**Введение**

На учебной практике была выбрана цель, разработать web-ресурс на тему: «Трудоустройство 1С программистов «1CJob»».

Цель проекта заключается в создании многофункционального и удобного в использовании программного инструмента для создания резюме. Основная цель — создание резюме и поиск работы для 1C программистов.

Далее приведём краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ задачи». Он посвящён изучению предметной области и организационно-экономической сущности поставленной задачи. Также в нем описано, как задача решается в настоящее время, перечислены входные и выходные данные. Также в этом разделе вы сможете узнать о том, как данная задача решается в настоящее время. Все входные и выходные данные тоже будут описаны в первом разделе. В подразделе «Инструменты разработки» будет рассмотрена среда, в которой создаётся данный проект. Здесь также будут установлены минимальные и оптимальные требования к аппаратным характеристикам, обеспечивающим правильное функционирование поставленной задачей.

В разделе «Проектирование задачи» будут рассмотрены основные аспекты разработки web-ресурса. Здесь можно будет узнать об организации данных в контексте среды разработки. В данном разделе будет чётко описан пользовательский интерфейс, составлены алгоритмы процесса обработки информации.

«Реализация задачи» – это третий раздел отчёта по практике, в котором описываются все элементы и объекты, которые будут использованы при реализации данного приложения. В этом разделе будут чётко описаны функции пользователя и их структура. Описано руководство программиста и будет предоставлена диаграмма компонентов.

Четвёртый раздел – «Тестирование». В нем будет описано полное и функциональное тестирование данной программы, т.е. будет оттестирован каждый пункт меню, каждая операция, которая выполняется web-ресурсом. Будут смоделированы все возможные действия пользователя при работе с web-ресурсом, начиная от входа на сайт заканчивая закрытием вкладки.

В разделе «Применение» будет описано назначение, область применения, среда функционирования данного программного продукта.

«Заключение» будет содержать краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств, описание степени автоматизации процессов на различных этапах разработки.

В разделе «Список используемых источников» будет приведён список используемых при разработке источников.

В приложениях к пояснительной записке будет приведены UX и UI проектирование сайта и диаграммы.

**1 Анализ задачи**

**1.1 Постановка задачи**

**1.1.1 Организационно-экономическая сущность задачи**

Темой данного проекта является разработка информационного интернет-ресурса «1CJob» для нахождения работы 1С программистам.

Целью разработки данного программного продукта является создание интернет-ресурса для нахождения работы 1С программистам с удобным и понятным интерфейсом.

Источники и способы получения данных:

* регистрация пользователей: данные о пользователях, необходимые для регистрации (электронная почта, пароль), получаются через процесс регистрации на сайте;
* заполнение анкет: ппользователи вводят данные в свои анкеты, предоставляя информацию о своем опыте, навыках, образовании и других релевантных аспектах;
* модерация данных администраторами: аадминистраторы сайта взаимодействуют с системой для модерации данных, подтверждения регистраций, и при необходимости редактирования или удаления анкет.

Информационная связь с другими задачами:

* База данных анкет: информационная связь с базой данных анкет пользователей. Все данные, введенные пользователями при регистрации и заполнении анкет, будут храниться в базе данных;
* Поиск и фильтрация: возможность поиска и фильтрации анкет для компаний. Здесь информационная связь может включать механизмы поиска по ключевым навыкам, опыту работы, образованию и другим параметрам;
* Управление профилем пользователя: личные кабинеты для пользователей и компаний, где они могут управлять своими профилями, просматривать и редактировать анкеты;
* Система аутентификации и безопасности: информационная связь с системой аутентификации и безопасности для обеспечения защиты личных данных пользователей;
* Статистика и отчетность: возможность просмотра статистики использования сайта, например, сколько компаний просматривают анкеты, какие анкеты популярны, и т.д;
* Техническая поддержка: информационная связь с системой технической поддержки для пользователей, если у них возникают вопросы или проблемы с использованием сайта.

Обзор существующих аналогичных программных продуктов:

* HeadHunter, SuperJob: обширные платформы с разнообразными вакансиями, включая те, которые требуют навыков 1С программирования;
* LinkedIn: профессиональная социальная сеть, предоставляющая возможность создавать профили и искать вакансии для 1С программистов;
* GitHub Jobs: платформа, где компании могут публиковать вакансии, связанные с программированием, включая 1С;
* Stack Overflow Jobs: специализированная платформа для разработчиков, где можно найти вакансии, требующие навыков 1С программирования;
* Jobvite, Greenhouse: системы управления трудоустройством, предоставляющие инструменты для поиска и привлечения технических специалистов, включая 1С программистов.

**1.1.2 Функциональные требования**

Описание перечня функций и задач, которые должен выполнять будущий ПП:

* регистрация и аутентификация;
* создание резюме;
* взаимодействие с администратором через форму обратной связи;
* модерация и управление содержимым;
* вход в личный кабинет.

**1.1.3 Описание входной, выходной и условно-постоянной информации**

Вся информация, которой оперирует пользователь в процессе решения задачи подразделяется на:

– входную информацию;

– выходную информацию;

– условно-постоянную информацию.

Входной информацией выступают: данные сайта, данные пользователя.

Выходной информацией выступают: измененные данные, сообщения.

Условно-постоянной информацией выступают: данные сайта, данные пользователя.

**1.1.4 Нефункциональные требования**

Требования к применению:

* надежность и стабильность работы сайта, чтобы пользователь мог получить необходимую информацию в любой момент времени;
* удобство использования, чтобы пользователь мог быстро и легко найти нужную информацию и выполнить необходимые действия;
* актуальность информации, предоставляемой сайтом, чтобы пользователь мог быть уверен в том, что получает актуальную и достоверную информацию.

Требования к производительности:

* масштабируемость;
* оптимизация интерфейса.

Требования к реализации:

* функциональность;
* удобство использования;
* скорость и надежность.

Требования к надежности:

* стабильность системы;
* тестирование и верификация.

Требования к интерфейсу:

* интуитивная навигация;
* адаптивность и отзывчивость;
* понятные форматы ввода и вывода;
* согласованный дизайн;
* контрастность и читаемость.

