

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на разработку

Разработка системы автоматической кодогенерации HTML и CSS на основе визуального
дизайна

СОДЕРЖАНИЕ

Введение	2
1 Термины и определения	3
2 Перечень сокращений.....	4
3 Основные сведения о разработке	5
3.1 Наименование разработки.....	5
3.2 Цель и задачи.....	5
3.3 Сведения об участниках разработки	5
3.4 Сроки разработки	6
3.5 Назначение разработки.....	6
3.5.1 Функциональное назначение	6
3.5.2 Эксплуатационное назначение.....	6
4 Описание предметной области	7
5 Требования к результатам разработки.....	11
5.1 Требования к функциональным характеристикам.....	11
5.2 Требования к показателям назначения	11
5.3 Требования к технологическому стеку	11
5.4 Требования к пользовательскому интерфейсу	12
5.5 Требования к видам обеспечения	17
5.5.1 Требования к математическому обеспечению	17
5.5.2 Требования к информационному обеспечению	17
5.5.3 Требования к метрологическому обеспечению.....	17
5.5.4 Требования к техническому обеспечению.....	17
5.6 Требования к надежности	17
5.7 Требования к безопасности	18
5.8 Требования к патентной чистоте	18
5.9 Требования к перспективам развития	18
6 Состав и содержание работ.....	19
7 Требования к документированию	21
8 Требования к приемно-сдаточным процедурам	22

Введение

Настоящий документ представляет собой техническое задание на разработку системы автоматической кодогенерации HTML и CSS на основе предоставленного визуального дизайна. Система предназначена для автоматизации рутинного процесса верстки, что позволит ускорить работу веб-разработчиков и дизайнеров, а также снизить количество ошибок, связанных с ручным написанием кода.

Настоящий документ содержит подробное описание всех аспектов проекта, включая цели и задачи разработки, требования к возможностям и интерфейсу, а также план работ.

Документ предназначен для исполнителя, который будет работать над проектом, а также для заказчика, чтобы контролировать ход разработки и убедиться, что результат работ соответствует содержанию документа.

1 Термины и определения

В данном разделе представлен список терминов и определений, содержащихся в настоящем документе.

A

A

A

2 Перечень сокращений

В данном разделе представлен перечень сокращений, используемых в настоящем документе.

ПК – персональный компьютер.

ОЗУ – оперативное запоминающее устройство.

ГБ – гигабайт.

ГГц – гигагерц.

ГОСТ – государственный стандарт.

ТЗ – техническое задание.

3 Основные сведения о разработке

3.1 Наименование разработки

Наименование данной разработки – Разработка системы автоматической кодогенерации HTML и CSS на основе визуального дизайна.

3.2 Цель и задачи

Целью разработки является создание программного средства для автоматической кодогенерации HTML и CSS, позволяющего сократить трудоёмкость и повысить точность при переводе визуального дизайна в веб-интерфейсы.

Результатом внедрения системы станет снижение временных затрат веб-разработчиков на ручную верстку макетов и повышение соответствия готовой страницы исходному дизайну. Для достижения цели необходимо:

- разработать интуитивно понятный графический интерфейс пользователя, состоящий из панели инструментов (с кнопками, блоками, окружностями и другими базовыми элементами), рабочей области (холста) и панели свойств;
- обеспечить возможность добавления элементов на холст путем перетаскивания и их интерактивного редактирования;
- реализовать панель свойств для точной настройки визуальных параметров выбранного элемента (размеры, отступы, цвета, тени, шрифты, выравнивание);
- разработать модуль генерации чистого, семантического HTML-кода и соответствующего CSS-кода, отражающего структуру и стили, созданные пользователем на холсте;
- предусмотреть функцию мгновенного копирования кода для дальнейшего использования.
- предусмотреть экспорт результатов анализа в распространённых форматах для дальнейшего использования.

3.3 Сведения об участниках разработки

Исполнителем является студент Колледжа ФГБОУ ВО «Вятский государственный университет» учебной группы ИСПк-406-52-00 Петухов Егор Игоревич.

Заказчиком является Колледж ФГБОУ ВО «Вятский государственный университет», представленный коллективом преподавателей в составе:

- Долженкова Мария Львовна – преподаватель по учебной практике УП.03.

3.4 Сроки разработки

Начало разработки – 18.09.2025.

Конец разработки – XX.11.2025.

3.5 Назначение разработки

В данном разделе описаны функциональное и эксплуатационное назначение разрабатываемой системы.

