

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені Ігоря Сікорського»

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**Кафедра системного програмування та спеціалізованих комп’ютерних систем**

**Лабораторна робота №1**

з дисципліни **«Програмування Web-інтерфейсів»**

Тема: **«Розробка клієнтської частини Web-додатка»**

Виконав:

студент І курсу

групи КВ-31мн

Зіменков Д. А.

Перевірив:

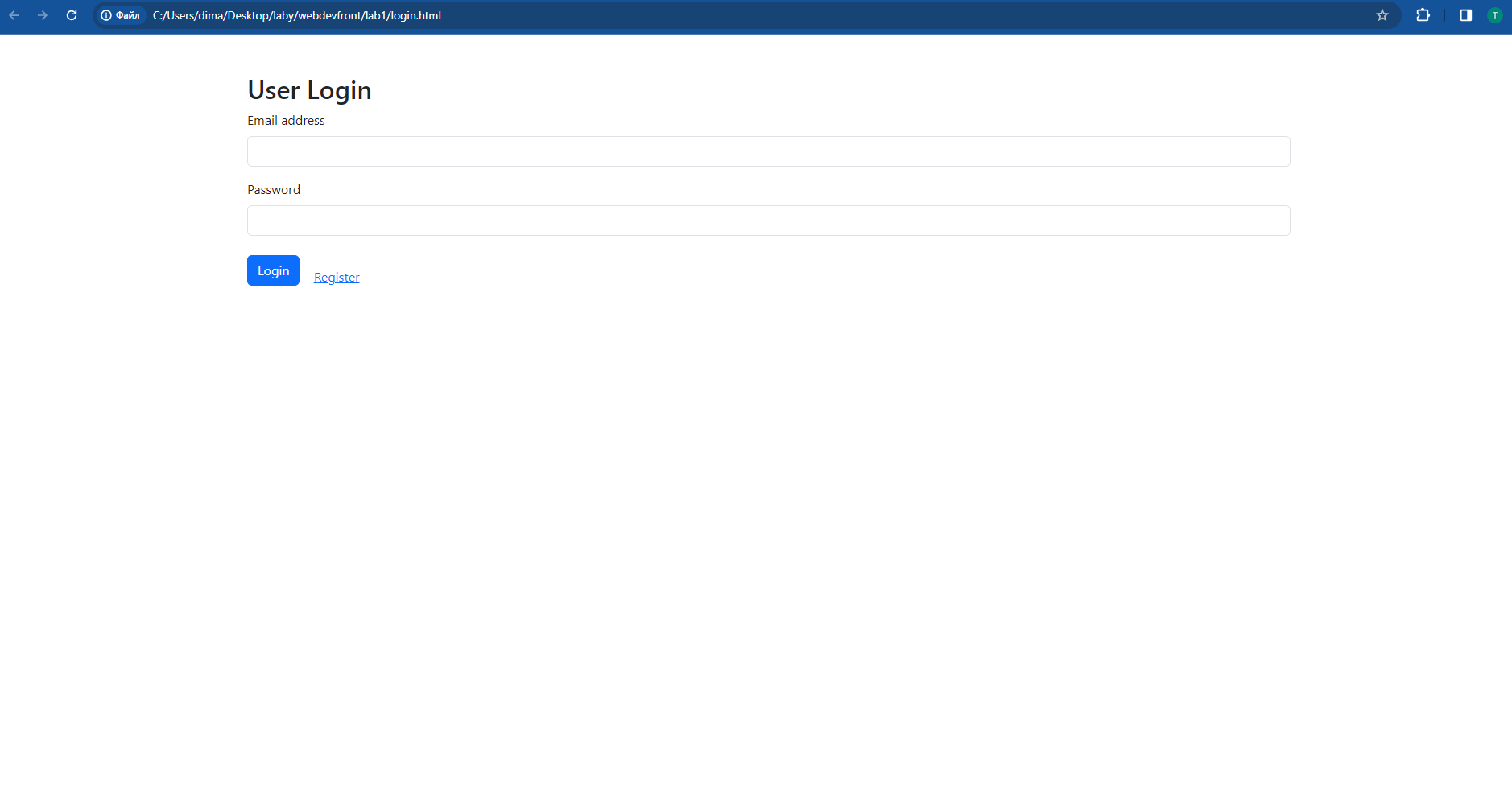
Київ – 2024

Завдання:

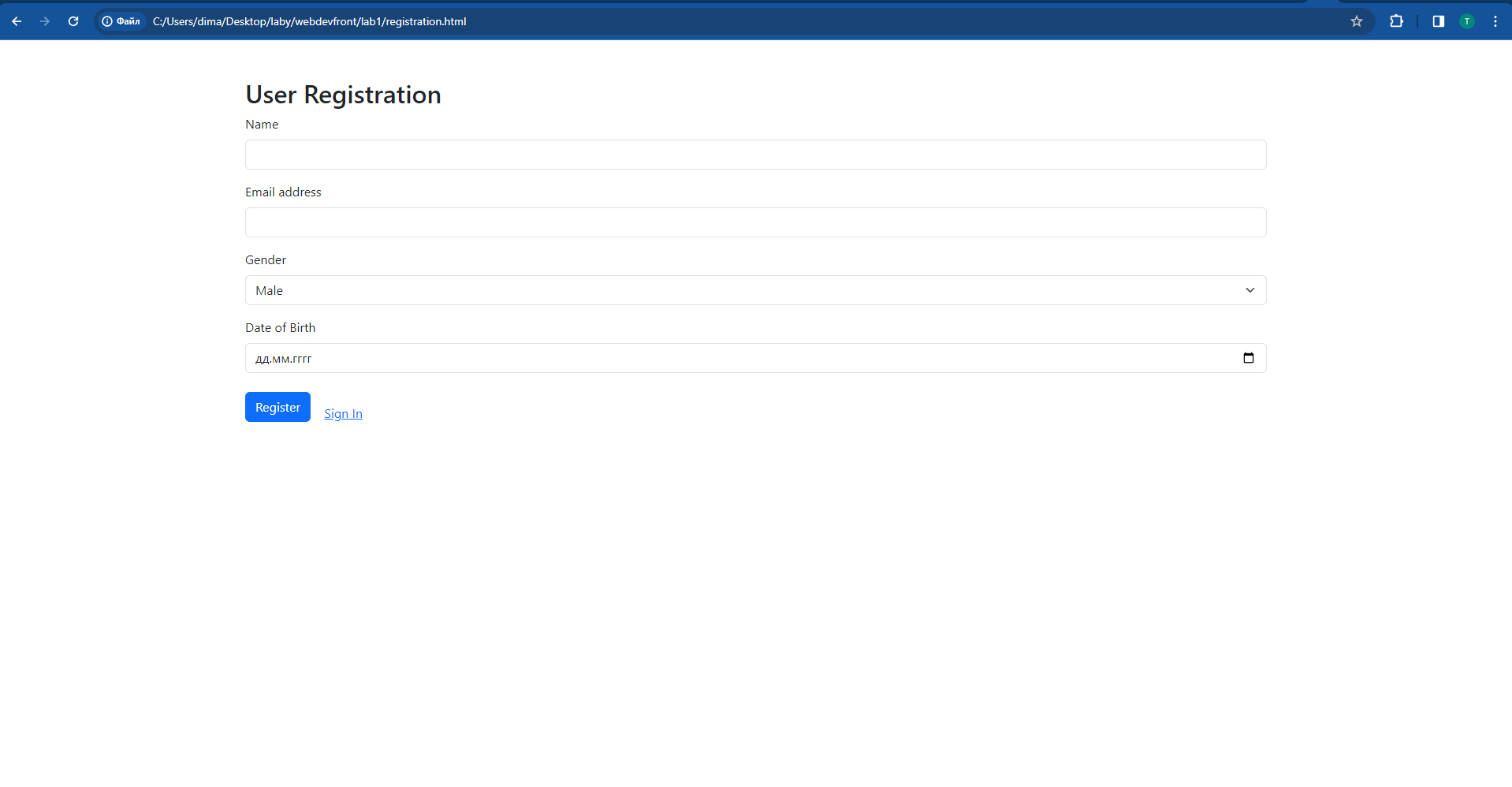
1. Запропонувати тематику Web-додатку (див. [рекомендований список](https://docs.google.com/presentation/d/11VUn3HeD7k5yySOE24mBtUQPQIdOALacPhC6qJwBZZM/edit#slide=id.gbc28739f99_0_174)).
2. Ознайомившись з [переліком обов’язкових сторінок](https://docs.google.com/presentation/d/11VUn3HeD7k5yySOE24mBtUQPQIdOALacPhC6qJwBZZM/edit#slide=id.gbccbe84ded_0_1), виконати їх кодування у вигляді HTML-сторінок:
   1. Створити [карту сайту](https://pixelfridge.digital/wp-content/uploads/2020/03/sitemap-1.jpg), а також [макети сторінок](https://en.wikipedia.org/wiki/Website_wireframe) за допомогою [diagram.net](https://app.diagrams.net/)
   2. Встановити середовище розробки (рекомендовано [VS Code](https://code.visualstudio.com/))
   3. Встановити платформу NodeJS та framework VueJS з бібліотекою [Vuetify](https://vuetifyjs.com/en/).
   4. Ознайомитись з [методичними рекомендаціями](https://docs.google.com/document/d/166zE3g4Nf-orQWLnuTHAURsTVvjkrAQ0vlVP44MwbhY/edit?usp=sharing) до лабораторної роботи
   5. Доповнити додаток базою даних (використати VueX або [Pinia](https://pinia.vuejs.org/), зберігати дані або на сервері або в localStorage) .
   6. Розробити та відлагодити Javascript/VueJS-файли.
3. За бажанням студента доповнити сайт іншими сторінками.
4. Створити репозиторій та завантажити усі файли на [github.com](https://github.com/).
5. Перевірити сторінки сайту на адаптивність до відображення на різних розмірах екрану (скористатись панеллю відлагодження Google Chrome або Mozilla Firefox).l

