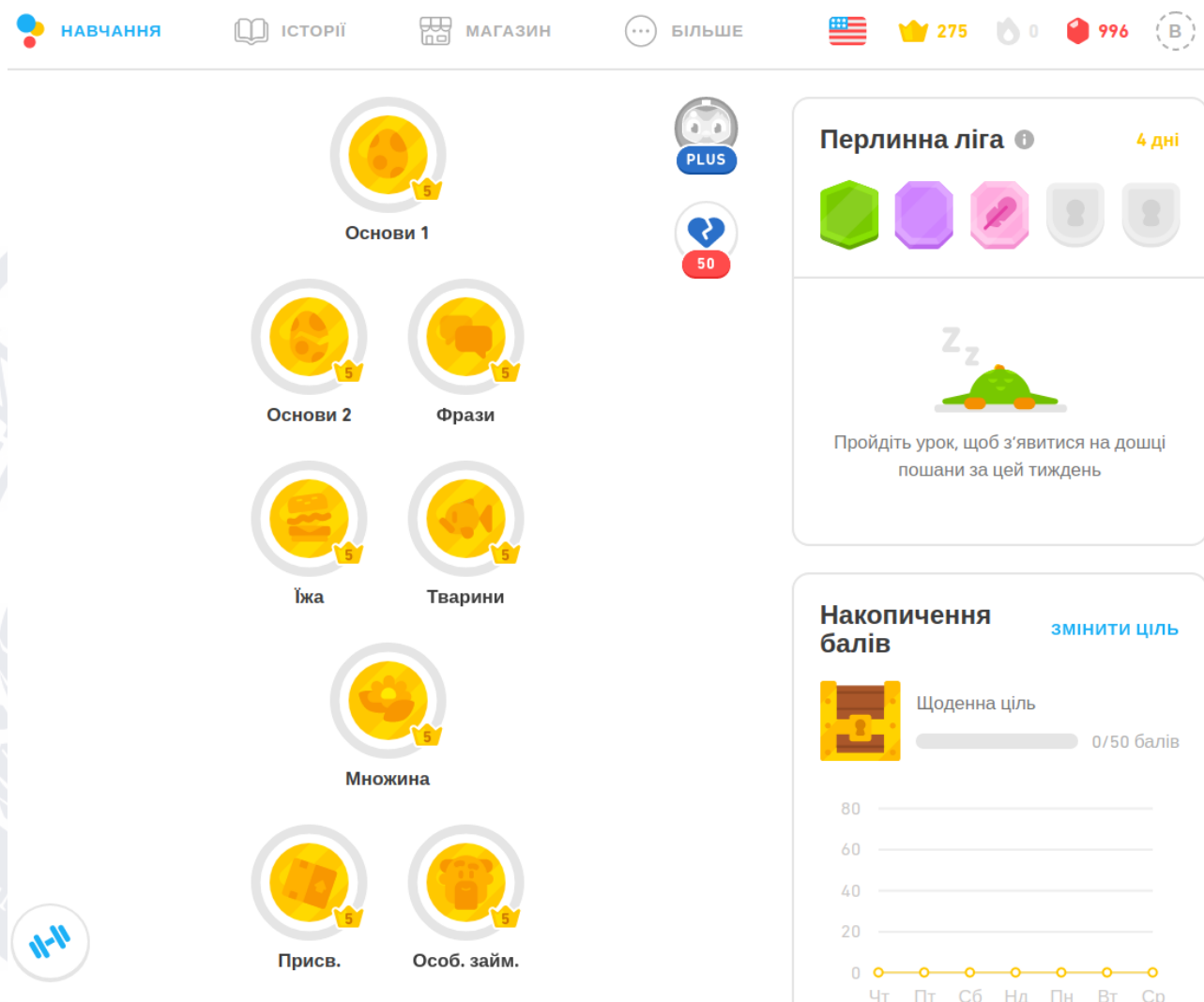


Рисунок 1.2 – Головний екран Duolingo

Даний сайт функціонує та навчає всіх бажаючих за типом «дерево досягнень»: тобто щоб перейти на новий рівень, слід виконати всі завдання на попередньому рівні, або набрати мінімальну кількість очок, які дають можливість перейти на новий рівень. В сервісу навіть є доступні мобільні додатки на Android та iOS, що дає змогу навчатись і за ПК і за телефоном, не втрачаючи при цьому прогрес вивчення обраної мови. Адже слід зазначити, що дана платформа дозволяє вивчати не тільки англійську мову, а й інші.

LearnEnglish [10] – це платформа не зовсім схожа на попередній приклад. На даному сайті зібрані різноманітні матеріали для вивчення англійської мови в

різних форматах та різними способами, як ось наприклад в уроках, іграх, спілкуванні в чаті та інше. Сайт доступний англійською мовою.

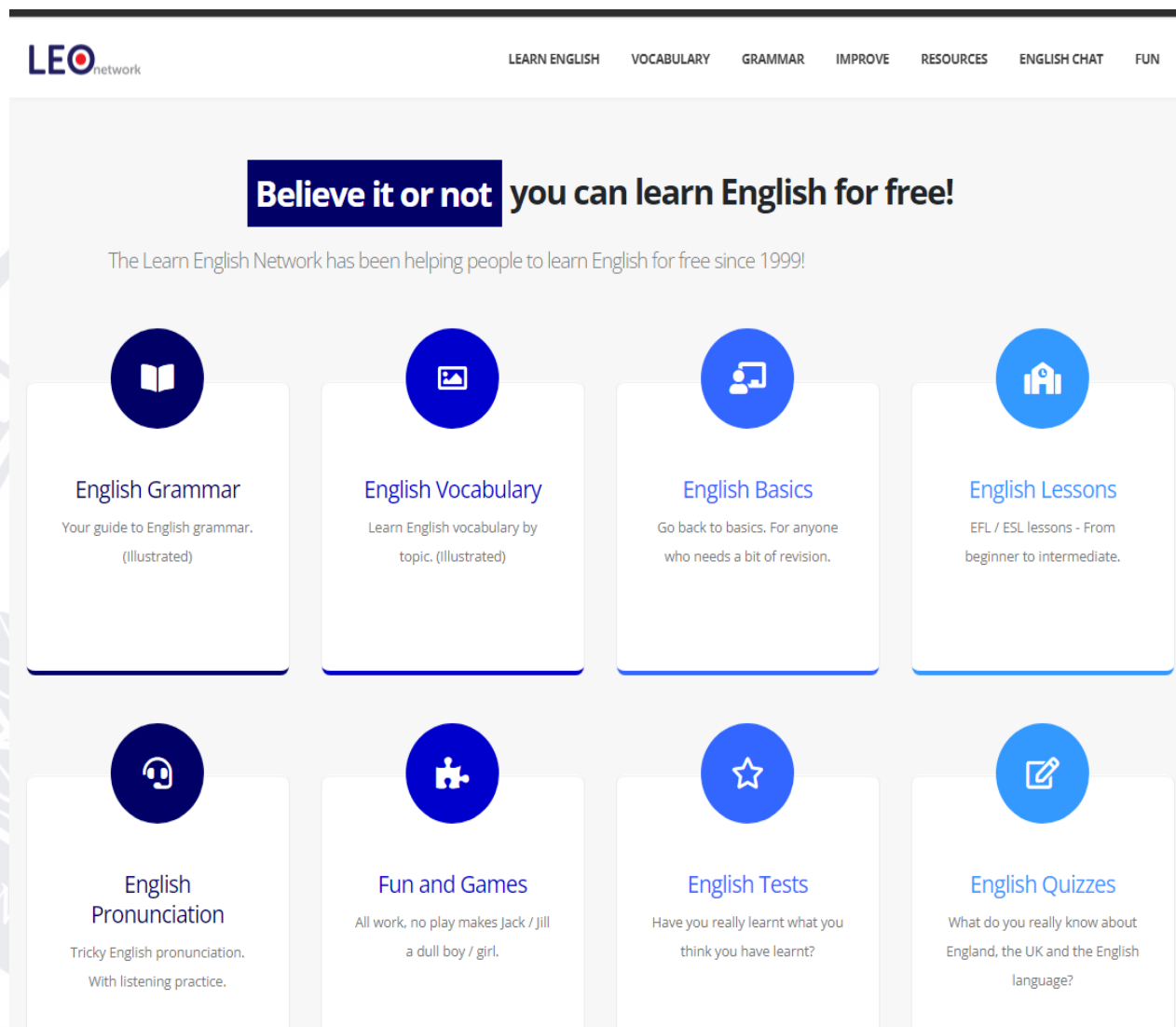


Рисунок 1.3 – Головний екран LearnEnglish

Situational English [11] – Веб-сайт спрямований для вивчення англійської мови через різноманітні ситуації. На сайті зібрано більше 200 статей, в яких, в залежності, від контексту, пропонуються готові вираження, реакції і відповідно переклад. На жаль, на сайті немає української мови, лише російська.