3.5.1 Функциональное назначение

Функциональное назначение ПО заключается в автоматизации процессов визуального проектирования веб-интерфейсов и генерации кода: предоставление инструмента для интуитивного создания макетов веб-страниц с помощью перетаскивания готовых элементов, визуального редактирования их свойств (позиционирование, размеры, стили), а также формирование семантически верного HTML-кода и соответствующего CSS-кода для дальнейшего использования в разработке.

3.5.2 Эксплуатационное назначение

Система может быть использована веб-дизайнерами, фронтенд-разработчиками, проектными менеджерами и студентами соответствующих специальностей для быстрого прототипирования пользовательских интерфейсов, обучения основам веб-вёрстки, а также для ускорения процесса разработки на этапе создания HTML/CSS макетов веб-страниц и их компонентов.

4 Описание предметной области

Создание веб-интерфейсов является фундаментальной задачей в процессе разработки веб-приложений и сайтов. Однако процесс ручной верстки по готовым макетам остается трудоемким и требует от разработчика значительных временных затрат и внимания к деталям. С широким распространением сложных и адаптивных пользовательских интерфейсов необходимость в эффективных инструментах их создания стала как никогда актуальной.

Традиционный подход к веб-разработке, при котором дизайнер создает визуальный макет в графическом редакторе, а фронтенд-разработчик вручную воспроизводит его на языке разметки и стилей, сопряжен с несколькими проблемами. Этот процесс может занимать от нескольких часов до нескольких дней, является подверженным человеческим ошибкам и часто приводит к незначительным, но критичным расхождениям между макетом и готовой страницей.

С появлением визуальных конструкторов и систем, подобных Figma, дизайн-процесс был значительно оптимизирован. Однако разрыв между этапом дизайна и этапом готовой верстки до сих пор существует. Разработчикам по-прежнему приходится вручную преобразовывать статичные макеты или, в лучшем случае, копировать отдельные CSS-свойства из инструментов дизайнера.

В наше время автоматизация кодогенерации стала ключевым направлением для ускорения веб-разработки и снижения ее порога входа. Именно здесь на передний план выходит роль систем автоматической генерации HTML и CSS кода.

Система автоматической кодогенерации HTML и CSS – это специализированное программное обеспечение, предназначенное для преобразования визуального представления интерфейса, созданного пользователем, в готовый для использования код.

Внедрение данного программного обеспечения преследует следующие цели:

- ускорение процесса создания прототипов и готовых макетов веб-страниц за счет визуального конструирования;
- автоматизация генерации семантически правильной HTML-разметки и соответствующего CSS-кода;
- снижение количества ошибок и расхождений с исходным замыслом за счет прямого визуального контроля на этапе проектирования;
- предоставление инструмента для обучения основам веб-вёрстки через визуальное взаимодействие;
- сокращение временных затрат на рутинную верстку и концентрация на решении более сложных логических задач.

Обзор аналогов

На рынке существует ряд, которые могут быть использованы в качестве аналогов.

Аналог №1 – «Webflow»

Webflow – это визуальная веб-разработческая платформа, позволяющая создавать сайты через визуальный редактор с генерацией чистого кода.

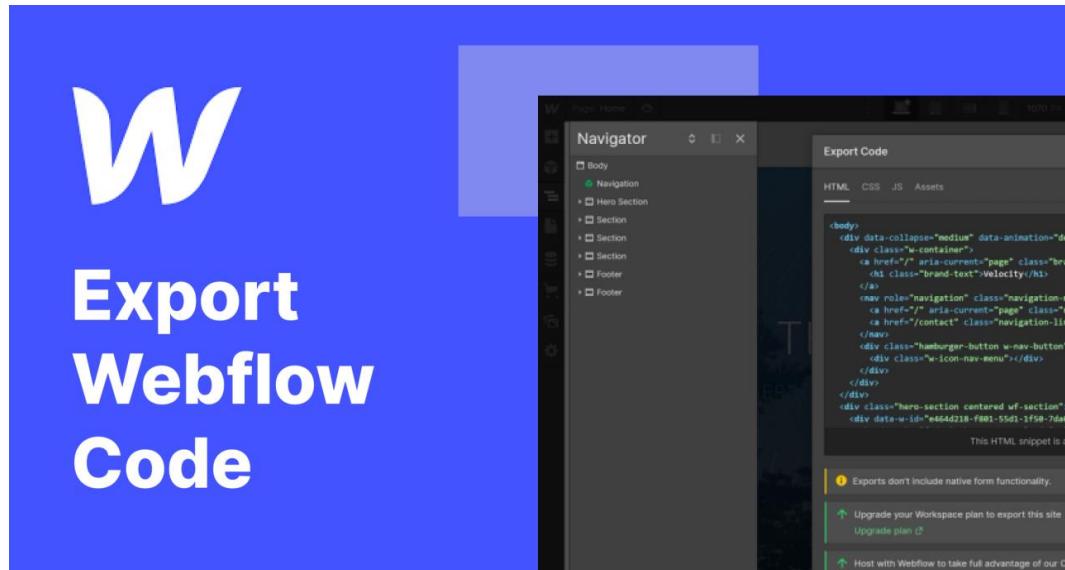


Рисунок 4.1 – Интерфейс конструктора Webflow

Преимущества:

- Мощный визуальный редактор с реальным отображением изменений
- Генерация семантического и адаптивного кода
- Экспорт готового HTML/CSS/JS

Недостатки:

- Платная подписка для экспорта кода
- Сложность обучения для новичков
- Привязка к экосистеме Webflow

Аналог №2 – «Bootstrap Studio»

Bootstrap Studio – это десктопное приложение для визуального создания веб-сайтов на основе Bootstrap с возможностью экспорта кода.



Рисунок 4.2 – Рабочее окружение Bootstrap Studio

Преимущества:

- Специализация на Bootstrap framework
- Богатая библиотека готовых компонентов и шаблонов
- Локальная работа без обязательной подписки
- Визуальное редактирование CSS свойств
- Предпросмотр в реальном времени

Недостатки:

- Сильная привязка к Bootstrap архитектуре
- Ограниченнная гибкость для кастомных решений без Bootstrap
- Требует знания структуры Bootstrap для эффективной работы
- Менее подходит для проектов, не использующих Bootstrap

Аналог №3 – «Pinegrow»

Pinegrow – это десктопный визуальный редактор для создания адаптивных веб-сайтов с возможностью работы с HTML, CSS, Bootstrap, Tailwind CSS и WordPress.



Рисунок 4.3 – Интерфейс редактора Pinegrow

Преимущества:

- Поддержка нескольких CSS фреймворков (Bootstrap, Tailwind CSS)
- Прямое редактирование HTML и CSS с визуальным отображением
- Многостраничное редактирование и мастер-страницы
- Локальная установка без подписки
- Интеграция с WordPress

Недостатки:

- Высокая стоимость лицензии
- Сложный интерфейс для начинающих
- Требует базовых знаний HTML/CSS для эффективной работы
- Ориентация на профессиональных разработчиков
- Отсутствие бесплатной версии

5 Требования к результатам разработки

В этом разделе описываются требования к разработке. Программное обеспечение должно соответствовать представленным в данном разделе критериям.

5.1 Требования к функциональным характеристикам

Программа должна осуществлять выполнение следующих функций:

- предоставление палитры базовых элементов интерфейса (кнопки, текстовые блоки, контейнеры, изображения, поля ввода, геометрические фигуры);
- добавление элементов на рабочую область методом перетаскивания;
- визуальное редактирование свойств элементов на холсте (изменение положения, размеров, порядка слоев);
- изменение визуальных свойств элементов через панель параметров (цвет фона, цвет текста, размеры, отступы, шрифты, границы, тени);
- генерация семантически корректного HTML-кода на основе созданной структуры;
- генерация соответствующих CSS-стилей, включая свойства позиционирования и внешнего вида;
- копирование сгенерированного кода в буфер обмена;

5.2 Требования к показателям назначения

Информационная система должна обеспечивать следующие показатели назначения:

- время на полный запуск системы не должно превышать 5 секунд;
- время отклика при выполнении основных операций (добавление элемента на холст, изменение его свойств, перерисовка интерфейса) не должно превышать 100 миллисекунд;
- система должна стабильно работать с проектами, содержащими до 100 различных элементов интерфейса на одном холсте;
- время генерации кода для проекта стандартной сложности (до 50 элементов) не должно превышать 3 секунд;

5.3 Требования к технологическому стеку

При разработке программы должен использоваться следующий технологический стек:HTML, CSS, JavaScript.

Иные требования к технологическому стеку не предъявляются.

5.4 Требования к пользовательскому интерфейсу

Пользовательский интерфейс должен быть интуитивно понятен и обеспечивать все необходимые возможности программы.

При запуске программы открывается стартовый экран с возможностью создать новый аккаунт. Прототип окна регистрации экрана представлен на рисунке 5.4.1.