Тематика веб-додатку – сайт для вивчення англійських слів наразі функціонал такий що ми маємо кнопку якою ми додаємо слово до нашої бази данних і потім виводимо список цих слів для вивчення, наразі кнопка значення додає до локалсторедж лише англійське слово без перекладу надалі планую це покращити, а кнопка Gettheword виводить 5 захардкоджених значеннь в майбутьніх лабах буде виводити список слів з нашої бази данних які ми додаємо маємо такі html сторінки:

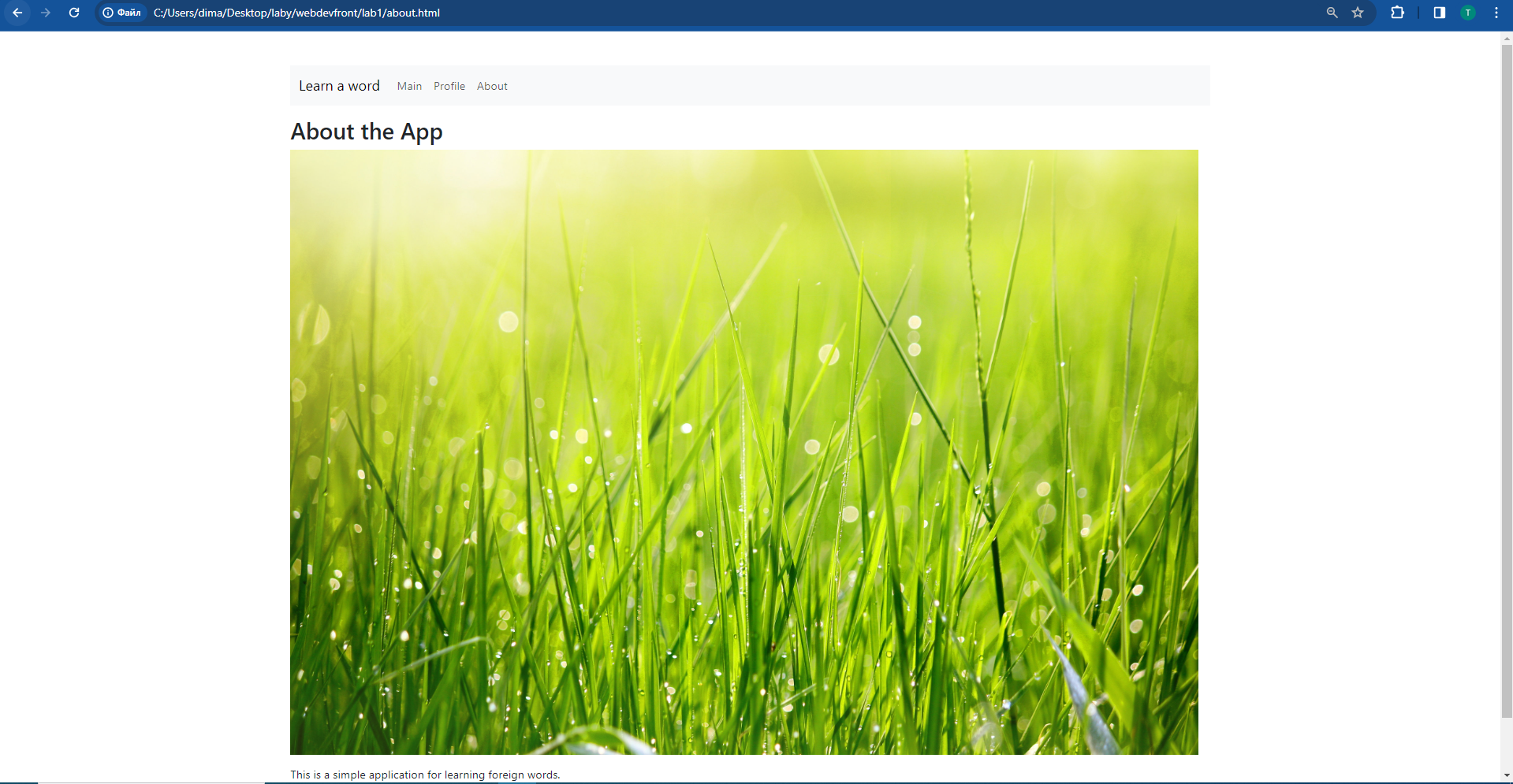
Login.html



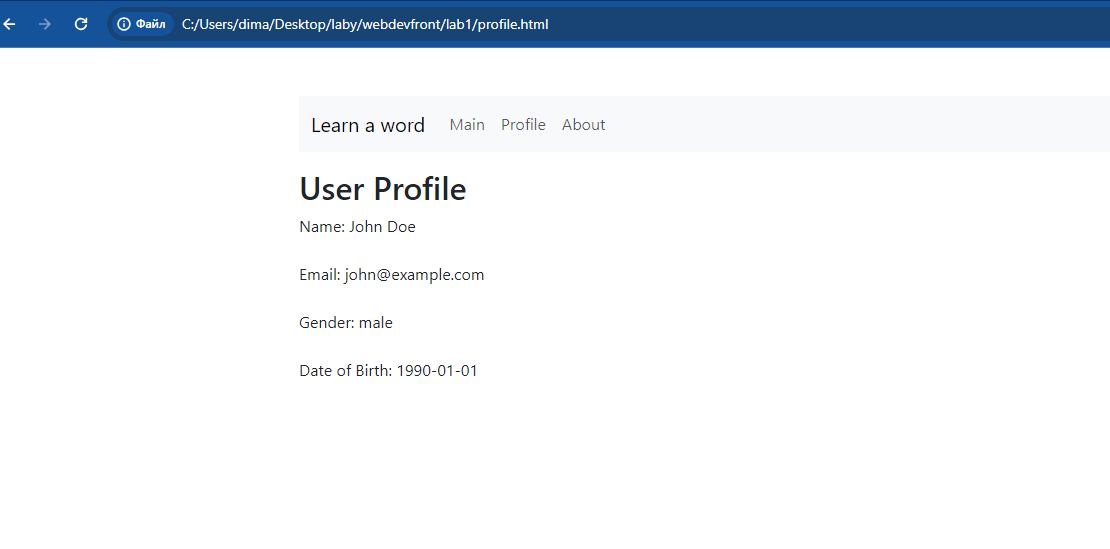