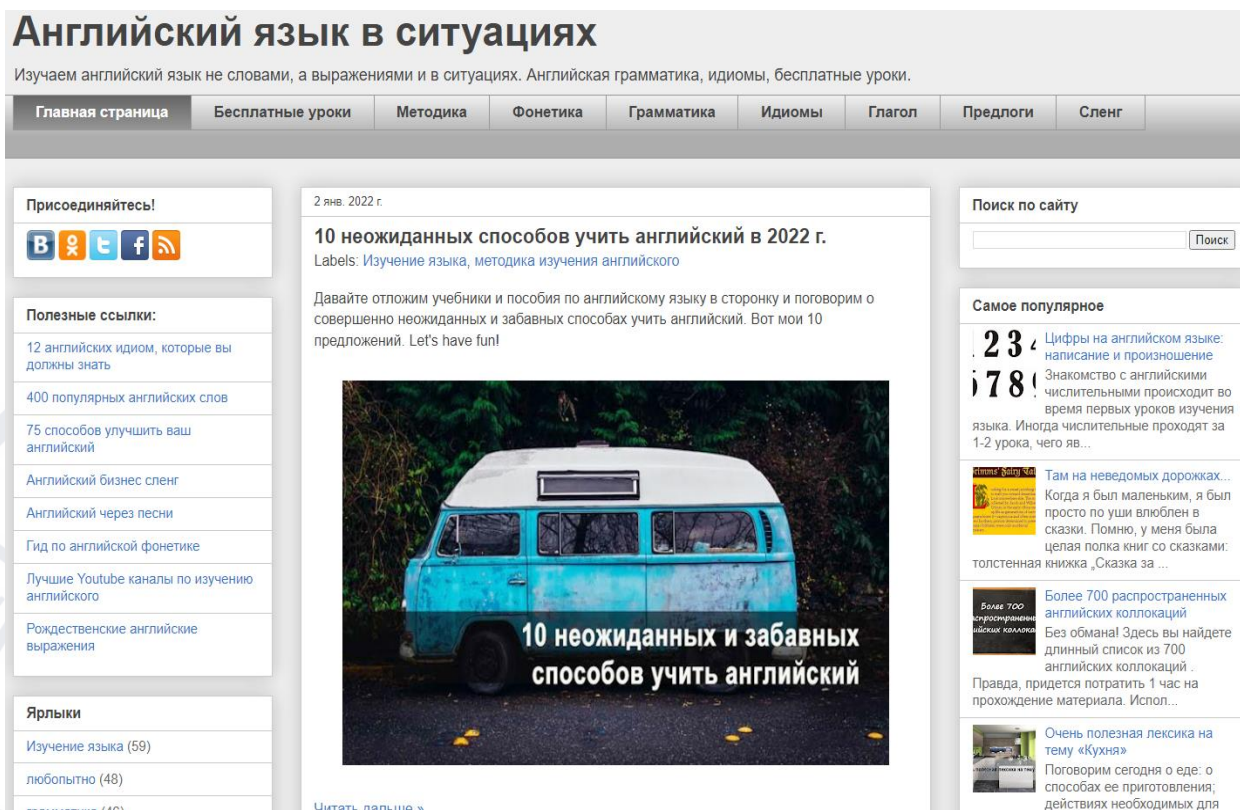


Рисунок 1.4 – Головной экран Situation English

FreeRice [12] – Даний веб-сервіс для вивчення англійської мови представлений в вигляді тренажеру для запам'ятовування та поширення словникового запасу саме на англійській мові з вправами по граматиці та різноманітними тестами. Даний сервіс існує за фінансування Всесвітньої продовольчої програми ООН, тому уроки побудовані у формі гри – за кожну правильну відповідь ви отримуєте трохи рису, щоб нагодувати голодних.

З огляду на всі, вищезгадані веб-додатки, особисто мною було вирішено визначити головні критерії, опираючись на які, мати змогу робити об'єктивно комфортний і правильний додаток, для вивчення англійської мови.

Тому, для зручності мною було вирішено поєднати головні критерії в таблиці 1.1. І після цього, уже опираючись на дану таблицю, робити висновки стосовно того, які помилки не слід в майбутньому повторювати з власним веб-додатком для вивчення англійської мови, і які переваги кожного з веб-додатків використовувати в своєму веб-застосунку.

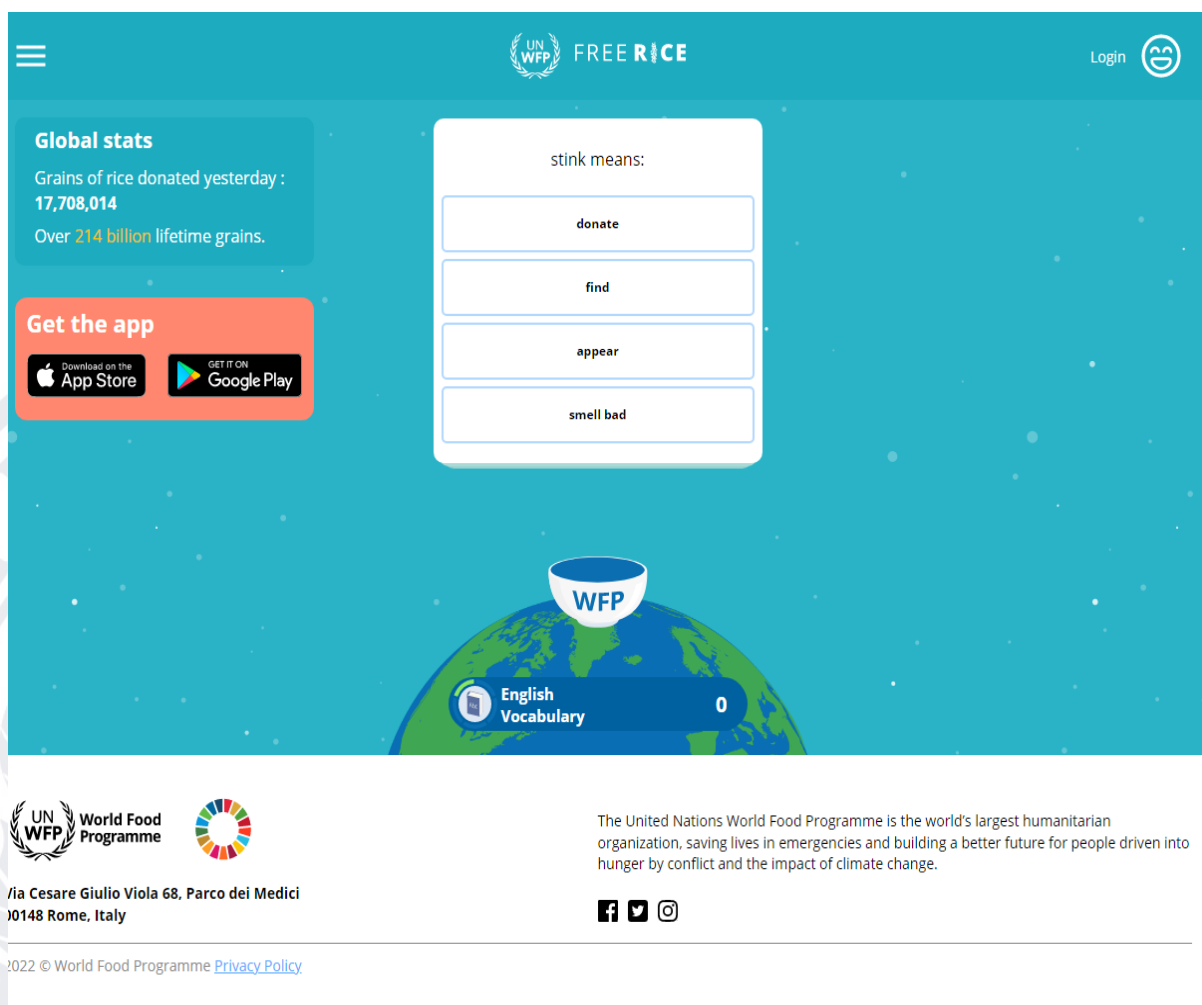


Рисунок 1.5 – Головний екран FreeRice

Тож, з огляду на це, зрозуміло що нашій веб-додаток для вивчення англійської мови слід робити безкоштовним. Це обумовлено тим, що в невідомого нікому додатку ніхто не буде купляти платні функції, і також тим, що платні функції зазвичай можуть відлякувати потенційного користувача веб-додатком.

Наступним критерієм, на мою думку, є одним з найголовніших. Обов'язково слід робити структуровану методику навчання, тому що так для користувача буде простіше запам'ятовувати інформацію, яка не буде хаотично розкидана по різних тематикам.

Однією з потрібних умов, слід щоб на перших етапах, кожна людина мала змогу користуватись додатком і відповідати на питання, тому важливо щоб мінімальний рівень знань для користувача був мінімальний.

Таблиця 1.1

Веб-додаток	FreeRice	Situation English	Duolingo	LearnEnglish
Платні функції	Так	Ні	Так	Так
Структурована методика навчання	Так	Ні	Так	Так
Рівень знань користувача	Початковий	Початковий	Початковий	Початковий
Збільшення складності завдань	Ні	Ні	Ні	Ні
Тренування написання	Ні	Ні	Частково	Частково
Мотивація для користувача	Збирання балів	Ні	Рейтинг	Рейтинг

Однією з важливих потреб для вивчення англійської є вивчення граматики. Тому вважаю за потрібне щоб на кожне питання користувач відповідав саме письмово, а не з варіантами відповідей.

Також, не можливо описати один із аспектів кожного з додатків. Це те, що має назву підвищення складності завдань. В усіх вищезазначених веб-додатках для вивчення англійської мови відсутній процес підвищення складності в запитаннях. На мою думку це є недоліком, тому що користувачу з часом може стати просто не цікаво відповідати на запитання, і користуватись веб-додатком в загальному.

І останнім важливим критерієм, на мою думку, успішного веб-додатку для вивчення англійської мови, є те щоб для користувача була мотивація в

проходженні рівнів, і власне кажучи, щоб відчувати свою значущість і відчуття боротьби в порівнянні з іншими людьми.

Висновок до розділу 1

В цьому розділі було описано аналіз обраної предметної області, а також був проведений огляд існуючих аналогів, розглянуті їх недоліки та переваги. Наступний розділ буде присвячений постановці задачі та основним теоретичним положенням.



РОЗДІЛ II

ОСНОВНІ ТЕОРЕТИЧНІ ПОЛОЖЕННЯ ТА ПОСТАНОВКА ЗАДАЧІ

Перш ніж перейти до розділу з використаними технологіями, не можливо не розповісти про проектування нашого веб-додатку для вивчення англійської мови. Слід визначити структуру цього додатку, як буде проходити процес навчання, які саме питання мають місце бути під час так званого процесу навчання, яке буде заохочення для користувачів даного веб-застосунку та інші питання та нюанси, які можуть виникати при створенні застосунку.