**1.2 Диаграмма вариантов использования**

Проектирование в разработке – это процесс создания плана или схемы для реализации программного обеспечения или системы. Цель проектирования - обеспечить соответствие системы требованиям заказчика и пользователей, а также упростить её разработку, тестирование и поддержку.

Диаграмма вариантов использования – это один из видов диаграмм UML, которая показывает, какие функции предоставляет система и как она взаимодействует с внешними сущностями, называемыми актерами.

Диаграмма вариантов использования состоит из следующих элементов:

* вариант использования – это овал с названием, который описывает конкретную функцию или сервис, который система предоставляет актеру;
* актер – это человек, организация или другая система, которая использует или влияет на систему;
* связь – это отношения, которые определяют, как элементы диаграммы взаимодействуют друг с другом и с системой;
* система – это то, что моделируется диаграммой вариантов использования.

Ниже на рисунке 1 представлена диаграмма вариантов использования для web-ресурса.

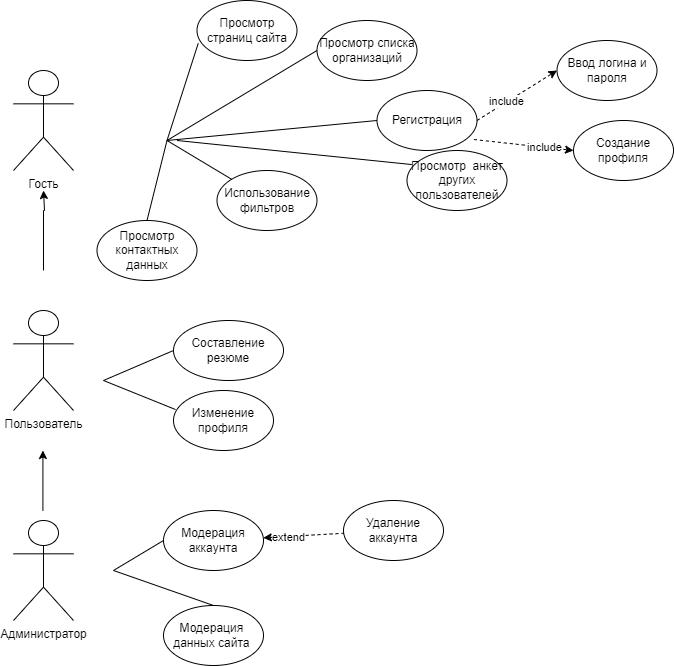


Рисунок 1 – Диаграмма вариантов использования

**1.3 Разработка плана работы над проектом**

Диаграмма Ганта — это популярный тип столбчатых диаграмм, который используется для иллюстрации плана, графика работ по какому-либо проекту. Является одним из методов планирования проектов. Используется в приложениях по управлению проектами.

Диаграмма Ганта по разработке данного программного обеспечения представлена на рисунке 2.

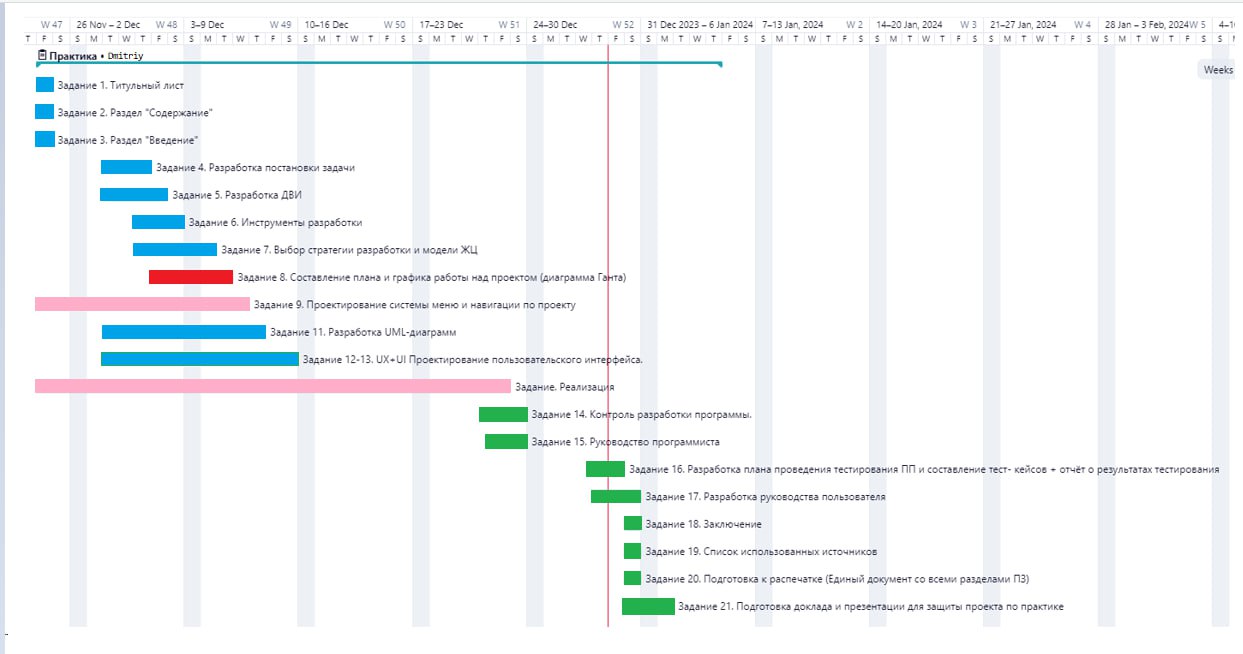


Рисунок 2 – Диаграмма Ганта

**1.4 Выбор стратегии разработки и модели жизненного цикла**

Таблица 1 – Выбор модели жизненного цикла на основе характеристик требований

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории требований | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| Являются ли требования к проекту легко определимыми и реализуемыми? | Да | Да | Да | Нет | Нет | Нет |

Продолжение таблицы 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Могут ли требования быть сформулированы в начале ЖЦ? | Да | Да | Да | Да | Нет | Нет |
| Часто ли будут изменяться требования на протяжении ЖЦ? | Нет | Нет | Нет | Нет | Да | Да |
| Нужно ли демонстрировать требования с целью их определения? | Нет | Нет | Да | Нет | Да | Да |
| Требуется ли проверка концепции программного средства или системы? | Нет | Нет | Да | Нет | Да | Да |
| Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| Нужно ли реализовать основные требования на ранних этапах разработки? | Нет | Нет | Да | Да | Да | Да |

Вычисления: 4 за каскадную, 4 за V- образную, 5 за RAD, 4 за инкрементную, 3 за быстрого прототипирования и 3 за эволюционную.

