



МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ  
УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

---

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И  
ТЕХНОЛОГИЧЕСКОГО ОБРАЗОВАНИЯ**  
Кафедра информационных технологий и электронного обучения

Основная профессиональная образовательная программа  
Направление подготовки 09.03.01 Информатика и вычислительная техника  
Направленность (профиль) «Технологии разработки программного обеспечения»  
форма обучения – очная

### **Курсовая работа**

«Организация и публикация электронного портфолио по модулю  
«Информационные технологии в управлении в IT-компаниях» с использованием  
генераторов статических сайтов»

Обучающейся 4 курса  
Леонтьевой Анны Викторовны

Научный руководитель:  
Кандидат физико-математических наук,  
доцент кафедры ИТиТО  
Жуков Николай Николаевич

Санкт-Петербург  
2023

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	3
Описание проекта и выбор методологии управления проектом .....	4
1.1. Предметная область .....	4
1.2. Модели управления разработкой .....	4
Управление проектом и выполненные задачи.....	<b>Ошибка! Закладка не определена.</b>
2.1. Внедрение основных принципов методики Scrum в процесс реализации программного продукта.....	9
2.2. Документальное сопровождение программного проекта .....	13
2.3. Анализ процесса реализации программного проекта по диаграмме сгорания задач.....	16
ЗАКЛЮЧЕНИЕ.....	18
ЛИТЕРАТУРА .....	19

## ВВЕДЕНИЕ

Деятельность ресторана в наши дни предполагает большие объемы отчетности по всем видам работ, происходящих в точках общественного питания. В связи с этим, целесообразна разработка информационной системы, которая позволит оптимизировать и упростить работу ресторана.

Реализация подобных информационных систем происходит в команде разработчиков, исходя из этого, актуальным вопросом является организация управления программным проектом. Для качественной реализации проекта необходим рациональный и корректный выбор методологии управления. Один из вариантов определения таковой – изучение существующих моделей и сопоставление ресурсов команды на начало проекта с соответствующими требованиями каждой модели.

Целью курсовой работы является реализация программного проекта по проектированию деятельности ресторана с применением методики управления Scrum.

Для выполнения поставленной цели необходимо выполнить следующие задачи:

- Исследовать методологии и методики управления;
- Определить рациональную для использования методику управления программным проектом;
- Изучить выбранную методику управления и внедрить основные принципы в работу команду разработчиков;
- Разработать программный проект по проектированию деятельности ресторана;
- Подготовить сопроводительную документацию к программному продукту.

Объектом является процесс разработки программного проекта по проектированию деятельности ресторана.

Предметом является реализация методики управления программным проектом.

## Описание проекта и выбор методологии управления проектом

### 1.1. Предметная область

Модуль «Информационные технологии в управлении в IT-компаниях» включает в себя следующие дисциплины:

- Основы электронного управления
- IT-менеджмент
- Системы бизнес-аналитики

Выполнение учебных заданий по этим дисциплинам предполагает использование онлайн-сервисов и специального программного обеспечения, такие как:

- BSCDesigner
- Bizagi Modeler
- Excel
- Yandex Data Lens
- Power point

После выполнения каждого учебного задания предполагается создание отчета.

### 1.2. Генераторы статических сайтов как инструмент реализации электронного портфолио

Статические и динамические web-сайты – это два основных вида web-сайтов. Статические сайты состоят из фиксированных web-страниц, каждая из которых представляет собой отдельный html-файл. Каждый такой файл хранится на сервере и передаётся пользователю без изменений. Динамические сайты отличаются генерацией контента в реальном времени, исходя из действий пользователя. Сравнением вышеперечисленных видов web-сайтов в контексте использования сайта студентом для электронного портфолио приведено в Таблице 1.