2.1 Визначення загальних процесів роботи додатку

Зважаючи на переваги та недоліки всіх додатків за ідентичною тематикою, які досліджувались в першому розділі, слід спроектувати і продумати логіку для додатку, який ми власне і будемо реалізовувати.

Почати хотілось би з простого, з вибору назви. Хотілось би, щоб вона асоціювалась з тематикою веб-додатку. Тому після довгих роздумів, було прийнято рішення присвоїти веб-застосунку назву «EasyLang». Дана назва ідеально поєднує класифікацію нашої програми, легкість сприйняття інформації та власне назву, яка легко запам'ятовується.

Стосовно кількості сторінок на сайті, та їх функціоналу. Під час виконання роботи, планується розробити три головні розділи:

- Головна сторінка.
- Сторінка з процесом навчання.
- Сторінка з таблицею рекордів.

На головній сторінці буде міститись коротка інформація про процес навчання та загальна інформація про веб-застосунок. З головної сторінки буде

можливість перейти на сторінку з таблицею рекордів та безпосередньо на процес навчання.

Таблиця рекордів буде містити звичайну таблицю з кількістю набраних балів, які відповідають нікнейму користувача, який їх набрав. В списку буде відображатись топ-10 користувачів сайту, і також позиція користувача стосовно інших користувачів, якщо він не входить в топ-10 цих користувачів.

Сторінка з навчальним процесом буде складатись з 4 рівнів, кожен з яких в свою чергу буде містити по 4 уроки. Зрозуміло, що рівні між собою будуть відрізнятись складністю питань, які будуть міститись в кожному з уроків. Обов'язковою умовою, для переходу на новий рівень буде виконання всіх уроків з попереднього рівня. Для переходу на новий урок всередині кожного з рівнів, достатньо відповісти правильно на 85% запитань з цього уроку. В будь-який момент користувач зможе припинити проходження прогресу в кожному з уроків, якщо йому наприклад потрібно буде відвідати таблицю рекордів, або відвідати головну сторінку сайту.

2.2 Визначення ключових моментів

При моделюванні було визначено наступні аспекти нашого веб-застосунку.

Стосовно питань і завдань, які будуть зустрічатись під час процесу навчання. Всі питання будуть взяті з просторів інтернету, і в тому числі з додатків, на переваги та недоліки яких було досліджено їх в першому розділі. Основною проблематикою під час створення додатку є не складність та суб'єктивна думка стосовно складності та якості запитань, для вивчення англійської мови в загальному. Тому буде зроблено файл формату .json , з потрібною кількістю запитань. Під час розробки ці питання з файлу будуть відображатись безпосередньо на екрані, в потрібних місцях.

Також однією з вимог до мобільного додатку, на мою думку, є можливість користуватись з мобільного телефону. Тому було прийнято рішення також розробити функціонал для всіх пристроїв ширина екрану, яких не менша ніж 360 пікселів.

Після повного проходження користувачем будь-якого з рівнів, веб-додаток повинен обрахувати кількість правильних відповідей, і вивести на екрані результат правильних відповідей. Якщо користувачем досягнуто більше 85 відсотків правильних відповідей, то на екрані показано відповідний текст. Аналогічно з негативним результатом.

Мною було вирішено, що під час відповіді на одне з будь-якого питання, не буде показано чи користувач правильно відповів чи ні. Вважаю, що користувач не повинен мати пригнічений стан, в разі декількох неправильних відповідей підряд або взагалі в разі декількох неправильних відповідей. Після відповіді на всі двадцять питань, які містяться в кожному з уроків, буде обраховуватись правильна кількість відповідей, і веб-додаток буде інформувати користувача на рахунок того, чи зміг користувач пройти певний рівень, і набрати потрібну кількість балів для цього.

Після проходження користувачем повністю всіх рівнів та уроків з повним прогресом правильних відповідей, користувач може заново пройти вже пройдені уроки, без впливу на кількість зароблених балів.

2.3 Постановка задачі

1. Створити веб-додаток.
2. Розробити логіку для вивчення англійської мови.

Висновок до розділу 2

В цьому розділі було описано основні теоретичні положення та аспекти нашого веб-додатку для вивчення іноземної мови. Було проведено аналітичну роботу з приводу того, як має виглядати веб-застосунок, та які ключові моменти повинні виконуватись під час його роботи. Також було затверджено постановку задачі. Наступний розділ буде присвячений вибраним технологіям для розробки, та власне, реалізації веб-додатку.

РОЗДІЛ III

РЕАЛІЗАЦІЯ ВЕБ ДОДАТКУ ТА ОПИС ВИКОРИСТАННИХ ТЕХНОЛОГІЙ

Для розробки веб-додатку, слід визначити одні з найголовніших технологій, які будуть використовуватись безпосередньо під час розробки. До таких можна віднести:

1. Мова програмування.
2. Веб-фреймворк.
3. Середовище розробки IDE.
4. Засіб хостингу застосунків.

3.1 Вибір мови програмування

Перш ніж перейти до вибраної мови програмування, не можна не сказати пару слів про HTML [12] [13] [14] і CSS [15] [16] [17].

HTML – це мова розмітки гіпертексту, яка дозволяє переглядати веб-сторінки у браузері. Процес роботи даної технології доволі простий, веб-браузери отримують HTML документ від певного сервера, або наприклад відкривають його локально на пристрої (якщо мова йде про розробку сайту з використанням даної технології), після чого інтерпретують код в інтерфейс, який і відображається безпосередньо на екрані монітора.



Рисунок 3.1 – Логотип HTML5

Елементи HTML можна назвати будівельними блоками для веб-сторінки. За допомогою так званих тегів можна виводити безпосередньо інформацію на сторінку, таку як зображення, відео, поля для вводу тексту та інші.

Загалом, можна визначити такі функції в HTML:

- Створення документу, який буде структурований за допомогою тегів: зображення, заголовки, таблиці, абзаци тощо.
- Отримання інформації через гіперпосилання.
- Створення різноманітних інтерактивних форм.
- Можливість додавати зображення, відео до тексту.

В HTML доступна функція для вбудовування скриптів, написаних на мовах програмування. Додавання для HTML сторінки CSS дає змогу змінити вигляд, компоновання та розміщення елементів.

CSS – це спеціальна мова стилю сторінок, яка призначена для того, щоб описати графічне представлення цих сторінок, які в свою чергу написані за допомогою HTML.

CSS використовується авторами веб-сторінок здебільшого задля того щоб визначити кольори, шрифти, верстку в загальному або інші складові сторінки.



Рисунок 3.2 – Логотип CSS

Можна виділити такі переваги CSS:

- Стили для сайту можуть міститися в одному .css-файлі, що дозволяє швидко корегувати їх зміст за потреби.

- Прискорення завантаження сторінок.
- Зменшення обсягів інформації, що передається на сервер.
- Сторінки зменшуються в об'ємі та стають більш структурованими.

Однак в подальшому веб-додаток не буде написаний в чистому вигляді, а за допомогою препроцесора SCSS [18] [19] файли розширення .scss буде інтерпретовано в файли з розширенням .css.



Рисунок 3.3 – Логотип препроцесора SASS.

Загалом, не буде помилкою назвати препроцесор SASS «синтаксичним цукром» [20] по відношенню до css. Оскільки, SASS розширює функціонал CSS, додаючи до останнього кілька зручних та потрібних для швидкої розробки, механізмів. Препроцесор SASS дозволяє визначати змінні. Це зручно тоді коли слід змінити колір відразу в декількох місцях під час розробки, і не стає проблемою змінити в одному місці. Також препроцесор SASS має, як на мене, одну з ключових особливостей – вкладені селектори. Це значно покращує та пришвидшує розробку, а також зменшує загальну кількість стрічок коду в файлі. Також препроцесор SASS має можливість створювати міксини, перебору змінних в циклі і успадкування для визначених селекторів.

Мовою програмування, яка буде використовуватись під час розробки даного веб-додатку, було обрано JavaScript [21] [22] [23].



Рисунок 3.4 – Логотип JavaScript.

JavaScript класифікують як динамічну та об'єктно-орієнтовну [24] мову програмування. Найчастіше дана мова програмування має можливість бути використаною для різноманітних сценаріїв в веб-сторінках, що в подальшому дає змогу взаємодіяти якимсь чином з користувачем, асинхронно обмінюватись даними, змінювати структуру або зовнішній вигляд сторінки.

Можна сказати, що JavaScript прототипна, скриптова мова програмування з динамічною типізацією, яка має функцію автоматичного керування пам'яттю.

Мова JavaScript використовується для:

- Створення односторінкових веб застосунків за допомогою таких технологій як React, Vue, AngularJS.
- Написання сценаріїв для надання сторінкам інтерактивності.
- Програмування на боці серверу за допомогою такої технології як NodeJS.
- Створення мобільних застосунків за допомогою React Native.
- Створення стаціонарних застосунків.
- Створення сценаріїв в прикладних програмах.

Мова програмування JavaScript – це мова програмування, яка при використанні дозволяє надавати веб-сторінці інтерактивності, тобто робити веб-сторінку такою, щоб з нею міг безпосередньо взаємодіяти користувач при відвідуванні цієї веб-сторінки. Послідовність дій та інструкцій виконується

інтерпретатором вбудованим в будь який звичайний веб-браузер користувача. Простими словами, код програми вбудовується в HTML-сторінку, і виконується код саме на стороні клієнта. Для виконання програми користувачу навіть не слід перезавантажувати веб-сторінку, всі програми написані за допомогою мови програмування JavaScript, виконуються в відповідь на якусь певну подію. Наприклад, перед відправленням даних з певної форми можна перевірити значення яке ввели в дану форму, або написати якусь умову перевірки цих значень, і якщо наприклад значення не пройшли перевірку, то заборонити відправлення цих значень. Приблизно такий алгоритм дій застосовується при формі входу на будь який сайт, коли слід перевірити на правильність поле логіну або пароллю [25] .