Register.html



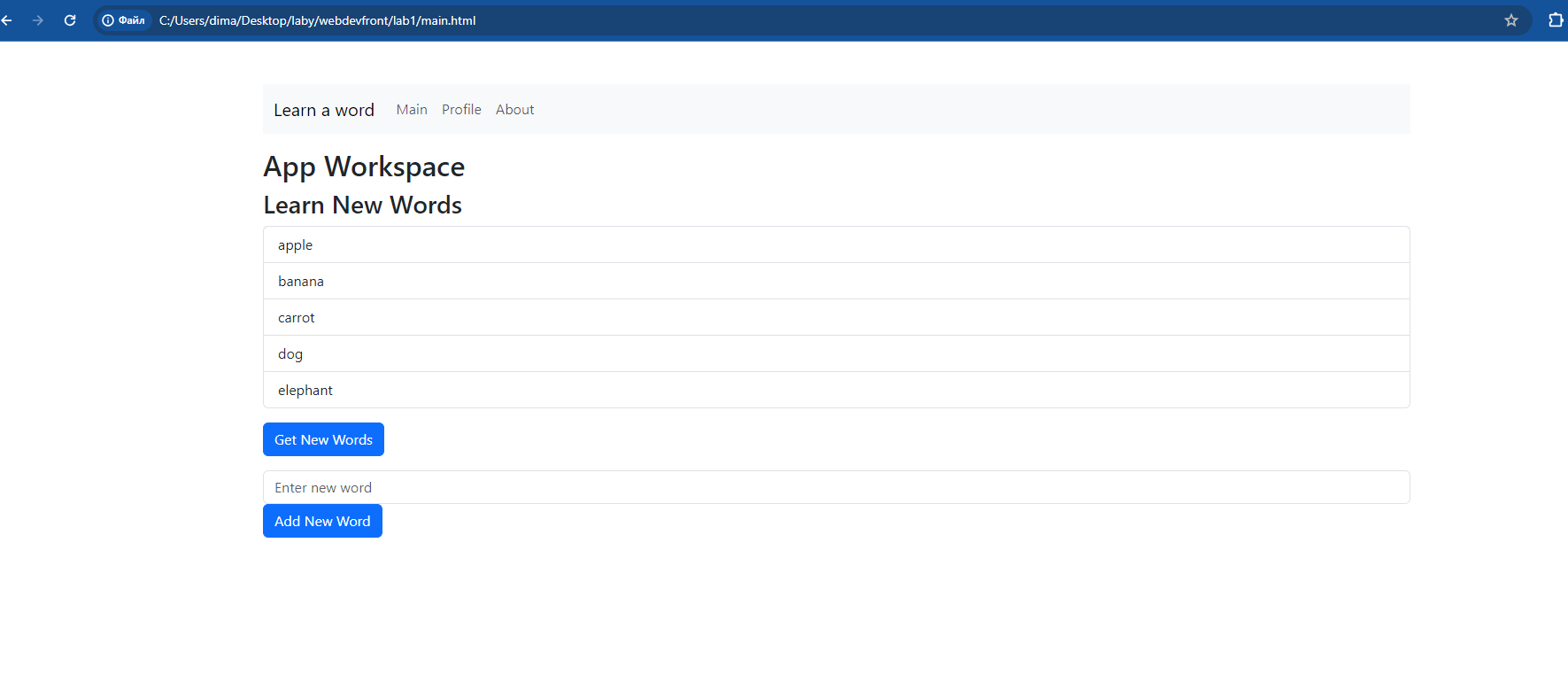
About.hmtl



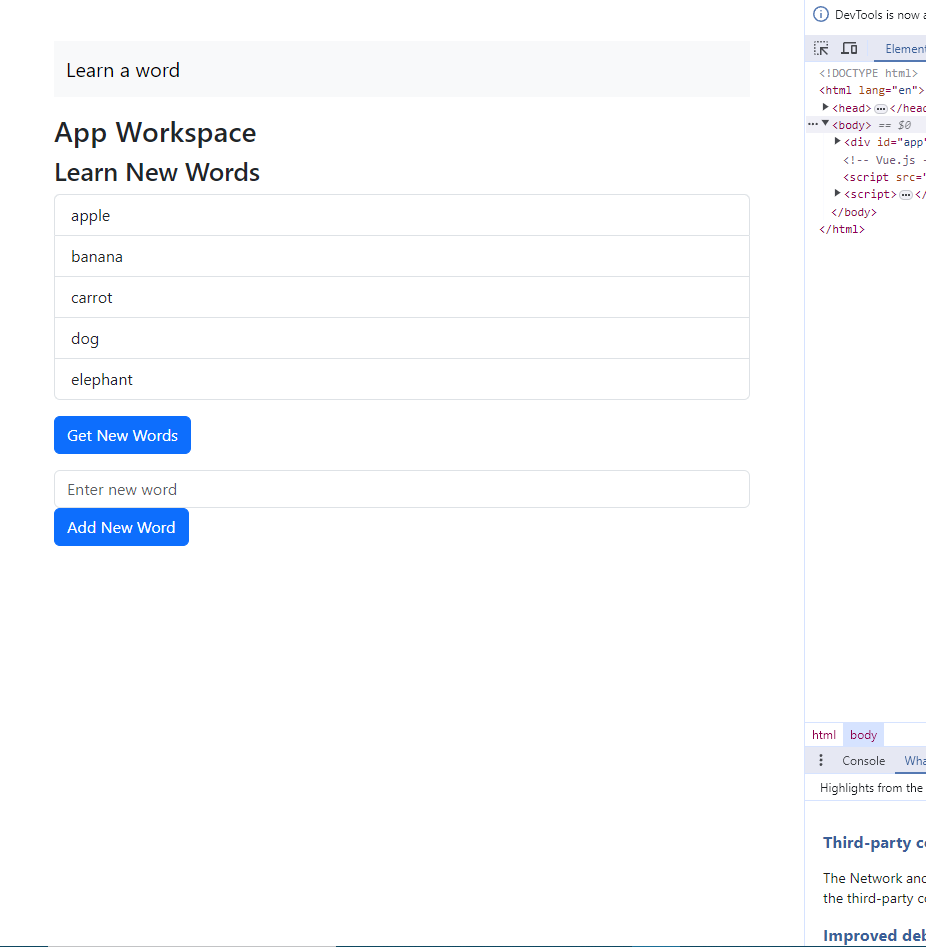
Profile.html



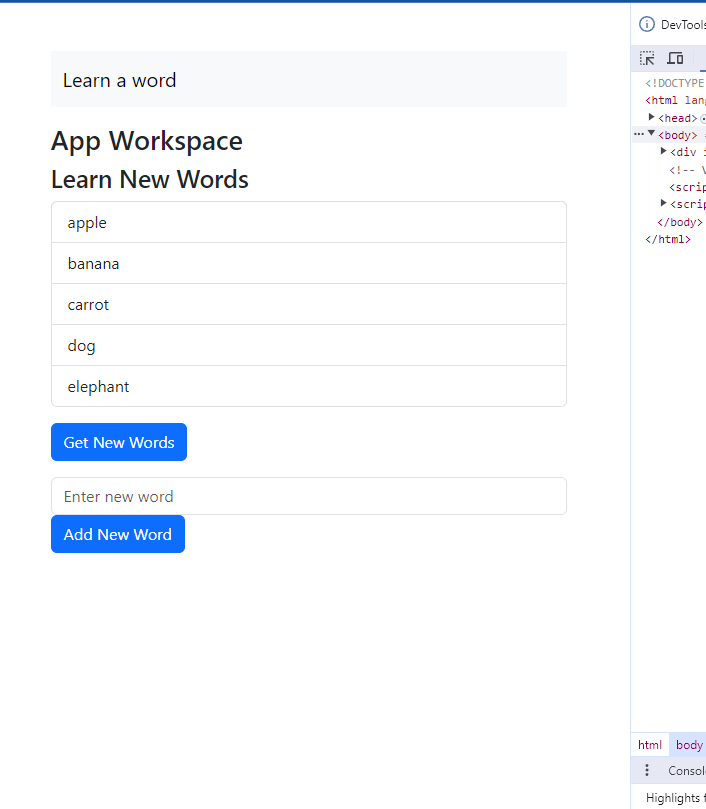
Main.html



Для планшету



Для телефону



Фрагмент коду на Vue.js

Src/App.vue

<template>

    <div id="app" class="container mt-5">

      <h2>App Workspace</h2>

      <div class="mb-3">

        <h3>Learn New Words</h3>

        <ul class="list-group">

          <li class="list-group-item" v-for="(word, index) in words" :key="index">{{ word }}</li>

