СОДЕРЖАНИЕ

ВВЕДЕНИЕ……………......………………………………………………………....…3

ГЛАВА Ⅰ. СРЕДА РАЗРАБОТКИ VISUAL STUDIO CODE…………………..……..4

1.1. Настройка среды разработки в Visual Studio для создания игрового приложения на основе Windows Forms…………………………………..………………...…….…..9

1.1. Создание проекта………………………………………………..…………...…...10

1.2 Создание макета и настройка формы………………………….………………….11

**Глава 2. Интерфейс игры**…………………………………………………………...14

2.1 Добавление компонентов в форму………………………………………………16

2.2 Разработка графического интерфейса игры……………………………………...27

**Глава 3. Программирование работы игры**……………………………………….32

3.1 Кодинг внутренней части игры…………………………………………………..34

3.2 Полный код программы игры…………………………………………………….46

ЗАКЛЮЧЕНИЕ……………………...………………………………….......................51

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТЧНИКОВ……...…………………………….53

ПРИЛОЖЕНИЕ…………………………………………………………………………

# **Введение**

Современная разработка программного обеспечения для веб-сайтов играет важную роль в корпоративной среде, позволяя компаниям эффективно представлять себя в интернете, взаимодействовать с клиентами и партнёрами, а также поддерживать свои бизнес-процессы. Одним из ключевых инструментов, используемых для создания корпоративных сайтов, является Visual Studio Code — мощный и гибкий редактор кода, который предоставляет разработчикам широкий набор возможностей для написания, тестирования и отладки веб-приложений.

**Visual Studio Code (VS Code)** является бесплатным редактором исходного кода, разработанным Microsoft. Он поддерживает множество языков программирования и фреймворков, включая HTML, CSS, JavaScript, TypeScript, Python и многие другие. VS Code отличается высокой степенью настраиваемости и расширяемости благодаря интеграции с различными плагинами и расширениями, что делает его удобным инструментом для создания сложных корпоративных веб-сайтов.

**Целью** данной выпускной квалификационной работы является разработка корпоративного сайта с использованием Visual Studio Code. Основной задачей работы является изучение принципов проектирования и реализации веб-сайтов, их адаптации к требованиям конкретной компании, а также оценка эффективности Visual Studio Code в контексте создания веб-приложений.

**В результате** работы планируется создание полноценного корпоративного сайта, демонстрирующего возможности Visual Studio Code в разработке современных и функциональных веб-приложений. В ходе работы будут рассмотрены основные этапы разработки, начиная с проектирования и заканчивая реализацией ключевых функциональных элементов сайта. Особое внимание будет уделено созданию удобного и интуитивно понятного пользовательского интерфейса, обеспечению безопасности данных и оптимизации производительности веб-сайта.

**Актуальность** темы обусловлена непрерывным развитием веб-технологий и ростом требований к качеству корпоративных веб-сайтов. Использование современных инструментов разработки, таких как Visual Studio Code, позволяет создавать конкурентоспособные продукты, соответствующие современным стандартам и ожиданиям пользователей.

Таким образом,выпускная квалификационная работа представляет собой всестороннее исследование процесса разработки корпоративного сайта с применением Visual Studio Code, что позволит не только углубить теоретические знания в данной области, но и получить практический опыт, необходимый для дальнейшей профессиональной деятельности в сфере веб-разработки.

В современном мире разработка корпоративных сайтов требует использования интегрированных сред разработки (IDE), которые предоставляют удобные и мощные инструменты для создания различных типов веб-приложений. Visual Studio Code от компании Microsoft является одним из таких инструментов, предлагая разработчикам гибкое окружение для написания, тестирования и отладки кода.

Одним из ключевых преимуществ Visual Studio Code является его простота в освоении и использовании. Благодаря множеству плагинов и расширений, разработчики могут легко адаптировать среду под свои нужды, что значительно ускоряет процесс разработки и упрощает решение различных технических задач.

Использование Visual Studio Code в сочетании с современными веб-технологиями обеспечивает удобный и эффективный способ создания корпоративных сайтов. Такой подход позволяет разработчикам сосредоточиться на создании контента и логики сайта, минимизируя время, затраченное на настройку среды разработки и решение других технических вопросов.

Данная выпускная квалификационная работа направлена на исследование возможностей и применение Visual Studio Code в контексте разработки корпоративных сайтов. В работе будут рассмотрены особенности проектирования и реализации веб-приложений с использованием данной среды, а также современные методы оптимизации и архитектурные подходы для достижения высокой производительности и качественного пользовательского опыта.