3.2 Вибір веб-фреймворку

Перш ніж перейти безпосередньо до обґрунтованого вибору веб-фреймворку не можливо не розповісти про основні застосунки для розробки веб-додатків, які зараз найпоширеніші та найпопулярніші в даній сфері програмування. Але навіть є умови, при яких кілька інтерфейсних фреймворків можуть бути дуже хорошим рішенням одночасно.

Тому, важливе питання – які нині тенденції розвитку front-end?

На мою думку, немає сенсу говорити, що існує якийсь унікальний веб-фреймворк, який найкраще підходить для вирішення будь-яких завдань та проектів, які поставить перед собою людина. Звісно, потрібно розуміти, що деякі фреймворки працюють краще, ніж інші, і вони є найкращі для конкретного випадку використання.

З огляду, на інформацію та статистику, яка викладена за посиланням [26], можна виділити ТОП-3 основних і найпоширеніших веб-фреймворків: React, Vue.js та Angular.

Тому хотілось би навести коротку інформацію про кожну з цих технологій, та описати головні недоліки та переваги кожного з цих фреймворків.

Angular — JavaScript-фреймворк з відкритим програмним кодом, який розробляє Google, який призначений для розробки односторінкових додатків, що складаються з одної HTML сторінки з CSS і JavaScript [27].



Рисунок 3.5 – Логотип Angular

З переваг даного веб-фреймворку можна виділити основні:

- Angular має в собі не тільки інструменти, але і шаблони певного дизайну для створеного проекту, що обслуговується. При правильному створенні програми Angular у вас не буде плутанини класів і методів, які є складними для перевірки і ще складніше тестувати.
- Angular має багато інструментів безпосередньо для створення програми. Завдяки HTML-елементи навіть можуть поводитися динамічно.
- Проста робота з формами.
- Angular підготовлений до мобільних пристроїв та десктопу — один фреймворк під безліч платформ.

З недоліків же розробники зазвичай виділяють такі:

- Рекомендується писати програми Angular на TypeScript, тому його потрібно розуміти.

- Щоб ще більше прискорити процес розробки, потрібно зрозуміти Angular CLI.
- Для встановлення Angular та інших компонентів активно використовується npm. Його також треба знати.
- Під час розробки програми дуже важливо мати можливість налагоджувати код. Потрібно вміти працювати в інструментах налагодження.

Як бачимо, потрібно багато чого вивчити, щоб почати створювати клієнтські веб-програми саме з використанням веб-фреймворку Angular.

Vue.js - JavaScript-фреймворк з відкритим вихідним кодом для створення інтерфейсів користувача, яка легко інтегрується у проекти з використанням інших JavaScript-бібліотек і також може функціонувати як веб-фреймворк для розробки односторінкових програм у реактивному стилі [28].



Рисунок 3.6 – Логотип Vue.js

Є певні переваги використання саме фреймворку Vue.js, які мають спонукати розробників використовувати безпосередньо його у своїх проектах. Один з них це те, Vue.js багато в чому схожий на Angular і React і продовжує користуватися зростаючою популярністю в порівнянні з іншими фреймворками. Тому людям, які до Vue.js мали досвід роботи з Angular або React мають змогу швидко його освоїти. Розмір так званого каркасу фреймворку 20 кілобайт, що полегшує для розробників завантаження файлів. Взагалі, якщо говорити про час завантаження та використання, то Vue.js в цьому плані перевершує інші фреймворки, які вже були згадані до цього.

Переваги даного фреймворку:

- Простота. Найголовніша ідея розробки фреймворку Vue.js полягає у успішному виконанні хороших результатів в плані розробки з мінімально можливими зусиллями розробника, грубо кажучи, щоб розробник міг кодувати, використовуючи лише кілька рядків. Vue.js так само як і React також відмінно підходить для роботи з компонентами.
- Інтеграція. Розробники даного фреймворку можуть інтегрувати Vue.js в інші фреймворки, такі як, наприклад, React, що дозволяє їм налаштовувати проєкт до тих вимог, які їм будуть потрібні під час розробки або підтримання веб-додатку. Тому, можна сказати, що завдяки простоті інтеграції, Vue.js стає популярним вибором для веб-розробки, оскільки за потреби його можна використовувати з різними існуючими веб-додатками.
- Дружній інтерфейс. На думку різних програмістів, фреймворк Vue.js не вимагає сумлінних годин навчання, та купу витраченого на це часу, що вигідно для програмістів-початківців в даній сфері програмування. Що стосується процесу навчання, цікаво зазначити, що Vue.js потребує, щоб програмісти лише знали деякі основи таких технологій як JavaScript, HTML та CSS, і не важко погодитись, що це дрібниці.
- Налаштування. Під час використання даного фреймворку розробники можуть легко назвати функцію, як їм подобається. Кожен сегмент може мати окремі функції, що полегшує налаштування програми відповідно до індивідуальних вимог кожного розробника.
- Документація. Під час вивчення будь-якої технології важлива щоб була доступно написана документація. Вона показує всі варіанти, що пропонуються платформою, і відповідні приклади використання. Документація на Vue.js постійно вдосконалюється та оновлюється. Він також містить простий вступний посібник і дуже хороший огляд API.

Незважаючи на переваги Vue.js, є кілька недоліків, які було б непогано усунути.

- Vue.js – це відносно новий фреймворк, який поки що не підтримується великою спільнотою активних розробників.
- Відсутність плагінів. Оскільки Vue.js доволі молодий фреймворк, в ньому відсутні деякі потрібні плагіни. Наприклад для використання Google Maps розробники часто вдаються до написання цієї складової на мові програмування JavaScript.

З огляду на приведені вище недоліки та переваги даного фреймворку і опираючись на ці факти, можна зробити такий висновок: фреймворк Vue.js є хорошим інструментом для розробки односторінкових додатків. Те що він має невеликий розмір та можливість налаштування «під себе», роблять його відмінним вибором для розробників, які шукають зручний інструмент для веб-додатків. Деякі з недоліків фреймворку Vue.js можуть бути легко виправлені з часом командою розробників даної технології, що дозволяє розробникам сподіватися, що ця платформа продовжуватиме покращуватися в близькому майбутньому [29].

React - JavaScript бібліотека для створення інтерфейсів користувача з відкритим кодом, яка потрібна, щоб вирішувати проблеми часткового оновлення вмісту веб-сторінки, з якими зустрічаються під час розробки деяких односторінкових застосунків. Розробляється з допомогою Facebook, Instagram і товариством індивідуальних розробників [30].

Бібліотека React дає змогу користувачам даної технології створювати великі веб-застосунки, які під час своєї роботи можуть використовувати такі дані, які можуть змінюватись з часом саме без перезавантаження сторінки. Мета даної бібліотеки полягає в тому, щоб бути швидкою, простою, масштабованою.

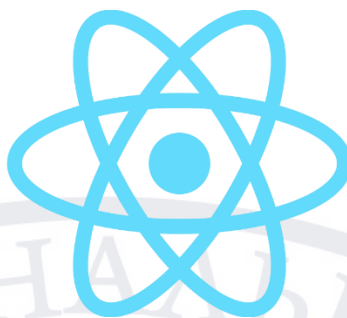


Рисунок 3.7 – Логотип React.

Головною особливістю React є те, що він при своїй роботі використовує віртуальний DOM [31], не покладається виключно на DOM браузера. Саме такий підхід дозволяє бібліотеці визначити, які частини DOM були змінені під час роботи, порівняно зі збереженою версією віртуального DOM, і внаслідок чого визначати, як буде найкраще та найефективніше оновити DOM браузера. Простими словами, можна сказати, що коли користувач співпрацює з елементами на сторінці, він вважає, що вона оновлюється вся, але насправді бібліотека оновлює лише необхідне для цього. Оновлення всієї DOM структури для того щоб зробити веб-сторінку «реактивною» є просто неефективно, оскільки даний підхід буде використовувати занадто багато ресурсів. Подібний процес на перший погляд може здаватись незрозумілим і енергозатратним, проте саме він займає набагато менше часу, оновлення всієї реальної DOM сторінки повністю. Тому, він оптимізує і полегшує роботу з DOM структурою.

Маючи досвід роботи з даною бібліотекою, можу виділити основні переваги даної технології:

- Зазначена вище технологія віртуального DOM.
- Висока продуктивність. Для будь-якої веб-сторінки одна з головних цілей – бути швидкою і забезпечити найшвидше обслуговування клієнта цієї веб-сторінки. Дана перевага можлива за допомогою вже вище згаданої DOM структури.
- Повторне використання компонентів. Основною ідеєю бібліотеки React є розбиття коду на компоненти. Частіше за все вони проєктуються та

створюються багаторазовими, щоб мати можливість використовувати його в інших частинах коду, або навіть по можливості в інших проектах.

- Односторонній потік даних. Або його ще можна назвати від батьківського елемента до дочірнього. В React-застосунку дані між елементами передаються лише одним способом. Такий підхід запобігає помилкам, також полегшує налагодження коду.
- Велика спільнота розробників. Компанія Facebook приклала чимало зусиль для того, щоб зробити бібліотеку React дуже хорошим інструментом для розробки та покращення користувацького інтерфейсу будь-яких веб-застосунків. Більше того, Facebook продовжує активно продовжує робити React більше ефективним і приємним у використанні.
- Браузерні інструменти розробника. Одним з таких є React Developer Tools – це безкоштовне розширення для таких браузерів як Google Chrome та Firefox, який надає набір інструментів для налагодження коду, моніторингу компонентів прямо в браузері, тощо.
- Хороша документація. Документація є одним із багатьох способів вивчити технологію в різних варіаціях: як звичайна теорія і з елементами практики, для поглибленого та базового вивчення, так і практичний приклад гри, для людей які люблять вивчати нові технології відразу на практиці.

Як і будь-яка інша технологія, бібліотека React можна характеризувати не тільки перевагами, а і певною кількістю недоліків.