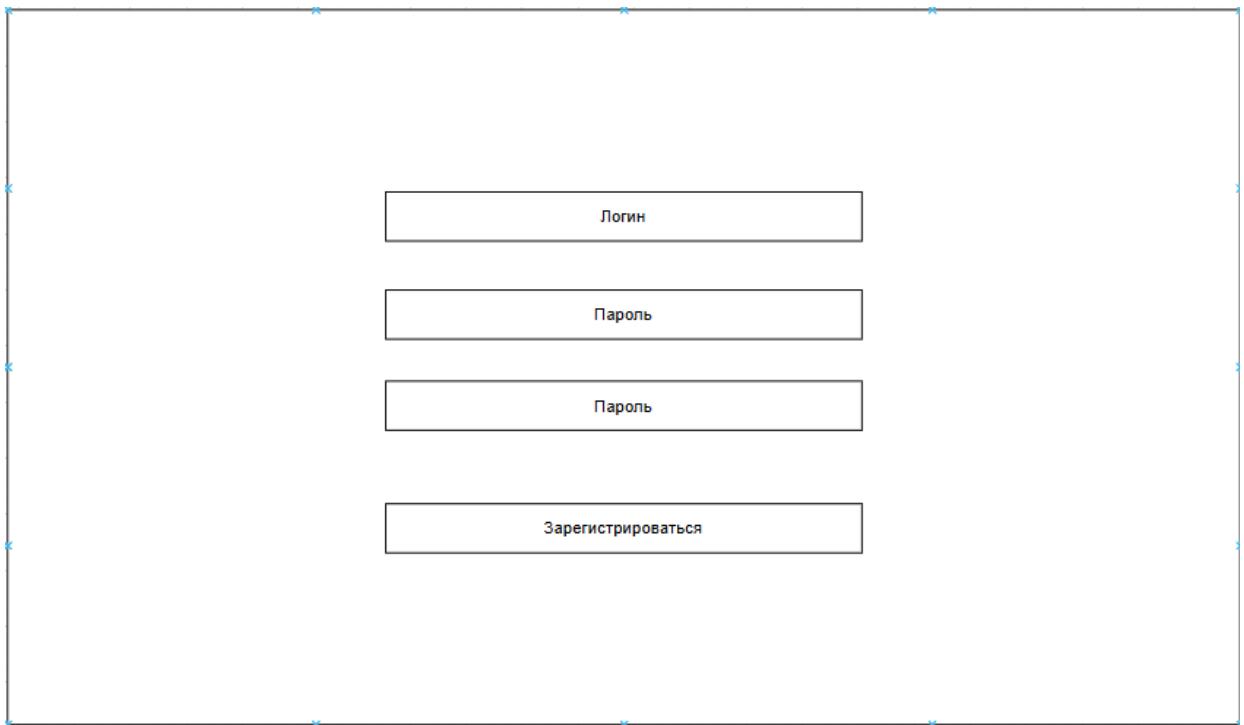


Рисунок 5.4.1 – Прототип экранной формы окна регистрации

На форме регистрации предоставляются следующие элементы:

- Поле «Логин» – для ввода уникального имени пользователя.
- Поле «Пароль» – для создания пароля учетной записи.
- Поле «Подтверждение пароля» – для повторного ввода пароля с целью исключения ошибки.
- Кнопка «Зарегистрироваться» – для отправки данных и создания новой учетной записи.

Для доступа к системе существует окно аутентификации пользователя. Прототип экранной формы окна входа представлен на рисунке 5.4.2.