Завершая введение, важно отметить, что данная выпускная квалификационная работа направлена не только на разработку конкретного корпоративного сайта, но и на изучение широкого спектра аспектов, связанных с использованием Visual Studio Code в контексте создания веб-приложений. Основываясь на анализе существующих подходов к разработке веб-сайтов и опыте использования Visual Studio Code в других областях разработки программного обеспечения, мы стремимся выявить преимущества и ограничения данной среды разработки в корпоративной веб-разработке.

Кроме того, данная работа предполагает важный вклад в научное и профессиональное сообщество, расширяя понимание о возможностях и перспективах использования Visual Studio Code в области веб-разработки. Представленный анализ и результаты исследования могут служить основой для дальнейших исследований в этой области, а также для развития новых методов и инструментов, способствующих улучшению процесса создания веб-приложений.

Таким образом, данная работа не только представляет актуальную задачу разработки корпоративного сайта с использованием Visual Studio Code, но и является важным шагом в совершенствовании практических и методологических аспектов веб-разработки, способствуя дальнейшему развитию этой динамичной области информационных технологий.

## **Методология**

В данной работе планируем создать корпоративный сайт с использованием HTML, CSS и JavaScript. В процессе разработки будут созданы основные компоненты сайта, такие как главная страница, разделы о компании, продуктах и услугах, рабочая команда организации, а также контактная информация. Графический пользовательский интерфейс будет разработан с использованием современных стандартов HTML и CSS, включая адаптивный дизайн для различных устройств и экранов. Также будет реализована интерактивная логика с помощью JavaScript, включая формы обратной связи, динамическое обновление контента и анимации.

**Целью** работы будет создание корпоративного сайта, который предоставит пользователям полную информацию о компании и её услугах, а также обеспечит удобный и приятный пользовательский опыт. Помимо этого, целью является разработка сайта, который будет не только функциональным, но и визуально привлекательным, что позволит улучшить имидж компании и привлечь новых клиентов.

**Этапы разработки:**

1. **Анализ требований и планирование:**
   * Изучение требований компании к корпоративному сайту.
   * Определение целевой аудитории и ключевых функциональных возможностей.
   * Создание плана проекта и определение основных этапов разработки.
2. **Проектирование структуры сайта:**
   * Разработка информационной архитектуры сайта, включая структуру страниц и навигацию.
   * Создание прототипов и макетов основных страниц сайта.
3. **Разработка пользовательского интерфейса:**
   * Написание HTML-кода для структуры страниц.
   * Использование CSS для стилизации элементов и создания адаптивного дизайна.
   * Разработка визуальных элементов и графики.
4. **Реализация интерактивных элементов:**
   * Использование JavaScript для добавления интерактивности и динамического контента.
   * Реализация форм обратной связи, анимаций и других пользовательских взаимодействий.
5. **Тестирование и отладка:**
   * Проведение функционального тестирования всех элементов сайта.
   * Обеспечение кроссбраузерной совместимости и адаптивности.
   * Отладка кода и исправление ошибок.
6. **Оптимизация производительности:**
   * Оптимизация загрузки страниц и графических элементов.
   * Повышение скорости работы интерактивных элементов.

**Ожидаемые результаты:**

1. Создание корпоративного сайта, полностью отвечающего требованиям компании.
2. Графически привлекательный и удобный пользовательский интерфейс.
3. Адаптивный дизайн, обеспечивающий корректное отображение на всех устройствах.
4. Интерактивные элементы, улучшающие пользовательский опыт.
5. Обеспечение безопасности и защиты данных пользователей.
6. Тщательно протестированный и оптимизированный сайт.
7. Документация по разработке и использованию сайта.
8. Получение практического опыта работы с HTML, CSS и JavaScript.
9. Углубление знаний в области веб-разработки и улучшение навыков программирования.
10. Возможность применения полученных знаний и опыта в будущих проектах.

Эти этапы и ожидаемые результаты помогут не только создать функциональный и привлекательный корпоративный сайт, но и получить ценный опыт в области веб-разработки, что станет важным вкладом в профессиональное развитие.

# **ГЛАВА I. СРЕДА РАЗРАБОТКИ VISUAL STUDIO CODE**

В рамках первой главы будут рассмотрены основные этапы настройки среды разработки в Visual Studio Code для создания корпоративного сайта.