Таблица 1

Статический сайт	Динамический сайт
Некритичен к программному обеспечению сервера	Критичен к программному обеспечению сервера

Возможен минимальный порог вхождения для создания, развёртывания и поддержки работы сайта	По сравнению со статическими сайтами, порог вхождения для создания, развёртывания и поддержки работы сайта
Отсутствие интерактивных форм взаимодействия с пользователем	Возможность интерактивного взаимодействия с пользователем и создания сложных механик взаимодействия
Невозможность унифицировать работу с повторяющимися блоками	Возможность унификации работы с повторяющимися блоками
Высокая скорость загрузки	Возможность долгой загрузки сайта из-за необходимости обязательного запуска кода

Создание электронного учебного портфолио – это обязательная часть учебного процесса, которая начинается с момента поступления студента. Поэтому выбор вида сайта должен предполагать отсутствие у начинающего разработчика глубоких профильных знаний для создания сайтов: в документе «Кодификатор проверяемых требований к результатам освоения основной образовательной программы среднего общего образования и элементов содержания для проведения единого государственного экзамена по информатике» не упоминается необходимых для реализации умений и требований, необходимых для создания динамического сайта<sup>1</sup>.

Отсутствие возможности реализации интерактивного взаимодействия с пользователем не критично в контексте предоставления архивных учебных файлов студента. При этом, постоянное и регулярное обновление информации в ходе обучения на статическом сайте будет происходить быстрее и легче, так как предполагает исключительно редакцию html-файла и загрузку новых файлов с отчетами по дисциплинам. Для формата электронного учебного портфолио студента больше всего подходит формат статического сайта.

---

<sup>1</sup> В пункте 2.13 упомянутого документа описывается необходимость умения работы с реляционными базами данных только для анализа, предоставления и обработки информации. При этом для создания динамического сайта необходимо, например, умение связывать сервер с базой данных, что не упоминается в кодификаторе.

Рассмотрим два варианта создания статических сайтов: создание ручная верстка сайта-портфолио с помощью HTML и создания аналогичного сайта с помощью генератора статических сайтов.

В контексте создания статических сайтов, оба варианта по уровню сложности подходят для студента, который начал обучение в университете по специальности, так как основы вёрстки затрагиваются в школьной программе согласно кодификатору. Рассмотрим принцип работы генераторов статических сайтов.

Генераторы статических сайтов отличаются наличием шаблонов – правил, по которым оформляются все страницы. Загрузка контента представляет из себя получение контента, например, в формате Markdown и автоматическое преобразование шаблонов и нового контента в HTML страницу по образцу. При этом изменение в файле Markdown будут автоматически переносится в итоговое отображение сайта.

Существует огромное количество генераторов статических сайтов (SSG – statis site generator). Для сравнительного анализа и последующей работы были взяты самых популярных генераторов по рейтингам различных интернет-ресурсов (по упоминанию).

Next.js — это готовый к использованию генератор статических сайтов на основе React. Этот генератор использует гибридный рендеринг веб-страниц. В результате статического рендеринга получается набор статических файлов, которые затем могут быть быстро доставлены клиентам. Гибридный подход включает возможность выполнения дополнительного динамического рендеринга на сервере для определенных компонентов или разделов страницы, которые требуют актуализации данных или вывода на основе пользовательских параметров. Это может быть полезно, например, для динамической загрузки контента, формирования списка постов или фильтрации результатов.

Next.js поддерживает TypeScript, что значительно упрощает разработку и поддержку программ на JavaScript, так как помогает выявлять ошибки на этапе разработки и предоставляет более точную информацию о коде. Также генератор отличается поддержкой многоязычных сайтов, Next.js аналитикой и осуществляет автоматические компиляцию и объединение.

Hugo – генератор статических сайтов, написанный на языке Go. Отличается высокой скоростью и гибкостью. Отличается многообразием типов контента, возможностью его вывода в различных форматах, а также динамическим контентом, управляемым API. Hugo имеет дополнительные шаблоны для оптимизации сайта, комментирования и аналитики.