Итог: на основе результатов заполнения таблицы 1 подходящей является RAD модель.

Таблица 2 **–** Выбор модели жизненного цикла на основе характеристик команды разработчиков

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории команды разработчиков  проекта | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |

Продолжение таблицы 2

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Являются ли проблемы предметной области проекта новыми для большинства разработчиков? | Нет | Нет | Нет | Нет | Да | Да |
| Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков? | Да | Да | Нет | Нет | Нет | Да |
| Изменяются ли роли участников проекта на протяжении ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| Является ли структура процесса разработки более значимой для разработчиков, чем гибкость? | Да | Да | Да | Да | Нет | Нет |
| Важна ли легкость распределения человеческих ресурсов проекта? | Да | Да | Да | Да | Нет | Нет |
| Приемлет ли команда разработчиков оценки, проверки, стадии разработки? | Да | Да | Да | Да | Да | Да |

Вычисления: 5 за каскадную, 5 за V-образную, 6 за RAD, 5 за инкрементную, 2 за быстрого прототипирования и 1 за эволюционную.

Итог: Итог: на основе результатов заполнения таблицы 2 подходящей является RAD модель.

Таблица 3 – Выбор модели жизненного цикла на основе характеристик коллектива пользователей

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории коллектива пользователей | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| Будет ли присутствие пользователей ограничено в ЖЦ разработки? | Да | Да | Нет | Да | Нет | Да |

Продолжение таблицы 3

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки? | Нет | Нет | Нет | Да | Да | Да |
| Будут ли пользователи вовлечены во все фазы ЖЦ разработки? | Нет | Нет | Да | Нет | Да | Нет |
| Будет ли заказчик отслеживать ход выполнения проекта? | Нет | Нет | Нет | Нет | Да | Да |

Вычисления: 0 за каскадную, 0 за V-образную, 2 за RAD, 1 за инкрементную, 3 за быстрого прототипирования и 3 за эволюционную.

Итог: на основе результатов заполнения таблицы 3 подходящей является модель Быстрого прототипирования и Эволюционная

Таблица 4 – Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Критерии категории типов проекта и рисков | Каскадная | V-образная | RAD | Инкрементная | Быстрого прототипирования | Эволюционная |
| Разрабатывается ли в проекте продукт нового для организации направления? | Нет | Нет | Нет | Да | Да | Да |
| Будет ли проект являться расширением существующей системы? | Да | Да | Да | Да | Нет | Нет |
| Будет ли проект крупно- или среднемасштабным? | Нет | Нет | Нет | Да | Да | Да |
| Ожидается ли длительная эксплуатация продукта? | Да | Да | Нет | Да | Нет | Да |

Продолжение таблицы 4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Необходим ли высокий уровень надежности продукта проекта? | Нет | Да | Нет | Да | Нет | Да |
| Предполагается ли эволюция продукта проекта в течение ЖЦ? | Нет | Нет | Нет | Да | Да | Да |
| Велика ли вероятность изменения системы (продукта) на этапе сопровождения? | Нет | Нет | Нет | Да | Да | Да |
| Является ли график сжатым? | Нет | Нет | Да | Да | Да | Да |
| Предполагается ли повторное использование компонентов? | Нет | Нет | Да | Да | Да | Нет |
| Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)? | Нет | Нет | Нет | Нет | Да | Да |

Вычисления: 3 за каскадную, 4 за V – образную, 3 за RAD, 8 за инкрементная, 6 за модель быстрого прототипирования, 7 за эволюционную.

Итог: на основе результатов заполнения таблицы 4 подходящей является модель V – образная.

Общий итог: в итоге заполнения таблиц наиболее подходящей является RAD модель.

**1.5 Инструменты разработки**

Для разработки UML-диаграмм использовалось приложение «Draw.io», так как данное приложение имеет удобный и понятный интерфейс для составления диаграмм.

Для UX/UI дизайна использовалась приложение «Figma», так как в данном приложении удобно работать с компонентами – заранее подготовленными элементами дизайна (кнопки, формы и т. д.), изменения которых распространяются на все его копии.

Для составления документации использовался «Microsoft Word», так как эта программа наиболее подходит для выполнения данной задачи.

Для разработки данного проекта будет выбран конструктор сайтов Tilda. Тильда (Tilda) – конструктор для быстрого создания лендингов, многостраничных сайтов и даже интернет-магазинов.

В конструкторе Tilda есть огромная коллекция блоков (480+): текстовые блоки, преимущества, этапы, команда, изображения/галерея, тарифы, отзывы, разделители, якорные ссылки, меню и тд. Каждый раздел содержит большое количество вариаций дизайна этих блоков.

Для удобного оформления одностраничных сайтов в конструкторе сделано много: блочная структура и множество готовых текстовых, мультимедийных секций, возможности по настройке стилей текста и форматированию блоков, Landing/лонгриды здесь можно презентовать красиво.

Также в Tilda есть такая функция как «Zero Block». Zero Block в Tilda – редактор для опытных дизайнеров для ручного создания блоков из различных элементов c нуля. В данном блоке можно контролировать каждую деталь блока (текст, изображения, кнопки, плашки): ее расположение, размер и на экранах с каким разрешением она будет находиться. Используя слои прозрачности и сложную типографику, можно создать свой блок с индивидуальным дизайном.

Преимущества Тильды:

– лёгкий и доступный интерфейс;

– удобные дизайнерские и типографические настройки;

– интеграция с другими сервисами;

– Zero Block – встроенный конструктор секций;

– своя система аналитики и CRM;

– подробный раздел FAQ;

Разработка проекта будет происходить на ноутбуке со следующими параметрами:

* процессор Intel Core i5-11400H 2.7 GHz;
* объем оперативной памяти 16.00 GB;
* объем места на жестких дисках 512 GB;
* видеокарта NVIDIA RTX 3050 Laptop 4GB;
* ОС Windows 11 Home.