В начале главы будет представлен обзор основных инструментов и компонентов Visual Studio Code, необходимых для работы с веб-проектами. Затем будет подробно рассмотрен процесс создания нового веб-проекта в Visual Studio Code, включая выбор расширений, настройку параметров и структуру проекта.

Далее будет рассмотрен процесс добавления веб-компонентов, таких как страницы, стили и скрипты, в проект корпоративного сайта. Будет рассмотрено использование встроенных и сторонних инструментов Visual Studio Code для создания и редактирования HTML, CSS и JavaScript файлов.

Также будет рассмотрен процесс настройки пользовательского интерфейса сайта с использованием фреймворков и библиотек, таких как Bootstrap или Materialize, для создания навигационных панелей, форм и других элементов пользовательского интерфейса.

Наконец, будет представлен обзор основных инструментов отладки и тестирования веб-приложений в Visual Studio Code, а также рассмотрены методы улучшения производительности и оптимизации проекта с использованием инструментов разработчика и расширений для анализа кода.

Эта глава позволит читателю ознакомиться с основными шагами настройки среды разработки в Visual Studio Code и подготовки к созданию корпоративного сайта.

**1. Обзор интегрированной среды разработки Visual Studio Code**

Visual Studio Code — это мощная и популярная интегрированная среда разработки (IDE), используемая для написания, редактирования, отладки и развертывания кода. Она поддерживает множество языков программирования и фреймворков, включая HTML, CSS и JavaScript, что делает её идеальной для разработки веб-приложений.

VS Code предоставляет разработчикам ряд функций, которые способствуют продуктивной и эффективной работе:

**Поддержка различных языков**: VS Code поддерживает HTML, CSS, JavaScript, TypeScript и многие другие языки.

**Расширения и плагины**: Возможность установки расширений для улучшения функциональности среды.

**Инструменты отладки**: Встроенные средства отладки, поддержка интеграции с внешними отладчиками.

**Контроль версий:** Интеграция с Git и другими системами контроля версий.

**Интеллектуальная поддержка кода**: Подсказки, автодополнение, синтаксическая подсветка, рефакторинг.

# **2. Установка и настройка Visual Studio Code**

Первым шагом для начала работы с VS Code является его установка и настройка:

**Скачивание и установка**: Посетите официальный сайт [Visual Studio Code](https://code.visualstudio.com/) и скачайте установочный файл для вашей операционной системы. Следуйте инструкциям по установке.

**Настройка окружения**: После установки откройте VS Code и настройте его под свои нужды. Вы можете изменить тему, установить шрифты, настроить привязки клавиш и другие параметры.

**Выбор темы и макета**:

* Откройте Visual Studio Code и перейдите в меню File > Preferences > Color Theme для выбора цветовой темы, которая вам больше всего подходит.
* Настройте расположение панели инструментов и боковой панели через View > Appearance.

**Установка необходимых расширений**:

* Перейдите на вкладку Extensions (расширения), нажав на иконку Extensions на боковой панели или используя сочетание клавиш Ctrl+Shift+X.
* Найдите и установите следующие расширения, полезные для веб-разработки:
  + **Live Server**: для локального сервера, позволяющего автоматически перезагружать страницы при изменении кода.
  + **Prettier - Code formatter**: для автоматического форматирования кода.
  + **ESLint**: для выявления и исправления ошибок в JavaScript коде.
  + **HTML Snippets** и **CSS IntelliSense**: для упрощения написания HTML и CSS кода.
  + **GitLens**: для улучшенной интеграции с системой контроля версий Git.

# **3. Создание нового проекта**

Для создания корпоративного сайта необходимо правильно организовать структуру проекта:

**Создание папки проекта**: Создайте новую папку для вашего проекта на локальном диске.

**Открытие папки в VS Code**: Откройте созданную папку в VS Code (File -> Open Folder).

**Инициализация проекта**: Создайте файлы `index.html`, `styles.css` и `script.js` в корневой папке проекта.

**Запуск Live Server**:

* Запустите локальный сервер, кликнув правой кнопкой мыши на index.html и выбрав Open with Live Server.
* Теперь при каждом сохранении изменений в файлах страница будет автоматически перезагружаться, позволяя видеть результаты в реальном времени.