Gatsby – генератор статических сайтов, который обладает самой большой экосистемой плагинов для работы с данными. Написан генератор на React, также как и Next.js, поэтому генератор поддерживает компоненты React.js

Позволяет работать с данными из API, баз данных, CMS, статических файлов.

Gatsby поддерживает отложенную и интеллектуальную предварительную загрузки.

Отложенная загрузка сайтов, или ленивая загрузка, используется для откладывания загрузки некоторых ресурсов на веб-странице до тех пор, пока они не станут необходимы для отображения контента, который видит пользователь. Интеллектуальная предварительная загрузка использует данные о поведении пользователей для предварительной загрузки ресурсов на веб-странице заранее, до того, как пользователь активно взаимодействует с ними. Это позволяет сократить задержку при загрузке ресурсов во время активного взаимодействия пользователя с сайтом.

Jekyll основан на Ruby и представляет собой простой генератор статических сайтов со встроенной поддержкой GitHub Pages. Генератор отображает шаблоны Markdown и Liquid. Поддерживает интерфейсные платформы, такие как Bootstrap и Semantic UI, а также поддерживает Content Management System (CMS). Один из самых простых генераторов, с которым легче всего начать работать, например, напрямую из GitHub Pages.

Nuxt.js – это генератор статических сайтов, который быстро экспортирует веб-приложения Vue.js в простые статические веб-сайты. Одной из особенностей Nuxt.js является возможность создания универсальных приложений. Универсальность (или изоморфность) означает, что приложение может выполнять серверный рендеринг, то есть HTML-страница генерируется на сервере с уже встроенными данными, после чего отправляется клиенту. Это полезно для SEO (от англ. search engine optimization — это процесс оптимизации веб-сайта с целью улучшения его видимости и рейтинга в поисковых системах) и обеспечивает более быстрое отображение первоначального содержимого страницы пользователю. Как часть экосистемы Vue.js, Nuxt позволяет использовать переходы и анимации. Генератор предоставляет модульную архитектуру, которая позволяет разработчикам легко добавлять и интегрировать функции в виде Nuxt модулей.

Далее предлагается рассмотреть каждый представленный генератор статических сайтов в контексте работы неопытного пользователя. Для сравнения были выведены следующие критерии, основанные на базовых необходимых требованиях такого пользователя инструменты:

- Подключение к Git Hub Pages (критерий обусловлен популярностью Git среды среди обучающихся, а также бесплатным хостингом для сайтов Git Hub Pages)
- Наличие актуальной документации и обучающей информации по использованию генератора и ее обновление
- Скорость обновления контента
- Необходимость изучения других дополнительных инструментов разработки (вёрстки). Этот пункт отражает порог вхождения.

Результаты сравнения по критериям приведены в Таблице 2.

Таблица 2

	Hugo	Gatsby	Jekyll	Next.js	Nuxt.js
Подключение к Git Hub Pages	Предлагает интеграцию с GitHub Pages через настройку сборки и развертывания	Предлагает интеграцию с GitHub Pages через настройку сборки и развертывания	GH Pages поддерживает Jekyll по умолчанию, есть возможность облегченной генерации сайта по шаблону генератора.	Предлагает интеграцию с GitHub Pages через настройку сборки и развертывания	Предлагает интеграцию с GitHub Pages через настройку сборки и развертывания
Наличие актуальной документации и обучающей информации по использованию генератора и ее обновление	Обновляющаяся документация, информационная поддержка от разработчиков, наличие курсов по созданию и использованию сайтов.	Обновляющаяся документация, информационная поддержка от разработчиков, наличие курсов по созданию и использованию сайтов.	Обновляющаяся документация, информационная поддержка от Jekyll и GH Pages, наличие курсов по созданию и использованию сайтов.	Обновляющаяся документация, информационная поддержка от разработчиков, наличие курсов по созданию и использованию сайтов.	Обновляющаяся документация, информационная поддержка от разработчиков.
Скорость обновления контента (указывается место в рейтинге по скорости)	1	5	2	4	3
Необходимость изучения других дополнительных инструментов разработки (вёрстки)	Markdown, основы работы с git и командной строкой. Могут помочь основы веб-вёрстки.	Базовые знания по JavaScript, React и GraphQL. Markdown, основы работы с git и командной строкой. Могут помочь основы веб-вёрстки.	Markdown, основы работы с git и командной строкой. Могут помочь основы веб-вёрстки.	Базовые знания по JavaScript, React. Markdown, основы работы с git и командной строкой. Могут помочь основы веб-вёрстки.	Базовые знания по JavaScript, React. Markdown, основы работы с git и командной строкой. Могут помочь основы веб-вёрстки.