**2. Проектирование задачи**

**2.1 Разработка структуры сайта, системы меню, навигации**

Для успеха сайта очень важно его проектирование, так как оно влияет на то, как пользователи взаимодействуют с сайтом и насколько он эффективен. Качественный сайт должен быть простым в использовании, иметь удобную навигацию, быстро загружаться и удовлетворять запросы пользователей (рисунок 3 и рисунок 4).



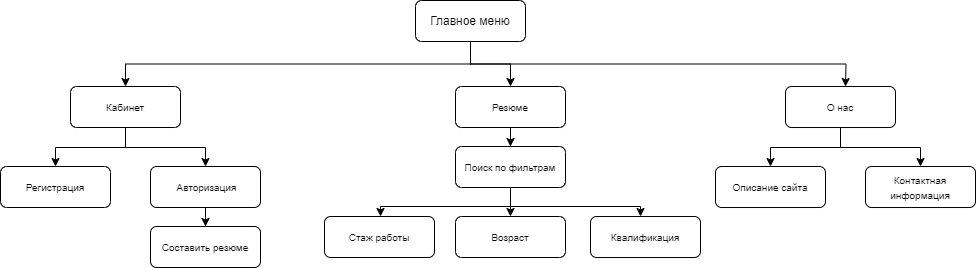
Рисунок 3 – Структура сайта

Рисунок 4 – Главное меню сайта

**2.2 Разработка UML-диаграмм**

**2.2.1 Диаграмма деятельности**

Диаграмма деятельности – это UML-диаграмма, на которой показаны действия, состояния которых описано на диаграмме состояний.

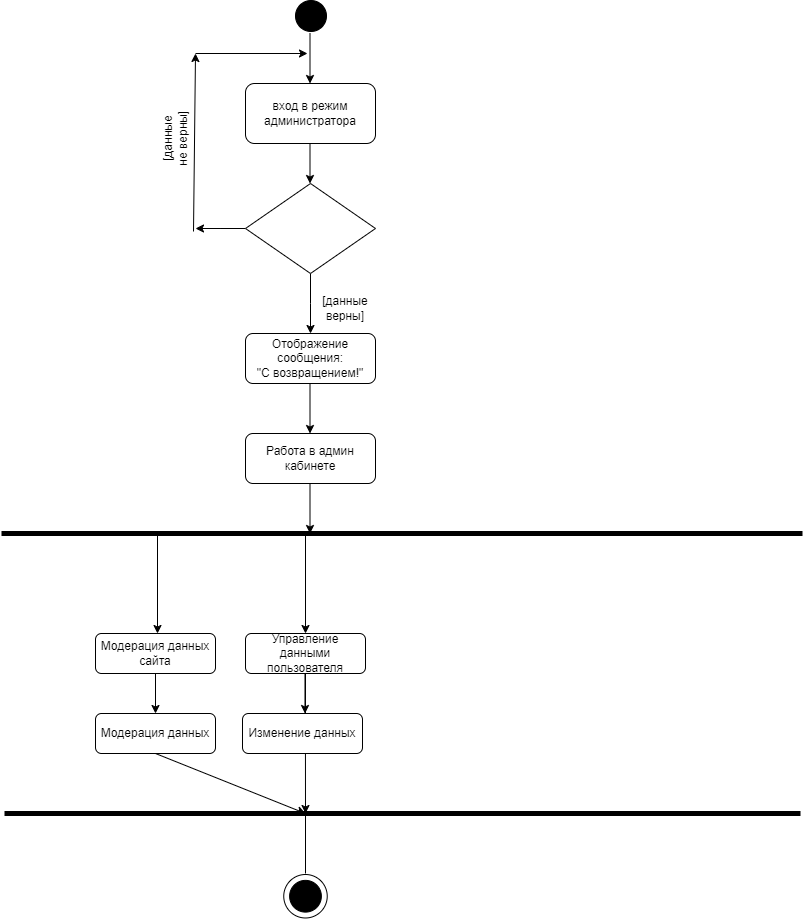


Рисунок 5 – Диаграмма деятельности

**2.2.2 Диаграмма последовательности**

Диаграмма последовательности — UML-диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл объекта и взаимодействие актеров информационной системы в рамках прецедента.

**Диаграммы последовательностей используются для уточнения диаграмм прецедентов**, более детального описания логики сценариев использования. Это отличное средство документирования проекта с точки зрения сценариев использования.

Диаграммы последовательностей обычно содержат **объекты**, которые **взаимодействуют в рамках сценария**, **сообщения**, которыми они обмениваются, и **возвращаемые результаты**, связанные с сообщениями.

**Объекты** обозначаются прямоугольниками с подчеркнутыми именами (чтобы отличить их от классов). **Сообщения (вызовы методов)** - линиями со стрелками. **Возвращаемые результаты** - пунктирными линиями со стрелками.

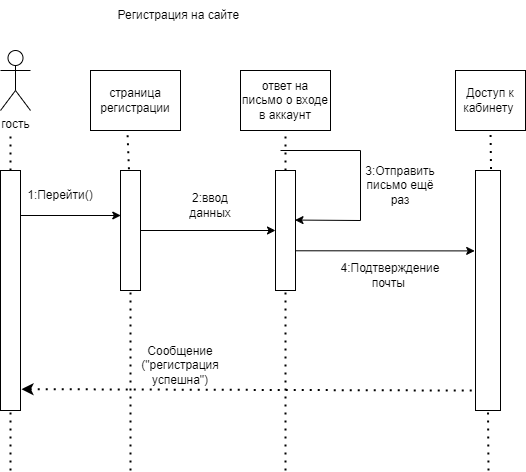
Прямоугольники на вертикальных линиях под каждым из объектов показывают **«время жизни» (фокус) объектов**. Впрочем, довольно часто их не изображают на диаграмме, все это зависит от индивидуального стиля проектирования.

Рисунок 6 – Диаграмма последовательности

**2.3 Разработка пользовательского интерфейса**

Важным элементом проектирования данного программного продукта является описание внешнего интерфейса разрабатываемого интернет-ресурса.