        </ul>

      </div>

      <div class="mb-3">

        <input type="text" v-model="newWord" class="form-control" placeholder="Enter new word">

        <button class="btn btn-primary" @click="addWord(newWord)">Add New Word</button>

      </div>

    </div>

  </template>

  <script>

  export default {

    data() {

      return {

        words: [],

        newWord: ''

      };

    },

    created() {

      this.loadUserWords();

    },

    methods: {

      // Метод для завантаження даних з локального сховища

      loadUserWords() {

        const userWords = JSON.parse(localStorage.getItem('userWords')) || [];

        this.words = userWords;

      },

      // Метод для додавання нового слова

      addWord(wordToAdd) {

        const userWords = JSON.parse(localStorage.getItem('userWords')) || [];

        userWords.push(wordToAdd);

        localStorage.setItem('userWords', JSON.stringify(userWords));

        console.log('Слово додано:', wordToAdd);

        // Оновити список слів

        this.loadUserWords();

      }

    }

  };

  </script>

  <style>

  #app {

    font-family: Arial, sans-serif;

    text-align: center;

  }

  </style>

Component.vue

<template>

    <div>

        <input type="text" v-model="newWord">

        <button @click="addWord">Add Word</button>

        <ul>

            <li v-for="(word, index) in dictionary" :key="index">

                {{ word }}

                <button @click="removeWord(index)">Remove</button>

            </li>

        </ul>

    </div>

</template>

<script>

export default {

    data() {

        return {

            newWord: ''

        };

    },

    computed: {

        dictionary() {

            return this.$store.state.dictionary;

        }

    },

    methods: {

        addWord() {

            if (this.newWord.trim() !== '') {

                this.$store.dispatch('addWord', this.newWord);

                this.newWord = '';

            }

        },

        removeWord(index) {

            this.$store.dispatch('removeWord', index);

        }

    },

    mounted() {

        this.$store.dispatch('initDictionary');

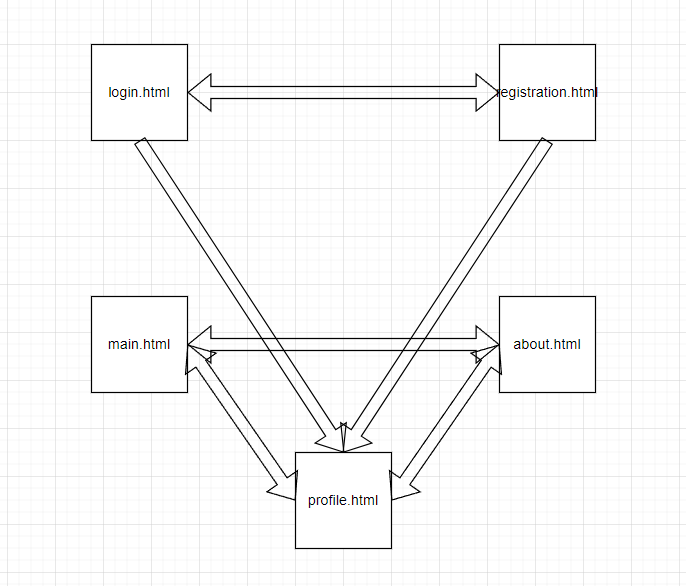
    }

};

</script>

В компоненті додані деякі функції які ще не реалізовано у 1 лабі але надалі вони будуть реалізовані та використовуватись.

Карта сайту



Нік у телеграмм @MR\_DEUS

Github: https://github.com/tetroboy/lab1webfront