# **4. Установка и использование расширений**

Расширения значительно расширяют возможности VS Code:

**Расширение Live Server**: Позволяет запустить локальный сервер и автоматически обновлять страницу при изменении кода.

**Расширение Prettier**: Автоматическое форматирование кода для поддержания единого стиля.

**Расширение ESLint**: Помогает выявлять и устранять ошибки в коде JavaScript.

Для установки расширений:

- Перейдите на вкладку Extensions (Ctrl+Shift+X).

- Найдите нужное расширение и нажмите кнопку Install.

# **5. Написание и управление кодом**

VS Code предоставляет мощные инструменты для написания и управления кодом:

**Редактор кода:** Подсветка синтаксиса, автодополнение, фрагменты кода.

**Обозреватель файлов**: Легкий доступ к структуре проекта и его файлам.

**Терминал**: Встроенный терминал для выполнения команд без выхода из среды разработки.

# **6. Отладка и тестирование**

VS Code поддерживает различные методы отладки и тестирования:

**Встроенная отладка**: Установка точек останова, просмотр значений переменных, пошаговое выполнение кода.

**Интеграция с внешними отладчиками**: Поддержка отладки с помощью браузеров и других инструментов.

**Средства тестирования**: Возможность написания и запуска тестов с использованием фреймворков, таких как Jest и Mocha.

# **7. Оптимизация и развертывание проекта**

После завершения разработки важно оптимизировать и развернуть проект:

**Оптимизация кода**: Минификация CSS и JavaScript, сжатие изображений, удаление неиспользуемого кода.

**Развертывание проекта**: Размещение сайта на хостинге, использование служб, таких как GitHub Pages, Netlify или Vercel.

# **Ⅱ. РУКОВОДСТВО ПО КОРПОРАТИВНЫМ ВЕБ-ПРИЛОЖЕНИЯМ В VISUAL STUDIO CODE**

Руководство по созданию корпоративных веб-приложений с использованием Visual Studio Code, мощной и гибкой среды разработки, поддерживающей широкий набор функций для веб-разработки. Visual Studio Code позволяет эффективно работать с HTML, CSS, JavaScript и другими веб-технологиями благодаря расширениям и встроенным инструментам. Характерной чертой Visual Studio Code является его расширяемость и поддержка различных языков программирования и фреймворков через Marketplace.

2.1. Создание многофункциональных интерактивных пользовательских интерфейсов

Visual Studio Code — это мощный редактор кода, который поддерживает разработку интерактивных и многофункциональных пользовательских интерфейсов для веб-приложений. С помощью различных расширений и встроенных инструментов разработчики могут быстро создавать и настраивать сложные интерфейсы, взаимодействующие с пользователем.

**1. HTML и CSS:**

- Использование расширений, таких как \*\*HTML Snippets\*\* и \*\*CSS IntelliSense\*\*, для ускорения написания кода и предотвращения ошибок.

- Подключение фреймворков, таких как \*\*Bootstrap\*\* или \*\*Tailwind CSS\*\*, для упрощения создания адаптивных и стильных интерфейсов.

**2. JavaScript и фреймворки:**

- Настройка и использование популярных JavaScript фреймворков, таких как **React**, **Angular** и **Vue.j**, с помощью соответствующих расширений и шаблонов проектов.

- Поддержка современных стандартов ECMAScript и инструментов, таких как **Babel** и **Webpack**, для трансформации и сборки кода.

**3. Интерактивные элемент:**

- Создание интерактивных элементов пользовательского интерфейса, таких как формы, модальные окна и таблицы, с использованием библиотек, таких как **jQuery** или **Axio**.

- Настройка событий и обработчиков событий для взаимодействия с пользователем, например, щелчки мыши, нажатия клавиш и отправка форм.

## **2.2. Отображение и обработка данных**

Веб-приложения часто требуют отображения и обработки данных, полученных из различных источников, таких как базы данных, API и файлы. Visual Studio Code предоставляет гибкие инструменты для работы с данными и создания динамических интерфейсов.

**1. API и AJAX:**

- Использование библиотек, таких как **Axios** или **Fetch API**, для выполнения AJAX-запросов и получения данных с удаленных серверов.

- Обработка и отображение данных с помощью JavaScript, обновление интерфейса в реальном времени без перезагрузки страницы.

**2. Работа с данными:**

- Отображение табличных данных с использованием библиотек, таких как **DataTables** или встроенные HTML таблицы, для создания динамических и интерактивных таблиц.

- Настройка фильтрации, сортировки и пагинации данных для улучшения пользовательского опыта.