Для разработки визуального дизайна использовались сдержанные, мягкие цвета для удобства использования программного продукта.

В ходе разработки был спроектирован удобный и простой дизайн.

Для организации эффективной работы пользователя нужно создать целостный программный продукт данной предметной области, в котором все компоненты будут сгруппированы по функциональному назначению. При этом необходимо обеспечить удобный графический интерфейс пользователя. Интернет-ресурс должен позволить пользователю решать задачи, затрачивая значительно меньше усилий, чем при работе с разрозненными объектами. Все исходные данные будут разделены на несколько групп.

Прототип – это наглядная модель пользовательского интерфейса. В сущности, это «черновик», созданный на основе представления разработчика о потребностях пользователя. Итоговое отображение программы может отличаться от прототипа.

**3. Реализация**

**3.1 Руководство программиста**

Данный программный продукт был разработан с помощью онлайн-конструктора сайтов Tilda. Данный конструктор предлагает большой выбор способов создания сайта, например, возможность создать сайт в редакторе с помощью шаблонов, создать сайт «с нуля», выбирая стилистику и виджеты, которые будут расположены на сайте. Большинство блоков были созданы с нуля, для этого использовался «Zero Block» (рисунок 7).

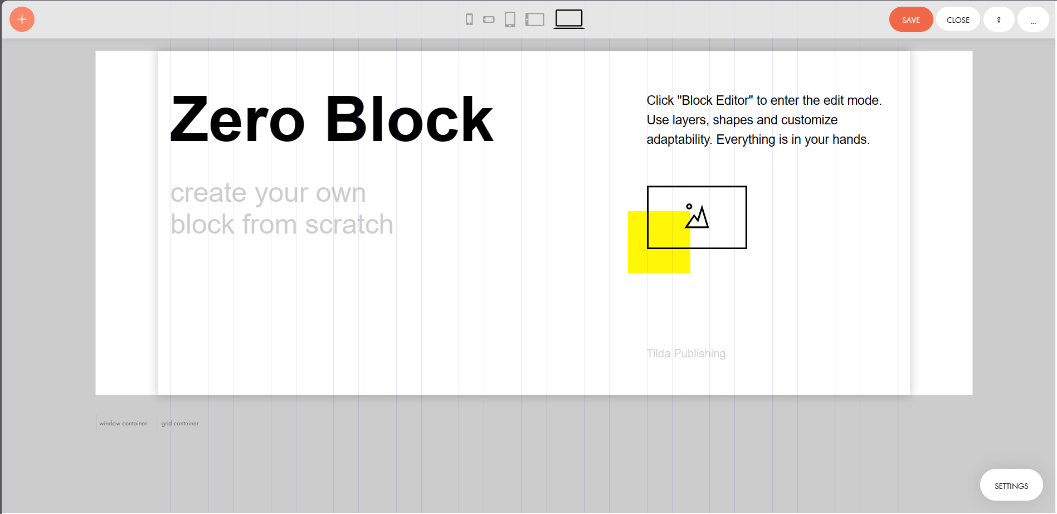
****

Рисунок 7 – Zero-блок

**3.1.1 Создание сайта**

Для того, чтобы создать свой web-ресурс в Tilda, необходимо сначала зарегистрироваться. Данные для входа на сайт администратора:

– Email: dimam3338@gmail.com

– Пароль: moto1234

После авторизации нужно нажать кнопку «Создать сайт». Tilda предложит шаблоны страниц, для сайта «1CJob» шаблон не брали: создавали пустую страницу. Появится возможность выбрать блоки для страницы. Для создания данного сайта первым был выбран Zero-блок.

**3.1.2 Настройка функциональности и дизайна**

Все элементы, которые имеются на экране, могут быть изменены, перемещены или удалены. Когда курсор наводится на блоки, расположенные на страницах web-ресурса, появляется возможность их редактирования. Практически все элементы могут быть переписаны, изменены по шрифту теста, размеру, форматированию, цвету и другим параметрам.

Tilda дает множество блоков для создания сайта, которые могут быть отредактированы, дополнены новыми элементами и удалены, изменены по фону, полям ввода и т.д. В Zero-блоке больше всего настроек – это чистое поле для создания своего блока.

Меню состоит из 8-ми пунктов:

– «Мои сайты» – ранее созданные сайты;

– далее идёт два пункта меню с названием в сайта, который редактируется в данный момент; там можно просмотреть страницы сайта и зайти в настройки, опубликовать страницы;

– «Предпросмотр» – позволяет посмотреть страницу перед публикацией;

– «Опубликовать»– публикует страницу и в появившемся окне даёт ссылку на неё;

–«Настройки» – настройки страницы, где можно изменить название, удалить, экспортировать, добавить описание или изменить адрес;

– «Помощь» – справочный центр Tilda;

– «Ещё» – содержит 9 пунктов («Настройка сайта», «Настройка страницы», «Шрифты», «Мои сайты», «Профиль», «Тарифы и оплата», «Помощь», «Блоки на странице», «Выйти»);

**3.1.3 Редактирование страниц**

При наведении мыши на блок, появляется «Меню редактирования»:

– «Настройки» – позволяют настроить отступы, цвет фона блока, диапазон видимости на устройствах, посмотреть id блока и создать класс( в других блоках эти настройки могут отличаться, в зависимости от его назначения);

– «Редактировать блок» (в других блоках вместо этого пункта «Контент») – позволяет добавить элементы на блок и редактировать их;

– «Дублировать» – дублирует блок;

–«Удалить» – удаляет блок;

– «Спрятать/показать» – скрывает блок с сайта;

– «Вырезать, копировать, вставить, выделить блок»;

– «Переместить вверх» – перемещает блок выше предыдущего;

– «Переместить вниз» – перемещает блок ниже следующего.