Данные о скорости работы были взяты из исследования «**Static Site Generators Build Performance Testing**».

Подводя итоги сравнительного анализа по выведенным критериям, для поставленной задачи – создание электронного портфолио неопытным в веб-разработке пользователем – оптимальным решением будет использование генератора статических сайтов Jekyll.



## Создание электронного портфолио

### 2.1. Создание электронного портфолио с помощью Jekyll

Создание электронного портфолио студента, как и любого другого статического сайта, начинается с выбора хостинга. Выбор GitHub Pages в контексте цели данной работы обусловлен несколькими причинами:

- Этот сервис абсолютно бесплатный;
- Git Hub Pages позволяет максимально быстро развернуть подготовленный сайт;
- Навыки работы с GitHub – обязательная часть обучения студента по профилю информационных технологий;
- Исходя из предыдущего пункта, большинство учебных репозиторий также будут располагаться на GitHub в процессе обучения;
- Возможность работать с контентом на сервисе GitHub.

Создание сайта-портфолио необходимо начать с регистрации на GitHub. После регистрации был создан новый репозиторий (Рисунок 4).

The screenshot shows the 'Create a new repository' page on GitHub. It includes fields for 'Owner' (leontevanna) and 'Repository name' (my\_portfolio), a 'Description' field, and options for 'Public' or 'Private' visibility. The 'Initialize this repository with' section has 'Add a README file' checked. At the bottom, there is a 'Create repository' button. Red boxes and numbers 1 through 4 highlight specific parts of the form: 1 points to the repository name field, 2 points to the visibility selection, 3 points to the README initialization checkbox, and 4 points to the 'Create repository' button.

Рисунок 1

В блоке 1 было указано название репозитория. Название будет задействовано в адресе сайта, поэтому существует необходимость указать название, связанное с тем контентом, который будет представлен на сайте.

Формат портфолио устанавливает необходимость к доступу к сайту из внешней среды.

Исходя из этого в блоке 2 было установлено значение «Public».

В данной работе использование генератора статических сайтов обусловлено в том числе легкостью добавления контента. Добавление любой информации на сайт происходит с помощью языка разметки Markdown. В блоке 3 была отмечена необходимость автоматически создать файл README.md, через который и будет добавляться информация на сайт.

Последним шагом на данном этапе является подтверждение создания репозитория.

После создания репозитория была открыта вкладка «Settings» / «Pages» (Рисунок 3).