- Синтаксис JSX. В бібліотеці React використовується JSX – так зване спеціальне розширення синтаксису для створення об'єктів JavaScript, але за допомогою синтаксису HTML. Таке розширення спрощує модифікацію DOM і також покращує читабельність коду. Однак для новачків такий підхід, може, на перший погляд, здатись важким та не зрозумілим. Тому для них потрібен час для розуміння цього синтаксису. Внаслідок цього деякі розробники часто вважають JSX недоліком бібліотеки React.

- Фокус на користувацькому інтерфейсі. React це саме та бібліотека, яка має в собі набір інструментів, які потрібні для створення користувацького інтерфейсу веб-застосунків. Але React не можна називати повноцінним фреймворком, хоча деякі розробники помилково так вважають. React – це не інструмент, який має в собі «все в одному» для створення цілого застосунку, який потрібний в тих чи інших цілях. Грубо кажучи, можна сказати що React відповідає тільки лише за представлення застосунку.

Тому, з огляду на всі вищевказані недоліки та переваги бібліотеки React, можна зробити висновок, що React створений і використовується для створення інтерактивних користувацьких інтерфейсів. Бібліотека себе в цьому добре показала, тому і має широке використання в даній сфері.

Бібліотека React характеризується великою кількістю переваг для розробників. Користувачі даного інструменту можуть запросто ними використовуватись, але при створенні веб-застосунку, слід також пам'ятати про недоліки даної технології. Тому що, як і будь яка інша технологія, бібліотека React їх має [32].

З огляду на всі вищезазначені короткі характеристики кожної з технологій, та з огляду через призму суб'єктивності, роблю вибір на користь бібліотеки React. Тому що, маю досвід роботи з даною технологією, та особисто мені вона здається найоптимальнішим варіантом для вирішення завдання, яке постає внаслідок цієї роботи.

Також, було б помилкою не розказати про ще одну технологію, яка буде безпосередньо використовуватись під час розробки та проектування веб-додатку для вивчення англійської мови. Мова йде про бібліотеку Redux.

Redux – відкрита бібліотека написана на мові програмування JavaScript, яка призначена для полегшення в управлінні так званим станом програм та компонентів написаних за допомогою мови програмування JavaScript. Це окрема бібліотека, але найчастіше використовується в парі з такими технологіями як

вищезазначені бібліотеки React та Angular для простішого та зручнішого побудови інтерфейсу користувача [33].



Рисунок 3.8 – Логотип Redux

З власного досвіду можу сказати, що бібліотека Redux дуже сильно покращує та полегшує взаємодію з станом кожного компоненту і внаслідок використання даної бібліотеки покращується взаємодія між компонентами.

Взагалі, використовувати бібліотеку Redux на будь-якому проекті не завжди є правильно. На думку фахівців, ось коли потрібно її використовувати [34]:

- Коли під час розробки проекту, є значна кількість так званих станів програми, які використовуються в великій кількості місць.
- Стан компонентів часто оновлюється.
- Логіка оновлення стану в компонентах є складною.
- Проект має кодову базу середнього або великого рівня за розмірами, і над нею працює велика кількість людей.

Зрештою, Redux це всього лиш додатковий інструмент, який не завжди потрібно використовувати в парі з React, і є кілька вагомих причин для його викристання, однак є і такі внаслідок яких використовувати Redux не слід. Тому слід зважено приймати рішення на рахунок цього. Щодо подальшої реалізації нашого веб-застосунку для вивчення англійської мови, Redux використовуватись буде, тому що, на мою думку, це допоможе швидше та якісніше реалізувати веб-застосунок.

3.3 Вибір середовища розробки IDE

Вибір комфортного і потрібного середовища розробки саме для проектування і розробки front-end частини веб-застосунку для вивчення англійської мови був зроблений на користь Visual Studio Code. Вибір цей було зроблено з наявного доволі великого досвіду роботи в даному середовищі розробки.

Visual Studio Code – програмний застосунок для створення, редагування і налагодження сучасних веб-застосунків і різноманітних програм. Visual Studio Code є безкоштовною програмою і має сумісність з Windows, Linux і OS X [35].

Особисто для мене, цей редактор коду зручний тим, що він містить у своїх функціях дебагер, інструменти для роботи з git, засоби рефакторингу, контексті підказки, автодоповнення конструкцій кодових і також навігацію по коду. І це тільки невелика збірка основних переваг. Варто зазначити, що це не інтегроване середовище розробки, а звичайний редактор коду. Різниця в том, що в редакторі менше функціоналу, але це не значить, що немає потрібних функцій для розробки, тому що, за допомогою так званих плагінів і розширень, кількість і функціонал яких дуже розширений, можна налаштувати редактор коду, так як заманеться будь-якому розробнику.

Як бачимо, з рисунку, який наведено вище, зліва в програмі є можливість переглядати, які файли доступні для редагування, доступний пошук по файлам, синхронізація з git та показ останніх внесених змін, дебагер, та вікно для встановлення різноманітних розширень.

Відкриті файли можна комбінувати і по два, і по три відкритих файли відразу, інколи це позитивно впливає на швидкість розробки продукту. Також є доступна підсвітка коду, і тема заднього фону, для якої є безліч користувацьких тем, які розробили інші люди.

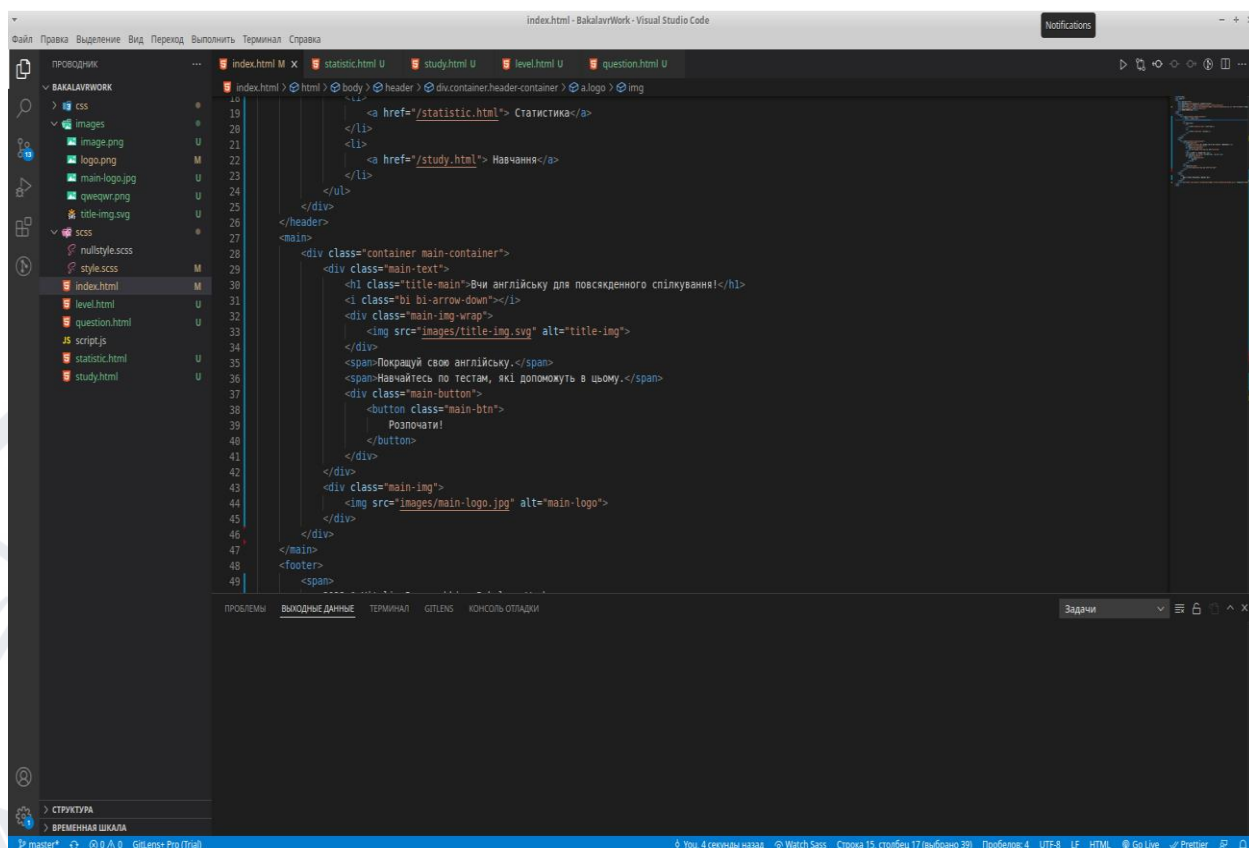


Рисунок 3.9 – Приклад робочого простору Visual Studio Code.

Знизу є приступний командний рядок, який дозволяє запускати проекти реакт, або відправляти зміни на git наприклад.

Це лише короткий опис можливостей даного середовища розробки, насправді це дуже потужний інструмент, тому і не дивно що він користується великою популярністю серед розробників.

Середовище розробки Visual Studio Code підтримує основні мови програмування і технології, одні з яких JavaScript, C#, C++, TypeScript, PHP, Python, HTML, CSS, SASS, React, Vue.js та інші. Як можна побачити, з вищевказаних раніше обраних технологій для розробки додатку, Visual Studio Code підходить ідеально.

3.4 Засіб для хостингу проекту.

Після вибору всіх необхідних інструментів, для розробки нашого веб-застосунку, для вивчення англійської мови, та після остаточного вибору

середовища розробки, наступним, і власне останнім кроком перед проектуванням та безпосередньою розробкою додатку буде вибір хостинг провайдеру, на серверах якого, власне, і буде неперервно працювати нашій додаток. Перед вибором серверу провайдеру, потрібно розуміти, що деякі провайдери повинні підтримувати інструменти та мову програмування, на яких, власне кажучи, і реалізований нашій додаток. В нашому випадку, це мова програмування JavaScript та бібліотека React.