Рисунок 8 – Меню редактирования Zero-блока



Рисунок 9 – Меню редактирования всех блоков

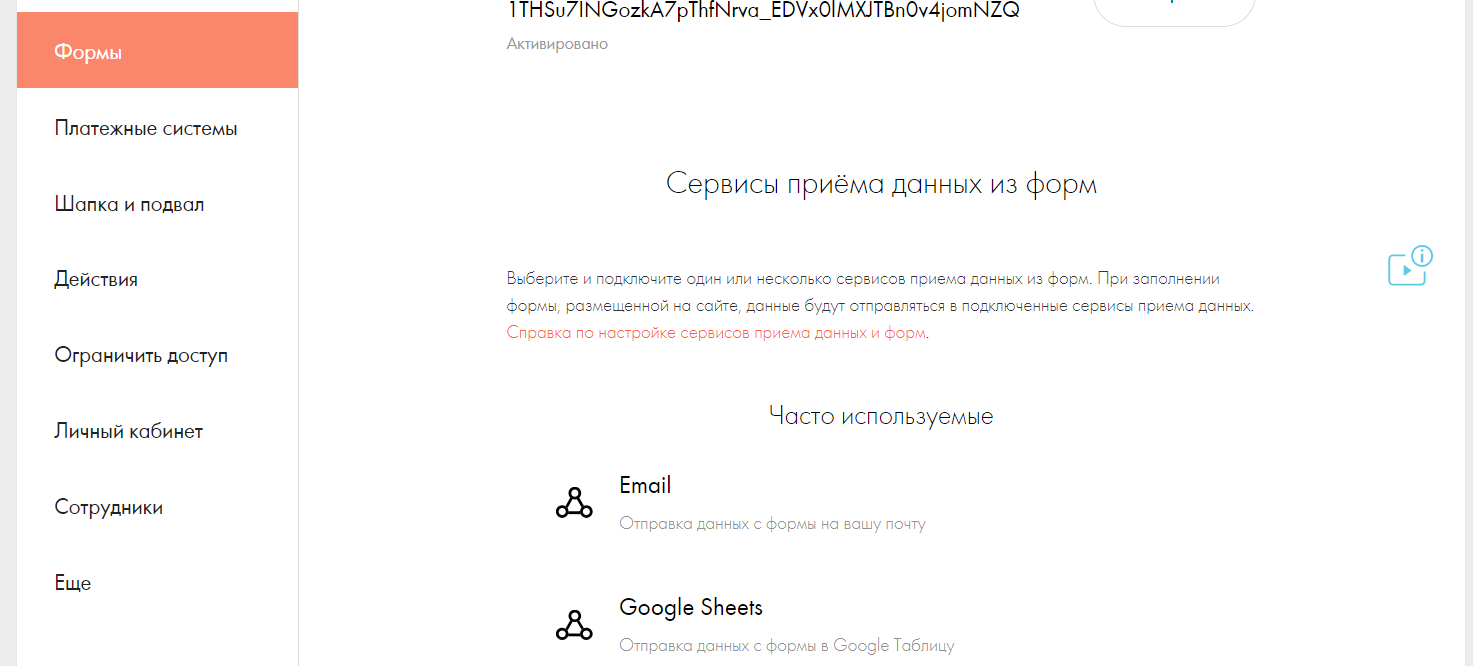
В настройках сайта, в вкладке «Формы» можно добавить отправку данных в google-таблицу (рисунок 10)  


Рисунок 10 – Отправка данных в гугл-таблицу

**4 Тестирование**

**4.1 Тесты на использование**

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения испытания реализации программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

Отчёт о результатах тестирования предоставлен в таблице 6.

Таблица 5 – Отчёт результатах тестирования

|  |  |  |  |
| --- | --- | --- | --- |
| Тест | Ожидаемый результат | Физический результат | Результат  тестирования |
| Проверка поля ввода «Имя» | Корректный ввод | Ввод значений прошёл успешно | Выполнено |
| Проверка поля ввода «Пароль» | Корректный ввод и введённые значения должны быть невидны | Ввод значений прошёл успешно и значений не видно | Выполнено |
| Проверка кнопки «О нас» | Переход на вкладку «О нас» | Переход прошел успешно | Выполнено |
| Проверка кнопки «Резюме» | Переход на вкладку «Резюме» | Переход прошел успешно | Выполнено |
| Проверка поля ввода «Email» | Корректный ввод и при вводе требуется ввести обязательный символ «@» | Ввод удался и если ты не ввёл символ «@» требует его ввести | Выполнено |
| Кнопка «Кабинет» | Переход на вкладку «Кабинет» | Переход прошел успешно | Выполнено |

Продолжение таблицы 5

|  |  |  |  |
| --- | --- | --- | --- |
| Логотип | Переход на вкладку «Главная» | Переход прошел успешно | Выполнено |
| Наши контакты | Сохранение данных в гугл-таблицу | Данные успешно сохранены | Выполнено |
| Наши контакты | Ввод данных | Данные успешно записаны | Выполнено |

**4.2 Отчёт о результатах тестирования**

При разработке программного продукта было решено множество проблем, например, не осуществлялся переход на новую вкладку по нажатию кнопки или открывался доступ к последнему вопросу в тесте, поэтому в будущем пользователь не столкнется с данными проблемами.

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

**5. Руководство пользователя**

Цель данного проекта заключается в создании сайта для нахождения работы 1С программистам с интуитивно понятным интерфейсом, его описанием и структурированностью.

Создаваемый сайт будет рассчитан на любого рода пользователей, которые хотят найти работу.

Для того, чтобы открыть сайт необходимо перейти по ссылке «http://1cjob.by.tilda.ws». Затем достаточно следовать приведенной инструкции ниже:

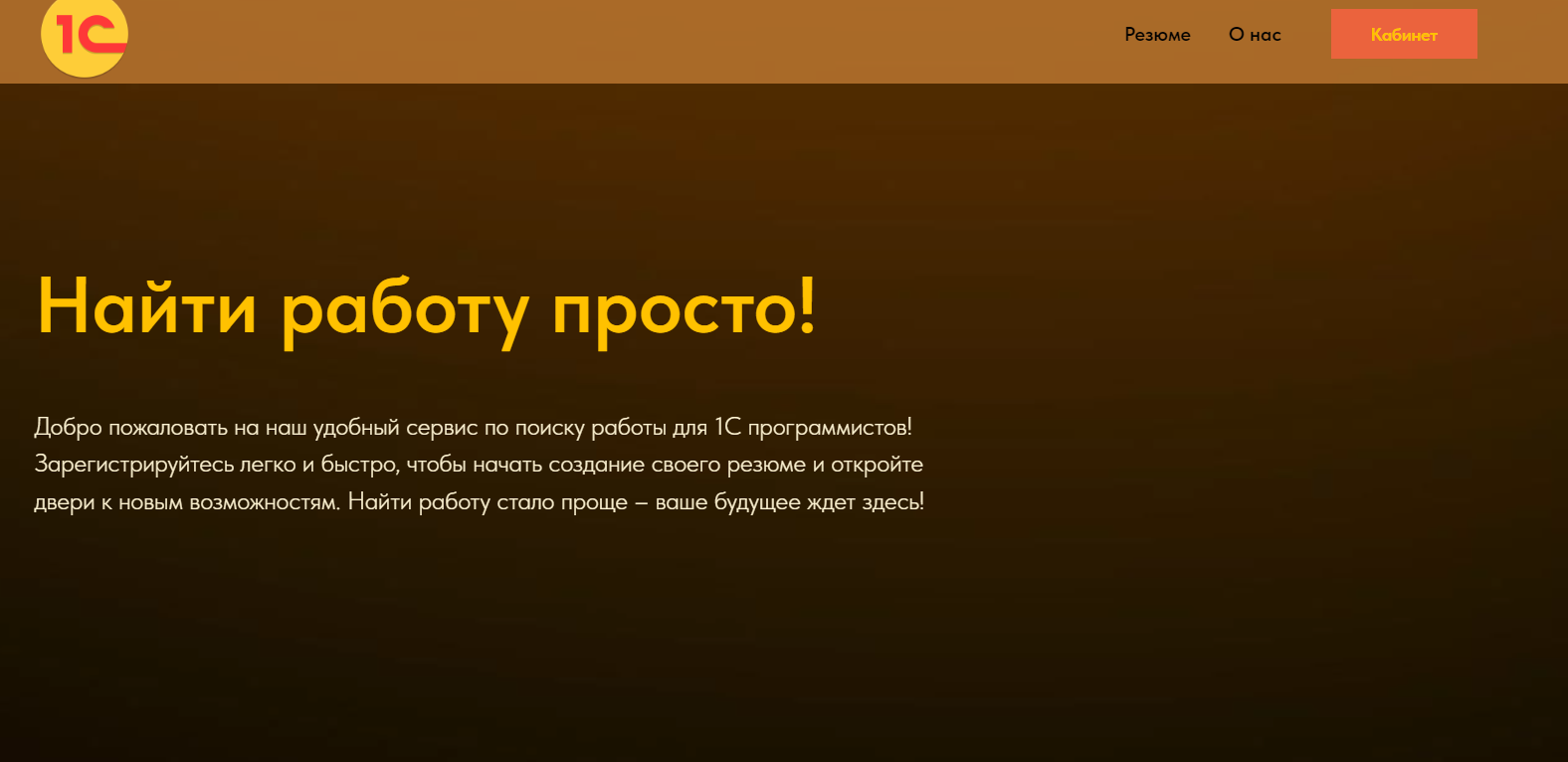
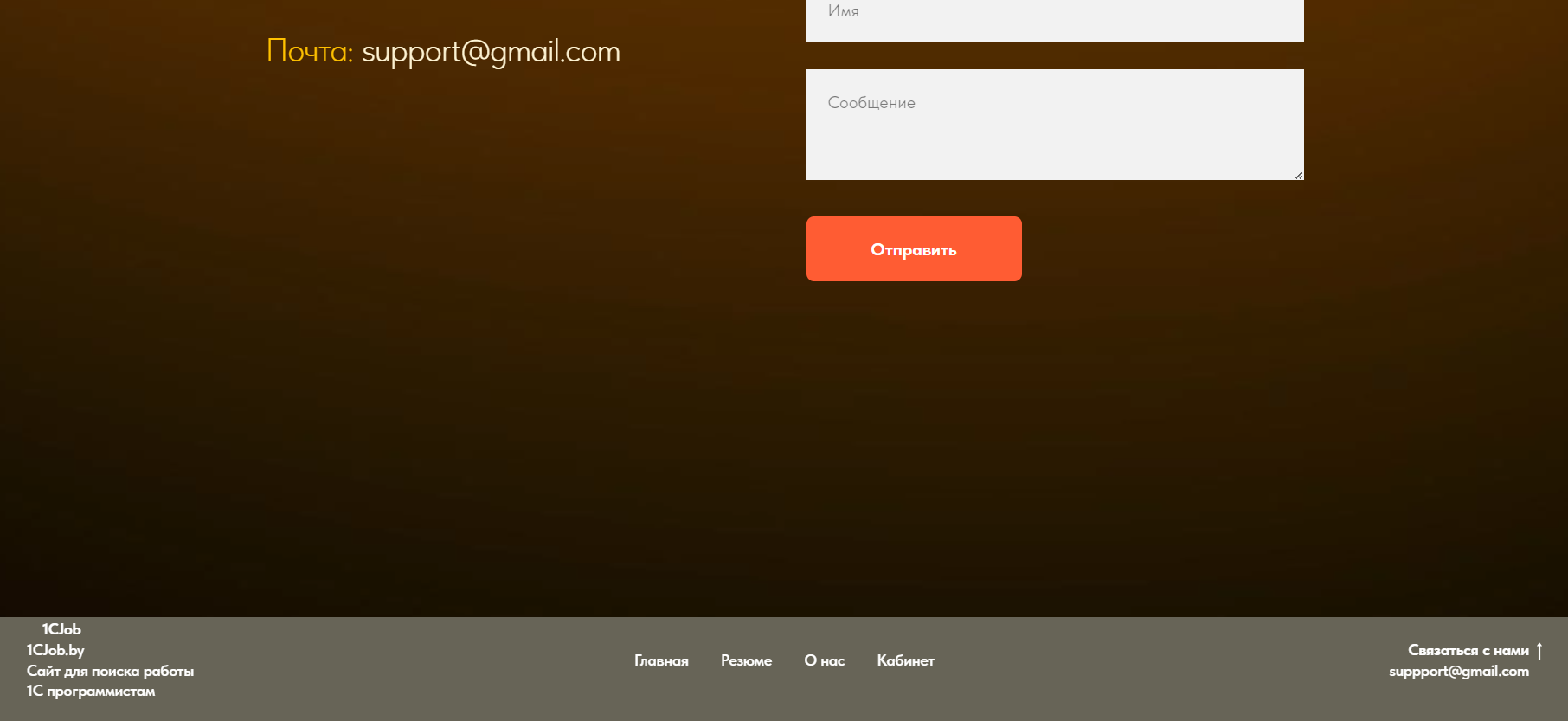
После открытия сайта нас встречает главная страница (рисунок 11).