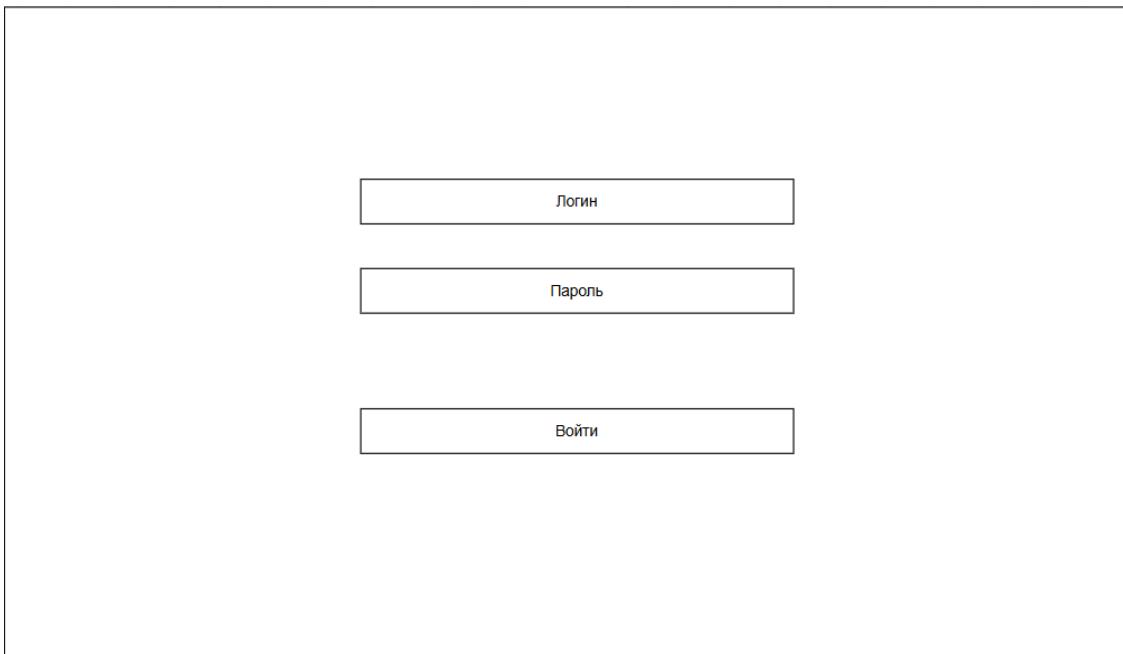


Рисунок 5.4.2 – Прототип экранной формы стартового окна

Форма входа в систему содержит минимально необходимые элементы:

- Поле «Логин» – для ввода имени пользователя.
- Поле «Пароль» – для ввода пароля учетной записи.
- Кнопка «Войти» – для отправки учетных данных и авторизации в системе.

При успешном входе в программу , открывается стартовый экран с возможностью создать новый проект. Прототип окна стартового экрана представлен на рисунке 5.4.3.

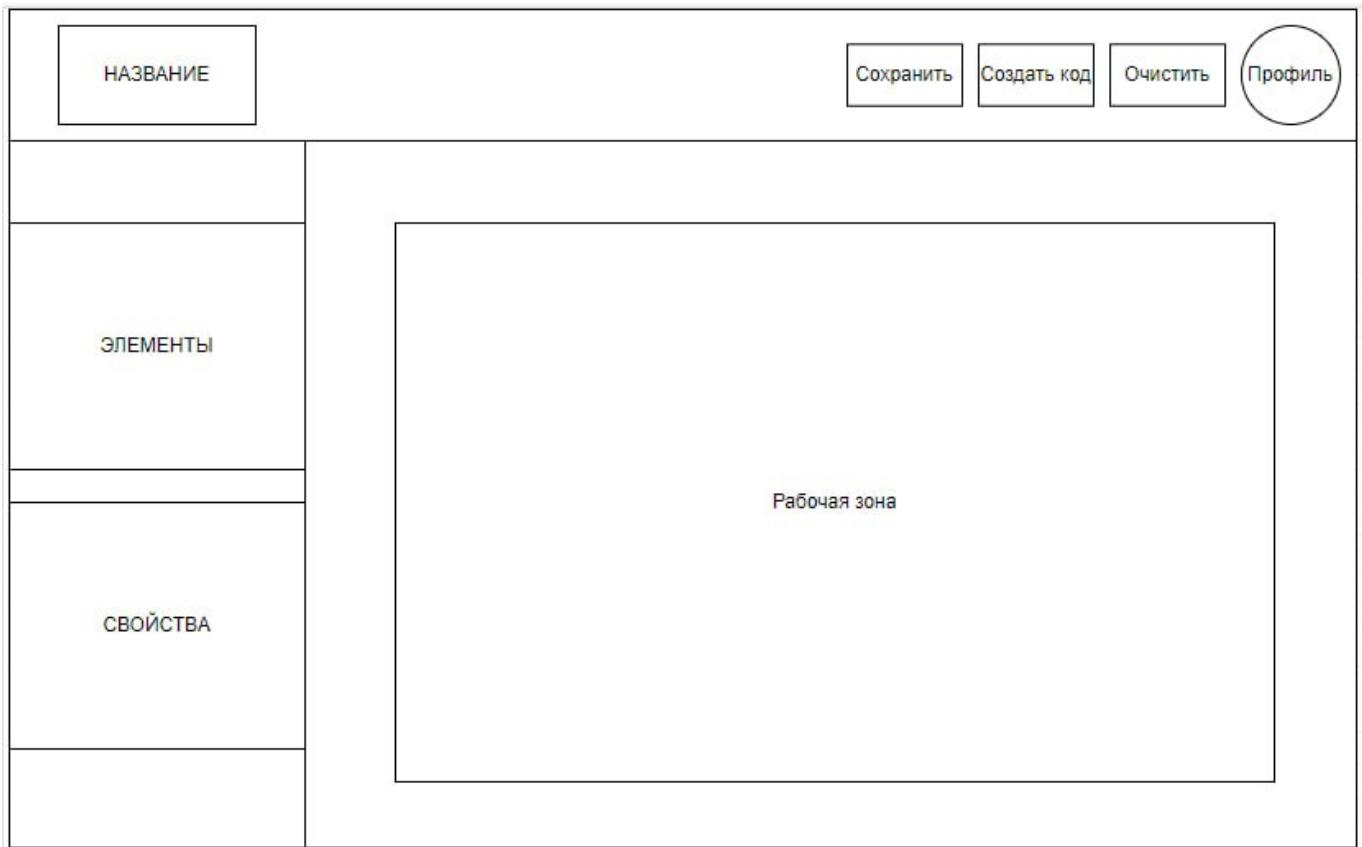


Рисунок 5.4.3 – Прототип экранной формы стартового окна

На стартовом экране предоставляются возможности:

- Панель элементов – содержит базовые компоненты для построения интерфейса (кнопки, текстовые блоки, изображения).
- Рабочая область – зона для визуального проектирования макета.
- Панель свойств – позволяет настраивать параметры выбранного элемента (цвет, размер, шрифты).
- Кнопки для управления проектом: «Сохранить», «Сгенерировать код», «Очистить рабочую область».
- Кнопка «Профиль» – предоставляет доступ к истории работы.

После завершения работы над дизайном пользователь может сгенерировать код, нажав кнопку "Сгенерировать код". При этом открывается окно генерации кода, прототип которого представлен на рисунке 5.4.3.

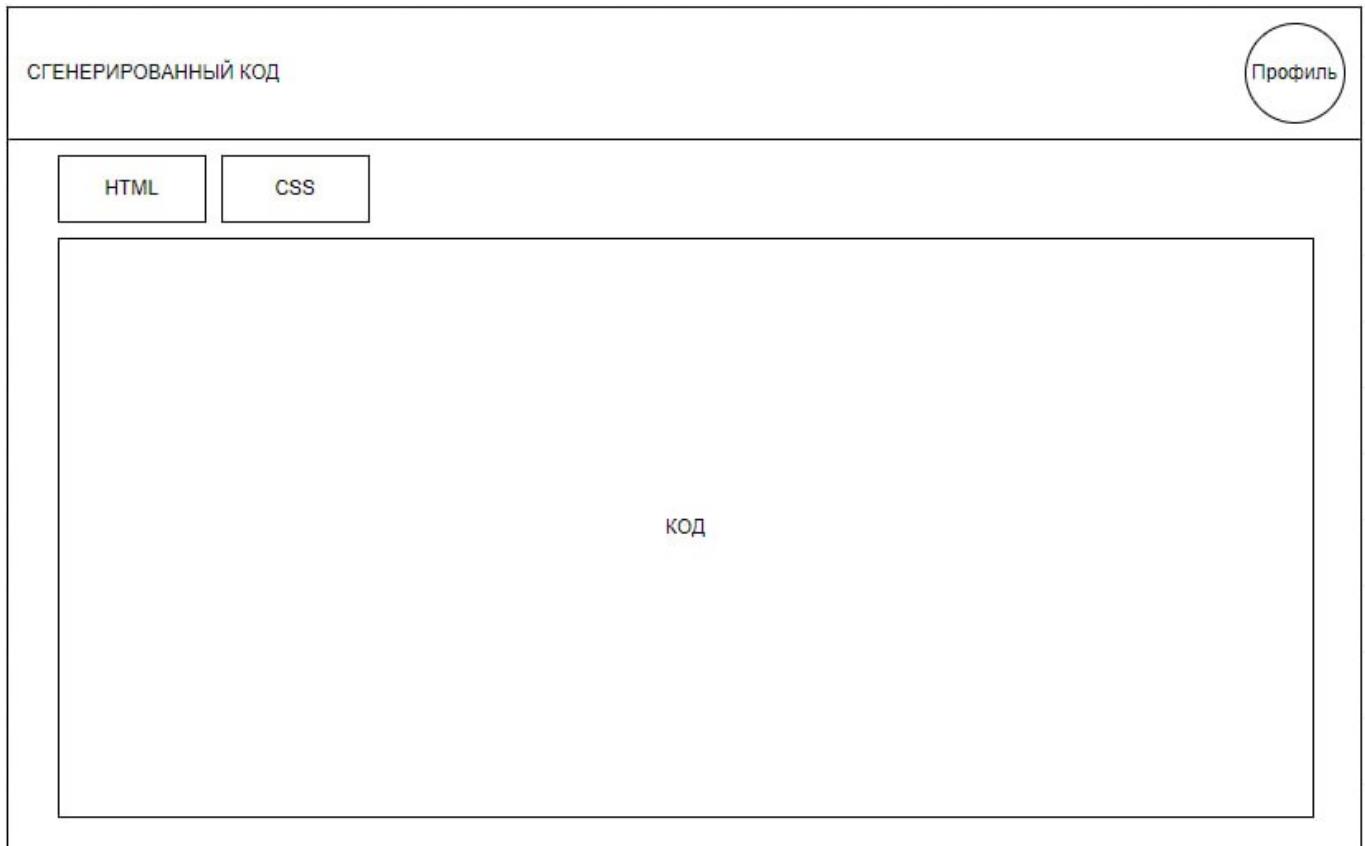


Рисунок 5.4.4 – Прототип окна генерации кода

В окне генерации кода представлены:

- две вкладки для просмотра сгенерированного кода: HTML и CSS;
- область отображения соответствующего кода для каждой вкладки;
- возможность копирования сгенерированного кода в буфер обмена.

При нажатии на кнопку «Профиль» в основном интерфейсе открывается окно управления проектами. Прототип экранной формы окна со списком проектов представлен на рисунке X.X.N.

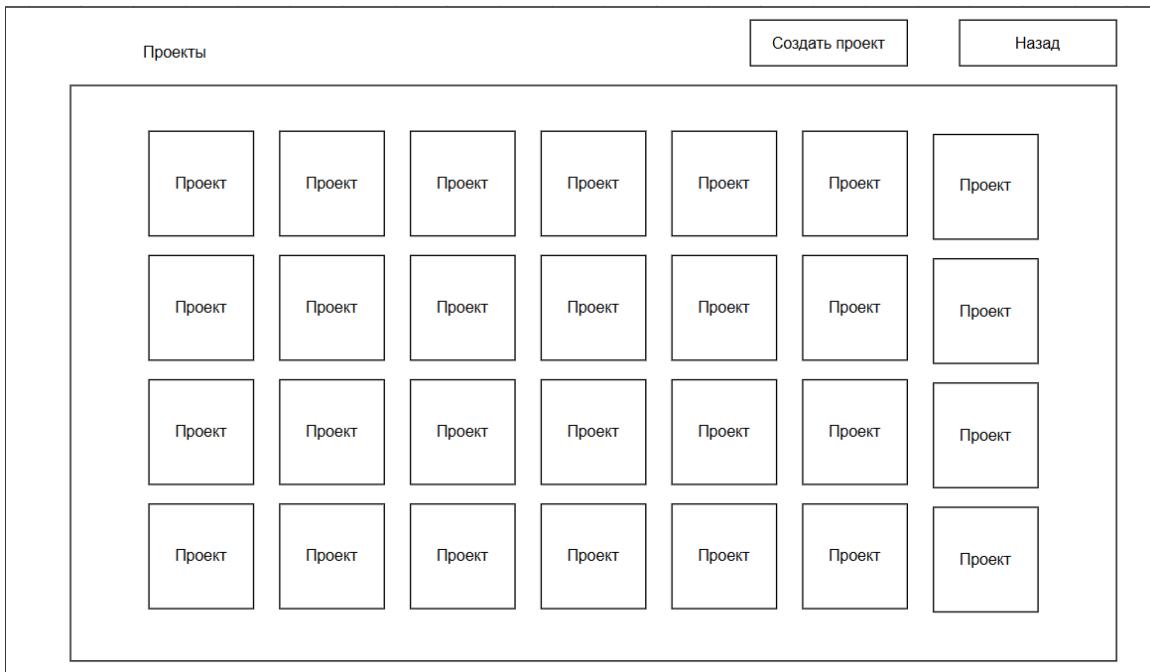


Рисунок 5.4.5 – Прототип экранной формы окна управления проектами

Окно управления проектами предоставляет пользователю доступ ко всем его сохранённым работам и включает следующие элементы:

- Список проектов – область, содержащая интерактивные кнопки с названиями всех ранее сохраненных проектов пользователя.
- Кнопка «Создать новый проект» – позволяет вернуться в главный редактор для начала новой работы.
- Кнопка «Назад» – обеспечивает возврат в основное рабочее окно программы без выбора проекта.

Для загрузки любого из сохранённых проектов пользователю достаточно выбрать соответствующую кнопку из списка, после чего проект будет автоматически загружен в рабочую область редактора.

5.5 Требования к видам обеспечения

В данном подразделе содержатся требования к видам обеспечения разработки.

5.5.1 Требования к математическому обеспечению

Требования к математическому обеспечению не предъявляются.

5.5.2 Требования к информационному обеспечению

В данном пункте содержатся требования к информационному обеспечению разработки.

5.5.2.1 Требования к форматам хранения данных

Требования к форматам хранения данных не предъявляются.