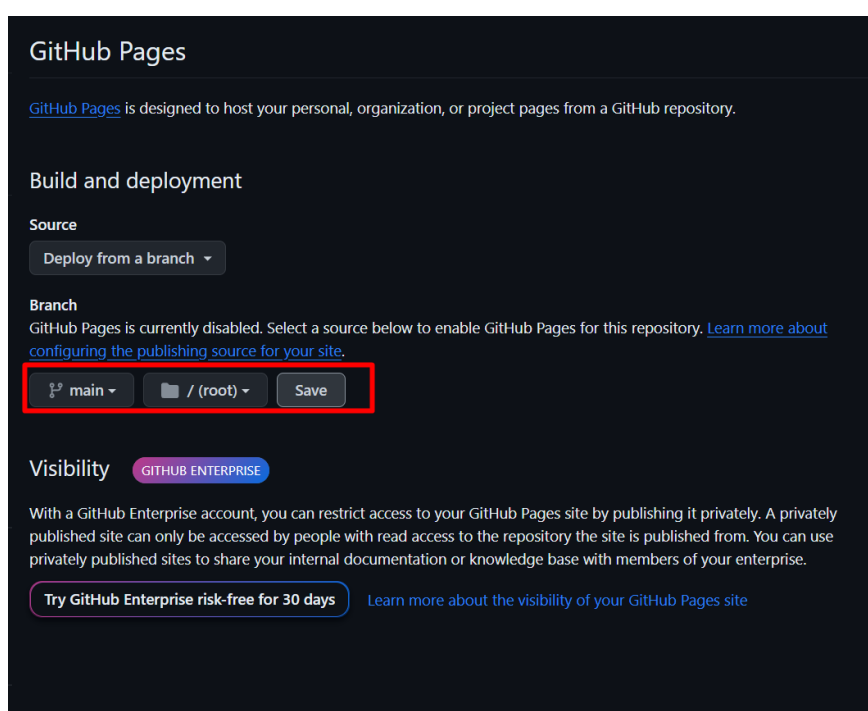


Рисунок 2

Для того, чтобы включить GitHub Pages, был выбран источник (ветка main, корневая папка). После этого изменения были сохранены. После подтверждения сохранения система начинает развертывание системы. Через несколько минут стала доступна ссылка на сайт формата «[https://\[имя\\_пользователя\].github.io/\[название\\_репозитория\]/](https://[имя_пользователя].github.io/[название_репозитория]/)».

При переходе по ссылке открылся созданный сайт (Рисунок 3). Контент, который был на сайте, был в файле README.md.

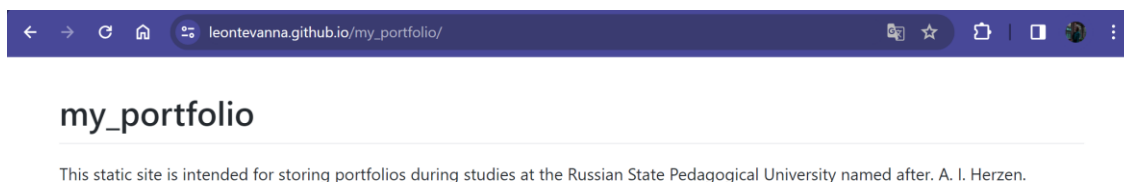


Рисунок 3

После развёртывания сайта открывается доступ к дополнительным настройкам страницы (рисунок 4).