Для обрання провайдеру, варто взагалі поглянути на наявні на так званому «ринку» пропозиції цих самих сервісів. Ключові компанії, які надають такі послуги, це Heroku, відгалуження в GitHub, яке має назву GitHub Pages та Vercel. Всі ці сервіси підходять для поставлених нами завданням, та деякі з них в свою чергу містять певні переваги та недоліки, про які йтиметься нижче.

Проаналізувавши таблицю 3.1, як хостинг провайдер було обрано GitHub Pages.



Рисунок 3.10 – Логотип GitHub Pages.

Тому що, цей провайдер не містить в собі важкого процесу, власне, хостингу застосунків, має безкоштовне використання, та для того щоб зробити це потрібно просто відправити свій проект на github.

Більш детальну інформацію про використання, властивості тощо, хостинг сервісів Heroku, github pages та vercel можна дізнатись за посиланнями [36] , [37] та [38] відповідно.

Таблиця 3.1 – Порівняння хостинг провайдерів.

Критерій	Хостинг провайдер		
	Heroku	GitHub Pages	Vercel
Можливість для роботи із сервером через API	Так	Так	Так
Об'єднання з середовищем розробки	Ні	Ні	Ні
Можливість хостингу з консолі середовища розробки	Так	Ні	Так
Безкоштовне використання	Так	Так	Так
Ціна за місяць користування	25\$	Тільки плата за підключення баз даних	30\$

3.5 Реалізація веб-додатку.

Опираючись на другий розділ, в якому було визначено що в загальному нашій веб-додаток буде мати три основні розділи (головна сторінка, таблиця статистики, процес навчання), зважаючи на всі процеси проектування, першою справою спочатку слід нашій майбутній веб-додаток зверстати. А саме, створити три файли розширення .html які будуть відповідати за вже вищезгадані сторінки статистики користувачів, процесу навчання та головної сторінки.

Перед тим як безпосередньо додавати контент на кожну з сторінок, слід створити так званий header (шапка) і footer (підвал) сайту.

В шапці сайту необхідно, щоб містилась інформація про назву сайту з одного боку, та навігація в вигляді вкладок доступних сторінок, які реалізовані на сайті.

Зрозуміло, що це лиш формальне представлення шапки. Але, зазвичай, саме так здебільшого проектується і реалізовується дана частина будь-якого сайту або веб-додатку.

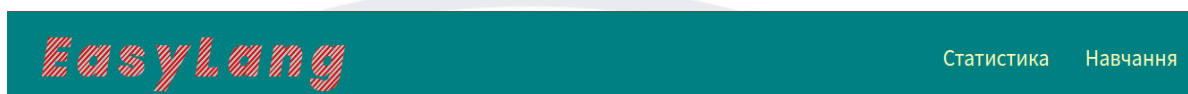


Рисунок 3.11 – Шапка веб-додатку.

В підвалі будь-якого сайту зазвичай розміщують навігацію меню, контактну інформацію, посилання на соціальні мережі. Також інколи розробники можуть розмістити логотип сайту. Ну і найчастіше додають інформацію про те, коли останній раз було внесено зміни до сайту, і хто розробив сайт.

Такий підхід ідеально підходить, якщо між шапкою і підвалом сайту розміщується багато контенту, і з'являється прокрутка сторінки. В разі того, коли користувач доходить до низу сторінку, він може скористатись меню, безпосередньо внизу сторінки, а не прокручувати сторінку на початок.

Оскільки, в загальному і здебільшого на всіх сторінках нашого веб-додатку буде міститись не так багато інформації, а вся інформація буде поміщатись повністю на екрані навіть без прокрутки, то в підвалі нашого сайту буде міститись лише інформація коли було розроблено веб-додаток, хто його розробив, та в рамках чого це було розроблено.



Рисунок 3.12 – Підвал веб-додатку.

Варто зазначити, що шапка і підвал сайту, будуть зустрічатись на кожній сторінці нашого веб-додатку.

На головній сторінці, мною було вирішено розмістити коротку інформацію про закликання до навчання, в супроводі з графічної картинкою, яка потрібна лише для привертання уваги користувача.

Також відразу під блоком про інформацію міститься кнопка, з надписом «Розпочати!» та з анімацією, яка змушує натиснути на себе, тим самим швидше розпочати навчання.

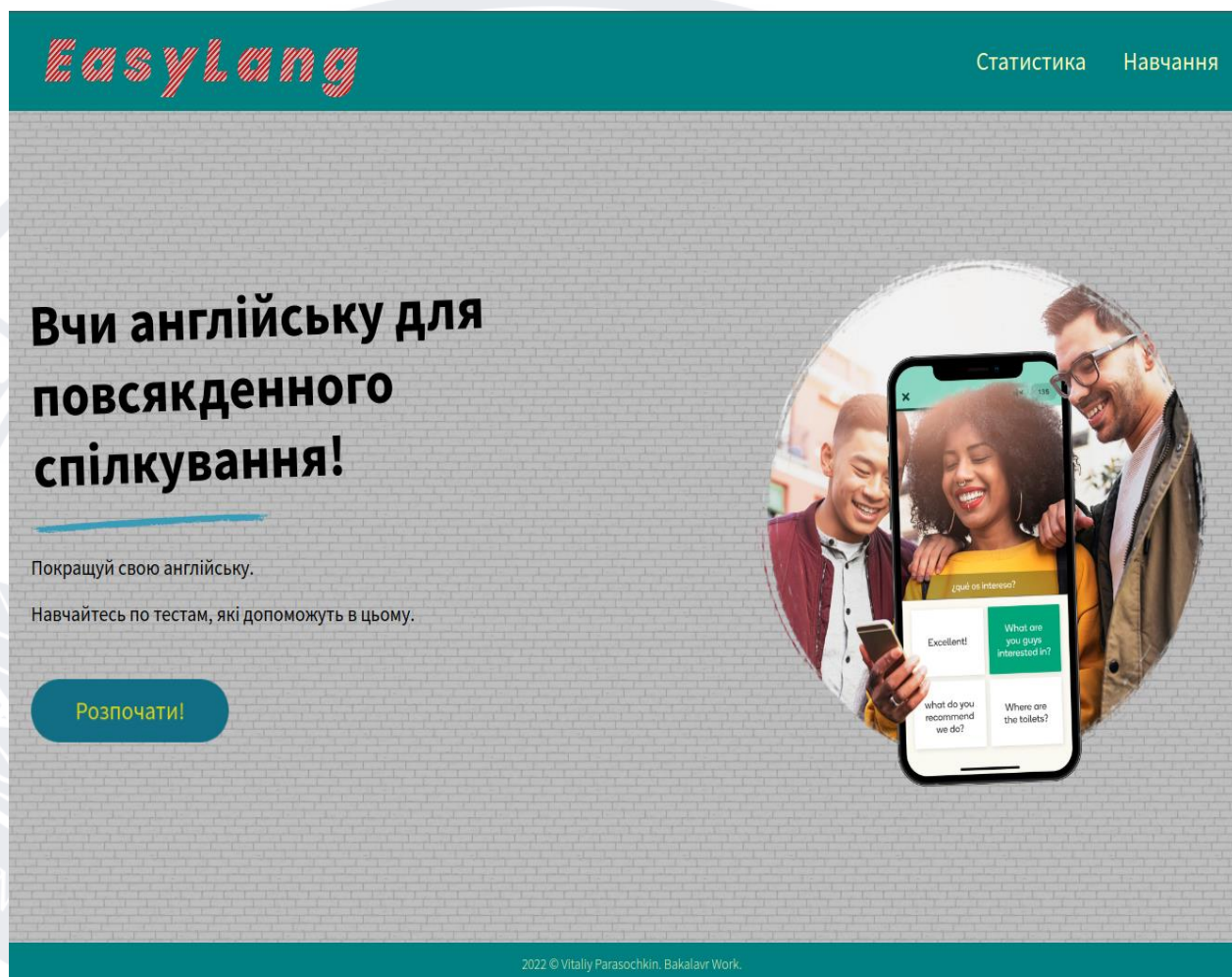


Рисунок 3.13 – Головна сторінка сайту.

Наступна сторінка, яка була реалізована – це сторінка статистики та результатів набраних балів іншими користувачами. Суть сторінки доволі проста, в ній буде зберігатись кількість набраних балів кожного користувача, який користується даним веб-додатком.

Результат буде зберігатись в форматі таблиці з двома полями: нікнеймом користувача та кількістю його набраних очок.



Рисунок 3.14 – Сторінка статистики користувачів.

Наступна сторінка, яку слід створити стосується процесу навчання в цілому. Відразу при натисканні на кнопку «Розпочати!» або ж якщо скористатись кнопкою в навігаційному меню, яке знаходиться в шапці профілю для переходу на сторінку навчання, ми побачимо на екрані сторінку з чотирма кнопками.



Рисунок 3.15 – Сторінка процесу навчання.

На даній сторінці знаходиться чотири кнопки, які відповідають за кількість рівнів складності, які реалізовані в веб-додатку.

Як вже було сказано в теоретичній частині про веб-додаток, допоки користувач не зможе повністю пройти всі уроки в кожному з рівнів, йому будуть недоступні інші рівні. Тому як видно на рисунку 3.15, лише перший рівень доступний, всі інші – ні.

Отож, після того як користувач зайде на сторінку навчання, та вибере один з рівнів, в нього на екрані буде розміщена сторінка з уроками для цього рівня.

На цій сторінці, користувач має змогу вибрати один з уроків, який доступний для відповідей на питання.