Рисунок 11 – Главная страница

Страницу можно пролистывать вниз, читая информацию до подвала (рисунок 12).

Рисунок 12 – Подвал

При нажатии на Хэдере или подвале на кнопку «Кабинет» нас перенесет на соответсвующую страницу (рисунок 13).



Рисунок 13 – Главная форма сайта

При нажатии на кнопку «О нас» нас перенесет на соответсвующую страницу (рисунок 14).



Рисунок 14 – Страница «О нас»

При нажатии на кнопку «Резюме» нас перенесет на соответсвующую страницу (рисунок 15).



Рисунок 15 – Страница «Резюме»

При нажатии на логотип нас перенесет на главную страницу (рисунок 16).

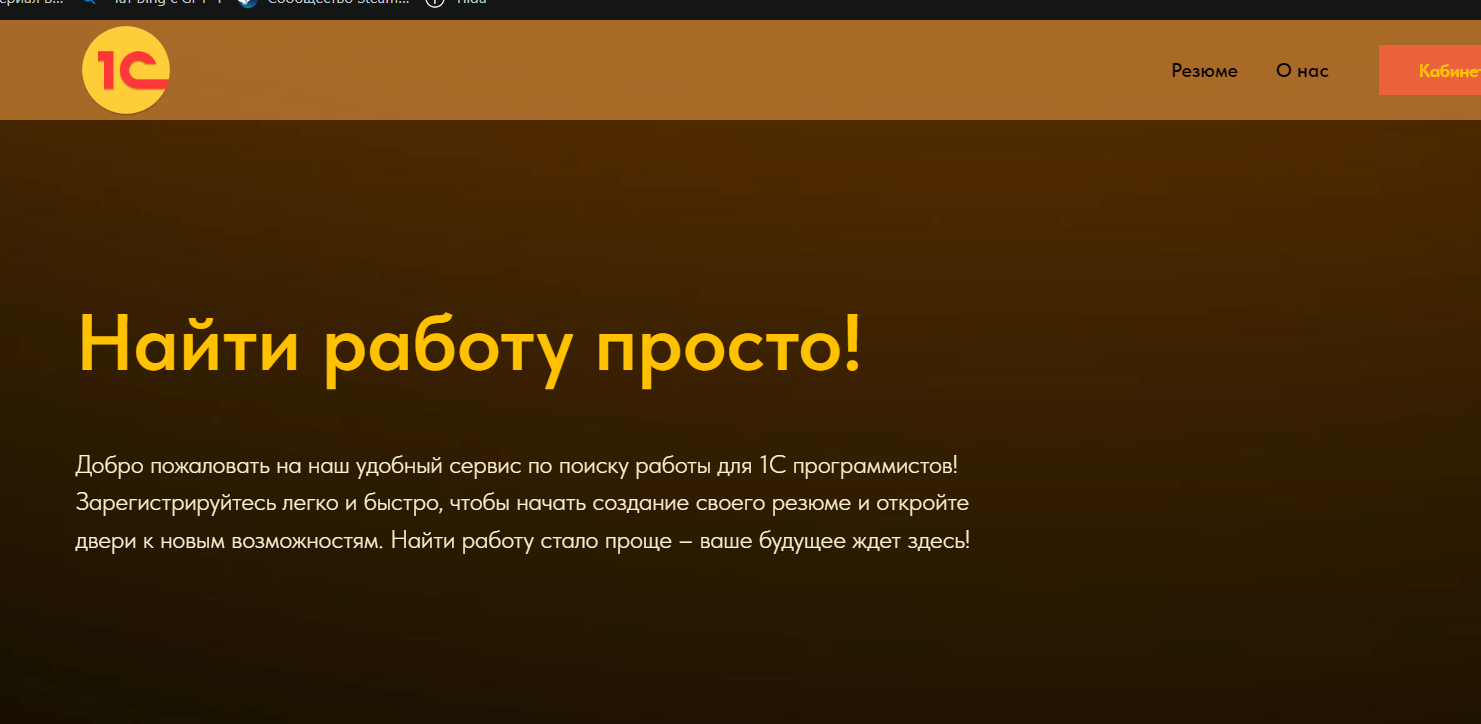


Рисунок 1 – Главная страница

**Заключение**

В заключении к проекту по теме “web-ресурс для трудоустройства 1С программистов”, реализованному на конструкторе Tilda, можно отметить следующее:

В ходе работы над проектом были успешно применены знания и навыки в области веб-разработки. Проект представляет собой web-ресурс, который был полностью разработан с использованием конструктора Tilda

HTML использовался для создания структуры медиаплеера, CSS - для стилизации и создания пользовательского интерфейса, а JavaScript - для обеспечения функциональности медиаплеера.

В результате был создан функциональный медиаплеер, который может воспроизводить, приостанавливать и управлять медиафайлами. Это доказывает эффективность использования HTML, CSS и JavaScript в веб-разработке.

В целом, проект был успешным, и он показал, как можно использовать базовые технологии веб-разработки для создания сложных и функциональных приложений. Этот проект служит отличным примером того, как можно применять и развивать полученные навыки на практике.

Исходя из этого, можно сделать вывод, что программа реализована успешно.

**Список использованных источников**

1. Изучаем Tilda https://web.universuspro.ru – 20 мин.

2. Как создать сайт на Тильде. Начало работы /Tilda Publishing YT. – 90 мин.

3. Создание веб-сайта для чайников 2-е изд. / Дэвид Кроудер – США, 2003. – 168 с.