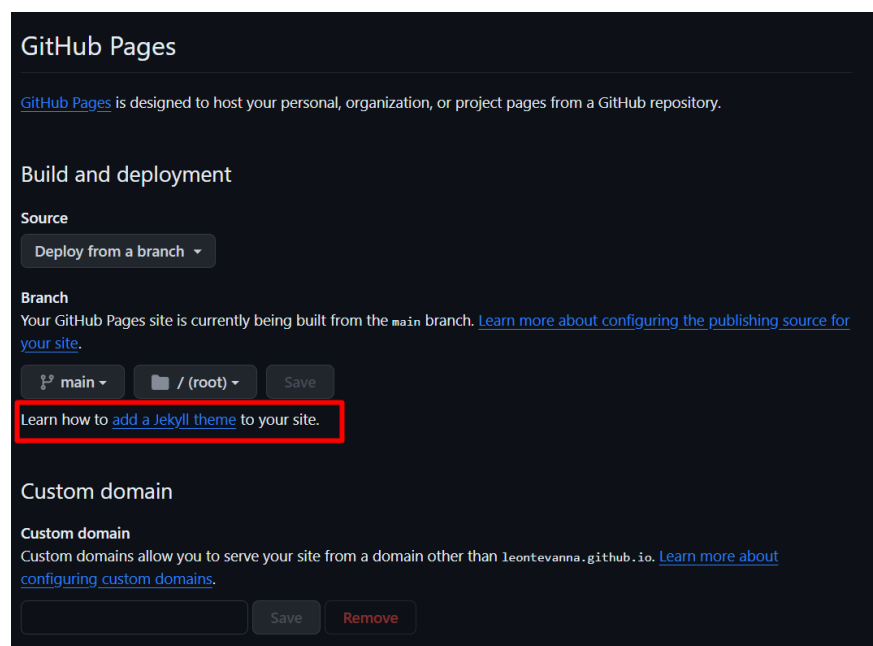


Рисунок 4

GitHub предлагает установить тему оформления сайта от Jekyll. После перехода по ссылке открывается документация GitHub по теме подключения тем оформления сайта от Jekyll. Согласно предложенной инструкции, были выполнены необходимые шаги, которые описываются далее.

GitHub Pages предлагает поддерживаемые темы Jekyll, которые представлены на отдельной странице (рисунок 5).

## Supported themes

GitHub Pages supports the following themes:

[Architect](#)  
[Cayman](#)  
[Dinky](#)  
[Hacker](#)  
[Leap day](#)  
[Merlot](#)  
[Midnight](#)  
[Minima](#)  
[Minimal](#)  
[Modernist](#)  
[Slate](#)  
[Tactile](#)  
[Time machine](#)

Рисунок 5

Была выбрана тема Cayman. Для ее установки был создан файл `_config.yml`, в котором была указана строчка, которая является инструкцией на языке YAML:

```
remote_theme: pages – themes/cayman@v0.2.0
```

После добавления файла конфигурации необходимо было подождать несколько минут, чтобы изменения отразились на сайте. На рисунке 6 можно увидеть, что изменения были успешно применены и тема Jekyll была установлена.

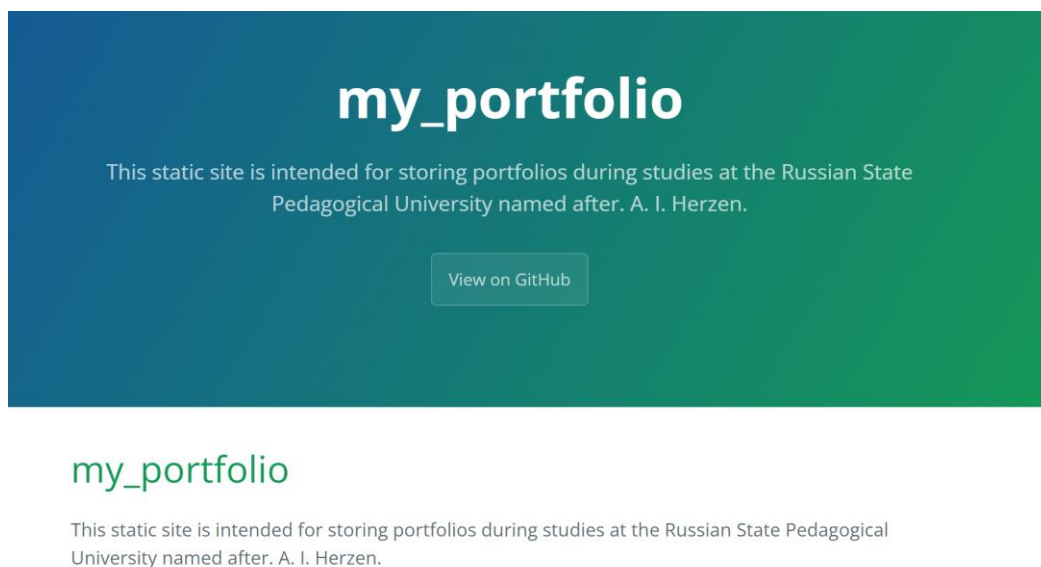


Рисунок 6

Чтобы добавить необходимый контент на сайт, необходимо было воспользоваться разметкой Markdown в файле README.md. Так как отчёты представляют собой pdf-файлы, было принято решение размещать их ссылкой на сторонний сервис: Яндекс Диск.