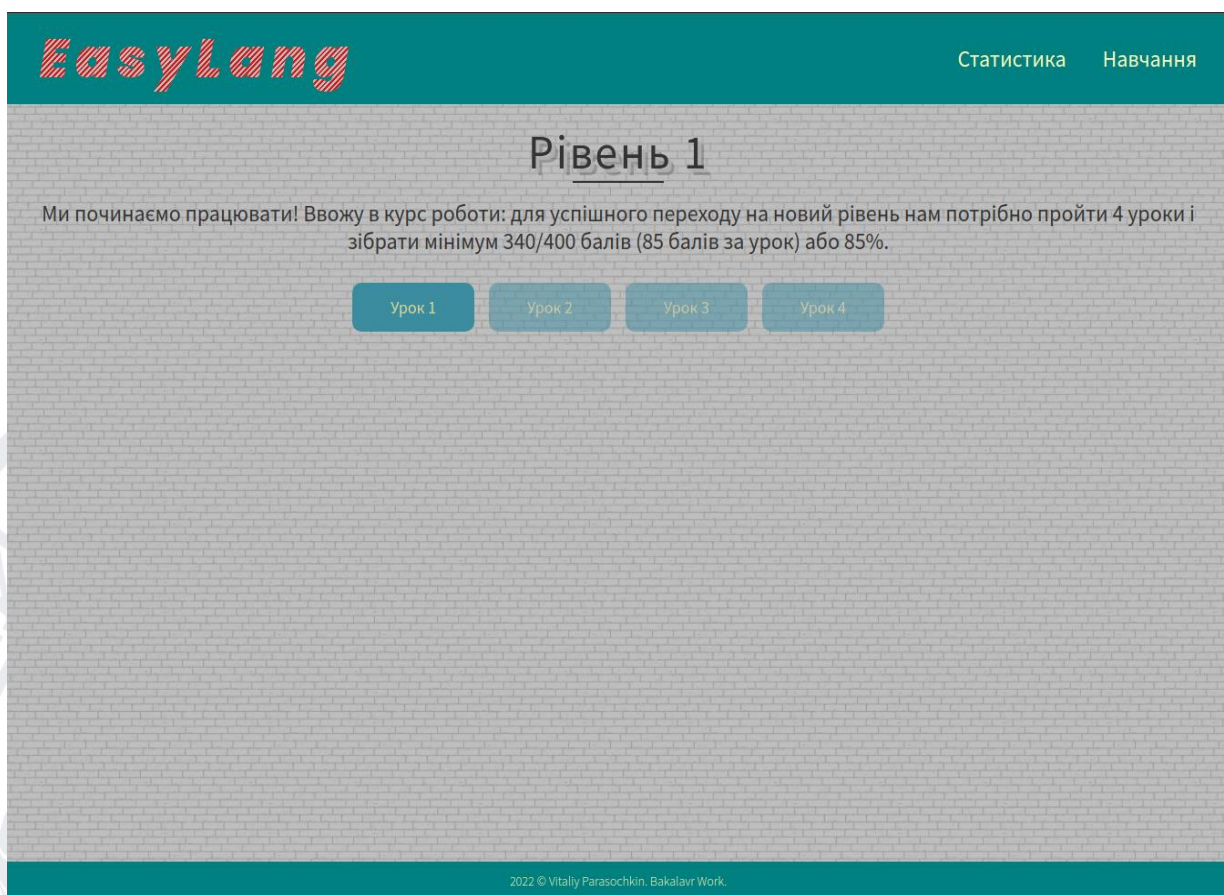


Рисунок 3.16 – Сторінка процесу навчання.

Також для користувача існує надпис, який інформує його про те, в якому форматі проходить процес відповідей на питання, та яку потрібну кількість балів слід набрати.

EasyLang Статистика Навчання

Питання 1 з 20

Напишіть українською

I have receiver your letter

Введіть відповідь...

ЗАКРИТИ УРОК НАСТУПНЕ ЗАПИТАННЯ

2022 © Vitaliy Parasochkin. Bakalavr Work.

Рисунок 3.17 – Екран сторінки з питанням.

На даній сторінці зображено кількість загальних питань та на яке наразі питання з загальної кількості питань користувачу необхідно відповісти. Потім іде суть запитання, це може бути завдання написати український текст на англійській мові, або аналогічно навпаки написати англійський текст на українській мові.

Як вже було згадано в другому розділі теоретичних відомостей про веб-застосунок, коли було вирішено щоб користувач на кожне з запитань відповідав не за допомогою варіантів відповідей, або так званих тестів чи вікторини, а саме щоб вручну писав відповідь. Тому після запитання, розміщено поле вводу відповіді.

Після поля з вводом відповіді мною було вирішено розмістити дві кнопки, одна з яких відповідає за перехід до наступного питання, а інша кнопка розроблена для того, щоб в будь який момент користувач міг завершити процес відповідей на питання, і повернутись до вибору уроку.

Оскільки результат того, чи набрав користувач потрібну кількість балів буде з'являтися саме після того коли він дасть відповіді на всі питання з уроку, потрібно якось сповіщати користувача про його успіх або невдачу.

Тому було вирішено розробити ще дві сторінки, одна з яких відповідає за успішне проходження уроку.



Рисунок 3.18 – Сторінка після успіху в проходженні уроку.

І сторінка яка буде з'являтися при не успішному проходженні уроку користувачем.



Рисунок 3.19 – Сторінка після невдачі в проходженні уроку.

Тому, після створення всіх необхідних статичних сторінок необхідних для правильного і коректного відображення візуальної інформації нашого веб-додатку для вивчення іноземної мови, нашій проект з точки зору архітектури створених файлів має ось такий вигляд як на рисунку 3.19.

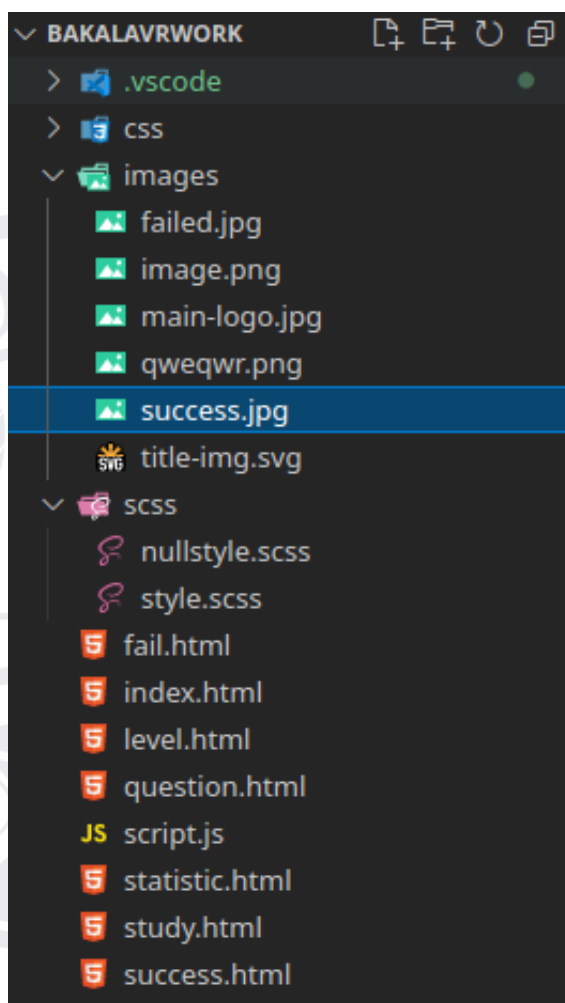


Рисунок 3.19 – Створені файли під час розробки веб-додатку.

Статичні створені файли слід поєднати між собою для переходу між сторінками, і також для створення функціоналу, щоб користувач, власне кажучи і міг відповідати на питання, і взаємодіяти з функціоналом нашого веб-додатку.

Тому наступним і фінальним кроком в створенні нашого веб-додатку є надання останньому інтерактивності та можливості взаємодії з користувачем.

Даний підхід реалізовується за допомогою вищезгаданої бібліотеки React та в нашому випадку в сумісності з бібліотекою Redux.

Після того, як для наших статично створених документів, було додано функціонал і інтерактивність, можна з впевненістю сказати, що нашій веб-додаток повністю готовий.

Змін в графічному представленні в порівняно з вищевказаними рисунками статичної верстки – немає. Єдине що змінилось, це те що користувач нарешті може взаємодіяти з сайтом, відповідати на питання, набирати певну кількість балів по ходу відповідей на запитання тощо.

Єдине що змінилось, це розміщення файлів в нашому каталозі, тепер вони виглядають інакше.

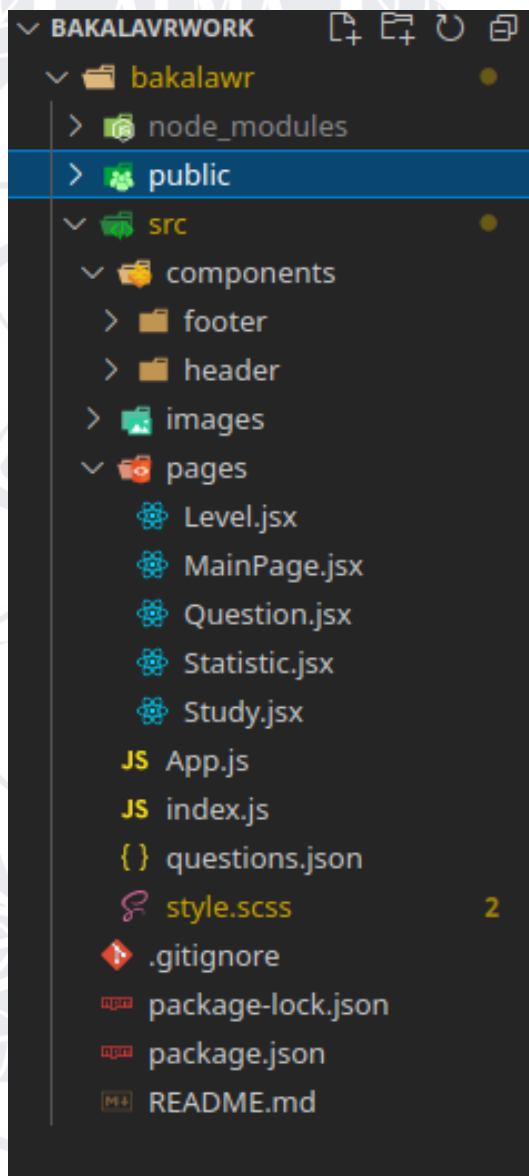


Рисунок 3.20 – Вигляд всіх файлів в каталозі веб-додатку.