5.5.2.2 Требования к лингвистическому обеспечению

Информационная система должна обеспечивать полную поддержку русского языка, включая локализацию интерфейса.

5.5.3 Требования к метрологическому обеспечению

Требования к метрологическому обеспечению не предъявляются.

5.5.4 Требования к техническому обеспечению

Минимальные технические требования к запуску ИС включают следующие компоненты персонального компьютера:

- процессор с частотой 2ГГц;
- ОЗУ 8 ГБ;
- монитор;
- мышь или тачпад;
- клавиатура;
- операционная система Windows 11.

5.6 Требования к надежности

Надежное функционирование ИС должно быть обеспечено выполнением организационно-технических мероприятий, указанных ниже:

- организацией бесперебойного питания технических средств;

- осуществлением контроля входных данных;
- регулярным следованием рекомендациям Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998 г. «Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств».

5.7 Требования к безопасности

Безопасное функционирование программы должно обеспечиваться комплексом организационно-технических мер, соответствующих ГОСТ 51188-98 «Защита информации. Испытания программных средств на наличие компьютерных вирусов».

Программа является клиентской веб-технологией и работает исключительно в браузере пользователя. Все данные (проекты, элементы интерфейса, настройки) хранятся локально на устройстве пользователя и не передаются на внешние серверы.

Программа должна обеспечивать конфиденциальность информации, исключая её передачу третьим лицам без согласия пользователя. При экспорте кода вся работа выполняется локально, без доступа к сети.

5.8 Требования к патентной чистоте

Программа не должна использовать результаты интеллектуального труда сторонних субъектов и нарушать права третьих лиц в соответствии с частью четвёртой Гражданского кодекса Российской Федерации от 18.12.2006 № 230-ФЗ.

В случае использования внешних библиотек или сервисов необходимо:

- применять их исключительно в рамках действующих лицензий или пользовательских соглашений;
- указывать авторство и условия использования в сопровождающей документации;
- обеспечивать, чтобы использование данных решений не нарушило патентную чистоту разрабатываемого программного обеспечения.

5.9 Требования к перспективам развития

Программа может иметь следующие направления дальнейшего развития:

- расширение библиотеки базовых элементов интерфейса за счёт добавления сложных компонентов (навигационные меню, таблицы, карусели изображений);
- добавление поддержки генерации JavaScript кода для интерактивных элементов;
- реализация функции совместной работы над проектом в реальном времени.

6 Состав и содержание работ

В рамках разработки программного продукта в соответствии с настоящим документом необходимо выполнить перечень работ, представленных в таблице 1.

Таблица 1 – Состав и содержание работ

№ этапа	Наименование этапа	Длительность	Состав работ	Результат
1	Изучение предметной области и аналогов	1 неделя	Анализ предметной области, изучение существующих программ для анализа и обработки речи, выявление их достоинств и недостатков	Сформировано представление о функционале будущей программы
2	Составление Технического задания	1 неделя	Разработка и утверждение ТЗ, определение требований к функционалу и интерфейсу	Утверждённое техническое задание
3	Проектирование	2 недели	Разработка архитектуры приложения, описание алгоритмов анализа речи, подготовка макетов интерфейса	Спроектированная структура ПО и макеты интерфейса
4	Реализация программного обеспечения	5 недель	Написание исходного кода программы, реализация функций импорта аудио, спектрального анализа и поиска звуковых сочетаний	Рабочий прототип программы
5	Тестирование программы	1 неделя	Проведение функционального тестирования, выявление ошибок и недоработок	Список ошибок и предложений по доработке
6	Доработка программы	2 недели	Исправление выявленных ошибок, оптимизация работы интерфейса и алгоритмов	Исправленная версия программы
7	Составление отчетной документации	1 неделя	В ходе работ происходит разработка и утверждение отчетной документации	Готовая отчётная документация

8	Приемно-сдаточные испытания	1 неделя	Представление программы и документации комиссии, внесение результатов испытаний в ведомость	Итоговая оценка
---	-----------------------------	----------	---	-----------------

7 Требования к документированию

Состав программной документации включает:

- техническое задание;
- руководство пользователя.

8 Требования к приемно-сдаточным процедурам

Испытания проводятся комиссией, в лице заказчика, которым является Колледж ФГБОУ ВО «Вятский государственный университет», представленный преподавателем учебной практики УП.03 – Долженковой Марией Львовной.

Во время испытаний комиссия проверяет работу программы в соответствии со следующими позициями:

- набор функциональных тестов;
- корректное функционирование заданных в техническом задании функций;
- возможность функционирования на ПК с указанными минимальными системными требованиями.

Комиссии должны быть представлены эксплуатационные документы, разработанная программа и доклад.