С помощью разметки была организована структура представления работ по дисциплинам модуля. Также для дополнительного оформления были добавлены изображения. Итоговый результат в файле README.md представлен на рисунке 7.

```
# Электронное портфолио

## На этом сайте представлено портфолио по модулю *"Информационные технологии в управлении в IT-компаниях"*
### Автор: Леонтьева А. В.


## Основы электронного управления

- Лабораторная работа #1
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)
- Лабораторная работа #2
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)
- Лабораторная работа #3
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)
- Лабораторная работа #4
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)
- Лабораторная работа #5
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)


## Системы бизнес-аналитики

- Лабораторная работа #1
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)
- Лабораторная работа #2
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)
- Лабораторная работа #3
  - [Ссылка на отчёт](https://disk.yandex.ru/i/bvSGPE4VQTyIhg)
- Лабораторная работа #4
```

Рисунок 7

## 2.2. Документальное сопровождение программного проекта

Сопроводительная документация является неотъемлемой частью программного проекта. Владелец проекта, реализуемого в данной курсовой работе, предоставил перечень обязательных документов, которые необходимо было подготовить.

Основные документы проекта:

- Документация о системе (на основании ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы»)
- Устав проекта
- План управления проектом
- База знаний

Документация о системе представляет собой документ, определяющий функциональные, архитектурные и технические решения проектируемого программного обеспечения. В документе приводятся основные сведения о реализуемом проекте.

Документация о системе реализовывалась на основании ГОСТ 34.602-89 «Техническое задание на создание автоматизированной системы» по требованию владельца проекта.

Основными задачами при написании данного документа являлись следующие пункты:

- Описание основных сведений о сторонах заказчика и исполнителя;
- Описание назначения и целей создания программного проекта;
- Описание характеристик объектов автоматизации;
- Определение требований к системе, в том числе функций подразделений ресторана, подлежащих автоматизации;
- Определение требований к содержанию системы, порядку приёма и ввода в эксплуатацию;
- Определение требований к документированию со стороны заказчика.

Была проведена отдельная встреча с разработчиками проекта для достижения максимально точных формулировок по технической части документа. Документация о системе была разработана корректно, в нужные сроки и была предоставлена владельцу проекта.

Устав проекта – это документ, в котором излагается видение проекта, его масштаб, цели, команда проекта и их обязанности, основные заинтересованные стороны, а также то, как он будет осуществляться, план реализации.

Основными задачами при написании данного документа являлись следующие пункты:

- Определение общих положений устава проекта;
- Описание нормативных ссылок на другую документацию, использующихся терминов, обозначений и сокращений;
- Описание измеримых целей и критериев успеха проекта;
- Описание высокоуровневых требований;
- Определение и описание допущений и ограничений;
- Описание совокупного риска проекта;
- Утверждение расписания и ресурсов проекта;
- Определение критериев выхода из проекта;

Устав проекта является важным документом для заинтересованных сторон проекта, так как обеспечивает юридическую безопасность в процессе реализации проекта.

Документация о системе была разработана корректно, в нужные сроки, согласована со Scrum-мастером и была предоставлена владельцу проекта.

План управления проектом (PMBoK) — это документ, описывающий, как проект будет исполняться, как будет происходить его мониторинг и контроль.

План управления проектом включает в себя:

- базовый план по содержанию;
- базовое расписание;
- базовый план по стоимости;
- план управления содержанием;
- план управления требованиями;
- план управления расписанием;
- план управления стоимостью;
- план управления качеством;
- план совершенствования процессов;
- план управления человеческими ресурсами;
- план управления коммуникациями;
- план управления рисками;
- план управления закупками;
- план управления заинтересованными сторонами.