**3. Привязка данных:**

- Использование фреймворков, таких как **Angular** или **Vue.js**, для создания двусторонней привязки данных между моделью и представлением.

- Автоматическое обновление интерфейса при изменении данных с использованием реактивного программирования.

**2.3. Развертывание веб-приложений**

После создания веб-приложения его необходимо развернуть на сервере или хостинге, чтобы пользователи могли получить к нему доступ через интернет. Visual Studio Code поддерживает различные методы развертывания и интеграцию с популярными хостинг-платформами.

**1. Настройка среды разработки:**

- Использование конфигурационных файлов, таких как `.env`, для управления переменными окружения и настройками приложения.

- Подключение к базам данных и другим сервисам через соответствующие драйверы и библиотеки.

**2. Развертывание на сервере:**

- Настройка и использование инструментов, таких как **Git**, для управления версиями и развертывания кода на сервере.

- Автоматизация развертывания с помощью CI/CD инструментов, таких как **GitHub Actions** или **Jenkins**.

3. **Облачные платформы**:

- Развертывание веб-приложений на облачных платформах, таких как **Heroku**, **Netlify** или **AWS**, с помощью соответствующих командных интерфейсов и интеграций.

- Настройка автоматических обновлений и мониторинга приложений для обеспечения их стабильной работы и быстрого реагирования на проблемы.

Использование Visual Studio Code для разработки корпоративных веб-приложений позволяет создать мощную и гибкую среду, поддерживающую все этапы разработки от создания интерфейса до развертывания на сервере. Следующая глава будет посвящена углубленному изучению инструментов и методик тестирования и отладки веб-приложений.

# **ГЛАВА Ⅲ. НАСТРОЙКА СРЕДЫ РАЗРАБОТКИ В VISUAL STUDIO CODE ДЛЯ КОРПОРАТИВНОГО САЙТА НА JS, CSS, HTML**

В этой главе будет рассмотрен процесс настройки среды разработки в Visual Studio Code для создания корпоративного сайта, используя JavaScript, CSS и HTML. Visual Studio Code является мощным и гибким редактором кода, который обеспечивает все необходимые инструменты для веб-разработки.

# **3.1. Создание проекта**

Создание нового проекта в Visual Studio Code является первым шагом на пути к разработке корпоративного сайта. В этом разделе подробно рассмотрены этапы создания и настройки нового проекта.

**3.1.1 Установка Visual Studio Code**

1. **Загрузка и установка**:
   * Перейдите на официальный сайт Visual Studio Code (<https://code.visualstudio.com/>).
   * Выберите версию для вашей операционной системы и скачайте установочный файл.
   * Запустите установщик и следуйте инструкциям для завершения установки.
2. **Запуск Visual Studio Code**:
   * После установки запустите Visual Studio Code.
   * Если это ваш первый запуск, настройте параметры интерфейса и темы для удобной работы.

**3.1.2 Создание нового проекта**

1. **Создание рабочей папки**:
   * Создайте папку на вашем компьютере, где будет храниться проект, например, CorporateSite.
   * Откройте эту папку в Visual Studio Code, выбрав File > Open Folder и указав путь к созданной папке.
2. **Создание базовых файлов проекта**:
   * В открытой папке создайте основные файлы проекта:
     + index.html для структуры HTML.
     + style.css для стилей CSS.
     + app.js для логики на JavaScript.

## **3.1.3 Настройка расширений Visual Studio Code**

1. **Установка необходимых расширений**:
   * Перейдите на вкладку Extensions (расширения) на боковой панели или используйте сочетание клавиш Ctrl+Shift+X.
   * Установите следующие расширения для улучшения опыта разработки:
     + **Live Server**: для запуска локального сервера и автоматической перезагрузки страниц.
     + **Prettier - Code formatter**: для автоматического форматирования кода.
     + **ESLint**: для выявления и исправления ошибок в JavaScript коде.
     + **HTML Snippets**: для удобного написания HTML кода.
     + **CSS IntelliSense**: для автозаполнения и подсказок в CSS.

#### **3.1.4 Запуск локального сервера**

1. **Запуск Live Server**:
   * Откройте index.html и нажмите правой кнопкой мыши на файл.
   * Выберите Open with Live Server. Это запустит локальный сервер, и откроется окно браузера с вашим сайтом.
   * Теперь при каждом сохранении изменений в файлах страница будет автоматически перезагружаться, позволяя видеть результаты в реальном времени.