Отже, після всіх зроблених дій, можна з впевненістю заявити, що веб-додаток працює, і ним можна користуватись.

Код для працюючого веб-додатку для вивчення англійської мови можна побачити відвідавши посилання [39].

Висновок до розділу 3

В цьому розділі було описано використані технології під час розробки веб-додатку для вивчення англійської мови, а також безпосередньо реалізація цього веб-додатку.



ВИСНОВКИ

В ході виконання бакалаврської роботи було досліджено і виділено основні ключові властивості та етапи і процес розробки веб-додатку. Було проведено аналіз переваг та недоліків, які існують в аналогів. Також розглянуто низку інструментів, які, безпосередньо, використовуються при розробці веб-додатків.

Також був проведений аналіз того, на скільки важлива англійська мова в житті кожного з людей, і на скільки потрібним є розробка веб-додатку для вивчення англійської мови, з урахуванням всіх недоліків аналогів.

Було докладно розглянуто такі технології як HTML, CSS, мова програмування JavaScript, бібліотека React та бібліотека Redux.

Під час виконання роботи, було повністю спроектовано та реалізовано веб-додаток для вивчення англійської мови, яким може користуватись будь-який користувач по бажанню.

Завдання було виконано в повному обсязі.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. ТОП-20 найпоширеніших мов [Електронний ресурс]. - 2020. Режим доступу до ресурсу: <https://chytomo.com/ukrainska-zajniala-44-tu-skhodynku-u-rejtynhu-najposhyrenishykh-mov/>
2. Що таке веб-додаток? [Електронний ресурс]. Режим доступу до ресурсу: <https://tebapit.com/%D1%89%D0%BE-%D1%82%D0%B0%D0%BA%D0%B5-%D0%B2%D0%B5%D0%B1-%D0%B4%D0%BE%D0%B4%D0%B0%D1%82%D0%BE%D0%BA/#:~:text=%D0%9F%D0%BE%D0%B4%D1%96%D0%B1%D0%BD%D0%BE%20%D0%B4%D0%BE%20%D1%82%D0%BE%D0%B3%D0%BE%2C%20%D1%8F%D0%BA%20%D0%BC%D0%BE%D0%B1%D1%96%D0%BB%D1%8C%D0%BD%D0%B8%D0%B9,%D0%B2%D0%B5%D0%B1%2D%D0%B1%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80%20%D1%8F%D0%BA%20%D1%81%D0%B2%D0%BE%D0%B3%D0%BE%20%D0%BA%D0%BB%D1%96%D1%94%D0%BD%D1%82%D0%B0.>
3. Facebook – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу : <https://uk.wikipedia.org/wiki/Facebook>
4. Instagram – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Instagram>
5. Internet Explorer [Електронний ресурс]. – 2022. Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Internet_Explorer
6. Google Chrome [Електронний ресурс]. – 2022. Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Google_Chrome
7. Mozilla Firefox [Електронний ресурс]. – 2022. Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Mozilla_Firefox

8. 20 безкоштовних ресурсів для вивчення англійської мови [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://vsviti.com.ua/interesting/society/56642>
9. Duolingo – Кращий в світі спосіб для вивчення англійської мови. [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://www.duolingo.com/>
10. Англійська мова в ситуаціях [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://www.situational-english.com/>
11. Freerice [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://freerice.com/categories/english-vocabulary>
12. HTML – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/HTML>
13. Дакет Д. «HTML і CSS: дизайн і створення веб-сайтів»: навчальний посібник 2020. С. 5-12. URL: <https://www.livelib.ru/book/1000773334-html-i-css-razrabotka-i-dizajn-vebsajtov-cdrom-dzhon-dakett>
14. Мак-Дональд М. «HTML5. Недостатнє керівництво» 2021. С. 10-13. URL: https://books.google.ro/books/about/HTML5_%D0%9D%D0%B5%D0%B4%D0%BE%D1%81%D1%82%D0%B0%D1%8E%D1%89%D0%B5%D0%B5_%D1%80%D1%83%D0%BA%D0%BE%D0%B2.html?id=ibcSM_cB9X4C&source=kp_book_description&redir_esc=y
15. CSS – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/CSS>
16. МакФарланд Д. «Велика книга CSS» 2018. С. 20-24. URL: <https://www.ozon.ru/product/novaya-bolshaya-kniga-css-135731189/?sh=AvEeLhAclg>
17. Дронов В. «HTML та CSS: 25 уроків для початківців». 2020. С. 25-31. URL: https://books.google.ro/books/about/HTML_%D0%B8_CSS_25_%D1%83%D1%80%D0%BE%D0%BA%D0%BE%D0%B2_%D0%B4%D0%BB%

[D1%8F_%D0%BD%D0%B0.html?id=m9UHEAAQBAJ&source=kp_book_description&redir_esc=y](https://uk.wikipedia.org/wiki/%D0%BD%D0%B0.html?id=m9UHEAAQBAJ&source=kp_book_description&redir_esc=y)

18. SASS – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Sass>
19. Прохоренко Н. «Bootstrap і CSS-препроцесор Sass» 2021. С. 344-346. URL: <https://anyflip.com/ulhe/bbxw/basic>
20. Синтаксичний цукор – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%87%D0%BD%D0%B8%D0%B9%D1%86%D1%83%D0%BA%D0%BE%D1%80>
21. JavaScript – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/JavaScript>
22. Сімпсон К. «Ви не знаєте JS». 2019. С. 11-13. URL: https://books.google.ro/books/about/%D0%92%D1%8B_%D0%BD%D0%B5_%D0%B7%D0%BD%D0%B0%D0%B5%D1%82%D0%B5_JS_%D0%A2%D0%B8%D0%BF%D1%8B_%D0%B8_%D0%B3.html?id=AQCfDwAAQBAJ&source=kp_book_description&redir_esc=y
23. Крокфорд Д. «Як влаштований JavaScript». 2020. С. 5-10. URL: https://books.google.ro/books/about/%D0%9A%D0%B0%D0%BA_%D1%83%D1%81%D1%82%D1%80%D0%BE%D0%B5%D0%BD_JavaScript.html?id=3f-eDwAAQBAJ&source=kp_book_description&redir_esc=y
24. Об'єктно-орієнтоване програмування – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%27%D1%94%D0%BA%D1%82%D0%BD%D0%BE-%D0%BE%D1%80%D1%96%D1%94%D0%BD%D1%82%D0%BE%D0%B2%D0%B0%D0%BD%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F

25. Мова JavaScript та її можливості. [Електронний ресурс]. - 2021. Режим доступу до ресурсу: <https://sites.google.com/site/webtehnologiitawebedizajn/mova-javascript-ta-ieie-mozlivosti>
26. Найпопулярніші Front-end Framework. [Електронний ресурс]. - 2021. Режим доступу до ресурсу: <https://stackdiary.com/front-end-frameworks/>
27. AngularJS – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/AngularJS>
28. Vue.js – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/Vue.js>
29. Плюси і мінуси використання Vue.js. [Електронний ресурс]. - 2022. Режим доступу до ресурсу: <http://fl-it.ru/kakov-plyus-i-minus-ispolyzovaniya-vue-js.html>
30. React – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/React>
31. Об'єктна модель документа [Електронний ресурс]. – 2021. Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%9E%D0%B1%27%D1%94%D0%BA%D1%82%D0%BD%D0%B0_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0
32. Плюси і мінуси React [Електронний ресурс]. – 2021. Режим доступу до ресурсу: <https://nuancesprog.ru/p/14500/>
33. Redux – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Redux>
34. Загальні Redux [Електронний ресурс]. – 2022. Режим доступу до ресурсу: <https://redux.js.org/faq/general>
35. Visual Studio Code – Вікіпедія [Електронний ресурс]. – 2022. Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Visual_Studio_Code

36. Хостинг провайдер Heroku [Електронний ресурс]. Режим доступу до ресурсу: <https://dashboard.heroku.com>
37. Хостинг провайдер GitHub Pages [Електронний ресурс]. Режим доступу до ресурсу: <https://pages.github.com/>
38. Хостинг провайдер Vercel [Електронний ресурс]. Режим доступу до ресурсу: <https://vercel.com/>
39. Відкритий проект на сервісі GitHub [Електронний ресурс]/ github.com. Режим доступу до ресурсу: <https://github.com/vitaliy-parasochkin/bakalavr-work>

Декларація щодо унікальності текстів роботи
та невикористання матеріалів інших авторів без посилань

Парасочкін Віталій Володимирович

Прізвище, ім'я, по батькові

Факультет інформаційних і прикладних технологій

Факультет

122 Комп'ютерні науки

Шифр і назва спеціальності

Сучасні інформаційні технології та програмування

Освітня програма

ДЕКЛАРАЦІЯ

Усвідомлюючи свою відповідальність за надання неправдивої інформації, стверджую, що подана кваліфікаційна (бакалаврська) робота на тему: «РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ ВИВЧЕННЯ АНГЛІЙСЬКОЇ МОВИ» є написаною мною особисто.

Одночасно заявляю, що ця робота:

- не передавалась іншим особам і подається до захисту вперше;
- не порушує авторських та суміжних прав, закріплених статтями 21-25 Закону України «Про авторське право та суміжні права»;
- не отримувались іншими особами, а також дані та інформація не отримувались у недозволений спосіб.

Я усвідомлюю, що у разі порушення цього порядку моя кваліфікаційна (бакалаврська) робота буде відхилена без права її захисту, або під час захисту за неї буде поставлена оцінка «незадовільно».

дата

підпис