Каждый документ-план определяет деятельность исполнителя проекта по конкретному вопросу, определяющему план.

План управления был написан силами всей команды разработчиков. Мною вносились правки в предоставленные планы при сведении в общий план управления проектом.

Документация о системе была разработана корректно, в нужные сроки и была предоставлена владельцу проекта.

База знаний по проектной деятельности (База знаний) – это электронная база знаний по организации деятельности предприятия. Она обеспечивает доступное и единое информационное поле для сотрудников предприятия. Написание и оформление базы знаний также стало задачей для команды от владельца проекта.

Для базы знаний прописывались следующие пункты:

- Сведения о компании;

- Инструкция первого дня;
- Инструкция каждого дня;
- Технические требования;
- Правила поведения;
- Документация (должностные инструкции сотрудников) ;
- Конференции;
- Ответы на часто задаваемые вопросы.

База знаний была написана силами всей команды разработчиков. Мною вносились правки в предоставленные планы при сведении в общий план управления проектом.

При написании всей документации изучались шаблоны написания соответствующих документов.

### 2.3. Анализ процесса реализации программного проекта по диаграмме сгорания задач

Диаграмма сгорания задач показывает скорость выполнения задач и оставшийся объем работы в течение спринта или проекта. По диаграмме сгорания можно спрогнозировать, успеет ли команда выполнить запланированные задачи и проанализировать работу команды.

По вертикальной оси отмечается суммарный объем задач команды, по горизонтальной – временные промежутки (дни одного спринта или спринты проекта). Таким образом, точка начала проекта располагается в верхнем левом углу, а точка конца проекта – правом нижнем.

На диаграмме можно увидеть две линии:

- Идеальная рабочая линия (голубая). Представляет собой прямую, построенную путем соединения точек начала и конца проекта. Эта линия – идеальный вариант рабочей линии, который показывает, как будут выполняться задачи, если все пойдет по плану;
- Актуальная рабочая линия (оранжевая). График визуализирует реальный ход работы над проектом.

Под анализом диаграммы сгорания задач понимается сравнение положения вышеописанных графиков относительно друг друга. Подобный способ визуализации помогает оценивать работы программы как в масштабах всего проекта, так и в отдельно взятых спринтах.

Диаграмма сгорания задач по проекту данной курсовой работы представлена на рисунке 2.3:



### *Рисунок 2.3 Диаграмма сгорания задач проекта*

Путём сравнения графиков на диаграмме, можно сделать следующие заключения.

Первую половину проекта (1-4 спринты) команда перевыполняла поставленные планы, так как актуальная рабочая линия находится под идеальной рабочей линией. Также можно говорить о том, что составленный план реализации проекта на этот промежуток времени был составлен некорректно.

Начиная со второй половины проекта, команда начинает отставать в выполнении плана. Этот вывод был сделан на основании того, что актуальная рабочая линия на этом промежутке лежит выше идеальной рабочей линии. Это можно связать с некорректно составленным планом работ или с усталостью членов команды.

В самом конце проекта команда снова начинает перевыполнять план по графику и, в итоге, закрывает проект вовремя.

Анализ диаграммы показал, что, несмотря на, местами, приличные отклонения актуальной рабочей линии от идеального графика в обе стороны, команда выполнила работу достаточно эффективно и в поставленный срок.

## ЗАКЛЮЧЕНИЕ

В ходе курсовой работы были выполнены следующие задачи:

- Были исследованы методологии и методики управления;
- Была определена рациональная для использования методика управления программным проектом;
- Была изучена выбранную методику управления. Были внедрены основные принципы в работу команду разработчиков;
- Был разработан программный проект по проектированию деятельности ресторана;
- Была подготовлена сопроводительная документация к программному продукту.

Все поставленные задачи были выполнены, что позволяет говорить о достижении цели курсовой работы.

## ЛИТЕРАТУРА