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**

«Особенности управление проектом по разработке социальной сети для книжных любителей»

Обучающегося 3 курса

Козырькова Игоря Викторовича

Научный руководитель:

Кандидат физико-математических наук,   
доцент кафедры ИТиЭО

Жуков Николай Николаевич

Санкт-Петербург

2021

**ОГЛАВЛЕНИЕ**

[ВВЕДЕНИЕ 3](#_Toc93617702)

[ОСНОВНАЯ ЧАСТЬ 4](#_Toc93617703)

[ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 4](#_Toc93617704)

[1.1Анализ предметной области 4](#_Toc93617705)

[1.2 Теоретические особенности процесса управления программным проектом. 5](#_Toc93617706)

[1.3 Техническое задание. 7](#_Toc93617707)

[ПРАКТИЧЕСКАЯ ЧАСТЬ 9](#_Toc93617708)

[2.1 Техническое задание 9](#_Toc93617709)

[2.2 Управление процессом разработки. 10](#_Toc93617710)

[2.3 Основные этапы разработки frontend – составляющей: 12](#_Toc93617711)

[2.4 Тестирование 16](#_Toc93617712)

[ЗАКЛЮЧЕНИЕ 19](#_Toc93617713)

[ЛИТЕРАТУРА 20](#_Toc93617714)

# **ВВЕДЕНИЕ**

Книги играли и продолжают играть ключевую роль в развитии нашей культуры. Гигантской, накопленной за века библиотекой является надежная память человечества, в которой запечатлены его достижения и мечтания, прозрения и заблуждения. Эту библиотеку создавали на камне и металле, глиняных табличках и деревянных дощечках, свитках папируса и пергаментных кодексах, пальмовых листьях и бересте, шелке и бумаге – изменялся материал и технология изготовления книг, но постоянным оставалось ее предназначение: хранить и передавать знания, опыт, художественные ценности. Традиционно чтение играло и продолжает играть важную социально значимую роль в обществе, в процессах социализации личности, оказывая большое влияние на развитие ее культурного потенциала. Несмотря на всю значимость чтения в современном мире наблюдается общее падение интереса к чтению.

XXI век для человечества считается информационным. Родился «виртуальный» мир, который имеет всеобщую доступность информации.  Стремление к быстрой конкретной информации, недостаточный уровень знаний ведет к снижению интереса к книге, именно поэтому социальная сеть книжных любителей имеет наибольшую актуальность в сегодняшнем дне, так как она может быть одним из тех инструментов, которые помогут вернуть былой интерес к чтению книг.

В данной курсовой работе будут рассматриваться вопросы, касающиеся создания фронтэнд-составляющей такой социальной сети.

Цель курсовой работы: Рассмотрение методы разработки, необходимы для создания социальной сети книжных любителей

**Задачи** курсовой работы:

1. Изучить и проанализировать литературу по темам «Социальная сеть», «Управление программным проектом» и «Frontend».
2. Выявить основные определения, касающиеся данных тем
3. Выбрать наиболее оптимальную методологию управления проектом для его реализации;
4. Реализовать основную часть приложения;
5. Провести ручное тестирование разработанного прототипа.

# **ОСНОВНАЯ ЧАСТЬ**

## ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

### **1.1Анализ предметной области**

Прежде всего необходимо определить, что такое социальная сеть и из чего она должна состоять.

Социальная сеть — онлайн-платформа, которая используется для общения, знакомств, создания социальных отношений между людьми, которые имеют схожие интересы или офлайн-связи, а также для развлечения и работы.

Выделяют следующие критерии социальной сети как отдельного ресурса:

* возможность созданий собственного аккаунта, хранящего личную информацию о человеке (имя, город проживания, интересы и другие социально -демографические характеристики
* возможность взаимодействия участников виртуального пространства
* возможность реализации собственных планов и идей (например, поиск новых друзей)
* возможность обмена ресурсами

Специалисты в области социальных наук считают социальные сети принципиально новым видом коммуникаций - виртуальное средство массовой коммуникации, которое обладает рядом особенностей и преимуществ:

* возможность взаимодействия различных социальных групп
* основная цель пространства - общение
* групповой характер общения
* общение рассматривается как отлаженный и систематизированный процесс
* между участниками общения может отсутствовать непосредственная связь

Существует классификация социальных сетей по их основной направленности:

* Социальные сети для общения (Relationship networks)  
  Это самые распространённые социальные сети. К ним относятся Facebook, Vk, Ok
* Социальные сети для обмена медиаконтентом (Media sharing networks)  
  Данный вид социальных медиа даёт пользователям широкие возможности для обмена видео и фото контентом. Сюда можно отнести Instagram, YouTube, Tik Tok
* Социальные сети для отзывов и обзоров (Online reviews)
* Социальные сети для коллективных обсуждений (Discussion forums)  
  К таким относятся сообщества, форумы, Q&A-сервисы (Reddit, StackOverflow, Digg)
* Социальные сети для авторских записей (Social publishing platforms)  
  К данному типу относятся сервисы для блоггинга и микроблоггинга, такие как Blogger, Twitter
* Сервисы социальных закладок (Bookmarking sites)  
  StumbleUpon, Pinterest, Flipboard — это сервисы, где пользователь собирает контент в свою личную библиотеку, на которую могут подписываться другие участники сообщества.
* Социальные сети по интересам (Interest-based networks)  
  Самой востребованной возможностью, которую предоставляют нам социальные сети — это найти единомышленников и людей со схожими интересами. Например, Last.fm - сеть для меломанов, Strava - для спортсменов, Peanut
* Социальные сети для знакомств  
  Всё большую популярность набирают сервисы для онлайн знакомств. К таким относятся Badoo, Tinder, Mamba

Таким образом, можно понять, что наша социальная сеть относится к классу “социальных сетей по интересам”, где под интересом подразумевается любовь к книгам и чтению.

### **1.2 Теоретические особенности процесса управления программным проектом.**

Классическая теория управления проектами основана на двух базовых понятиях: проекта как совокупности работ по получению уникального результата, оформленных в виде плана действий, и продукта как конечного результата реализации проекта

Проект — это комплекс взаимосвязанных действий, предпринимаемых с целью получения уникальных конкретных результатов при заданных ограничениях по времени, денежным средствам, ресурсам и качеству конечных результатов проекта.

Управление программным проектом будем рассматривать как деятельность по управлению всеми этапами ЖЦ программного продукта путем планирования, реализации, мониторинга и контроля проектных работ, в ходе которых достигаются цели проекта при нахождении компромисса между объемом работ, ресурсами, временем, качеством и рисками.

Цели программного проекта следует определять в виде желаемого результата, достигаемого командой проекта, при его успешной реализации. При формулировании целей проекта необходимо обеспечить:

* Конкретность — четкость и однозначность понимания результатов;
* Измеримость — возможность оценивания степени достижения результата, выраженного как в качественной, так и в количественной форме;
* Реальность — соизмерение возможностей команды проекта с факторами влияния внешней среды;
* Согласованность — единство мнений всех участников проекта в понимании конечного результата и координация действий по его достижению;
* Ограниченность по срокам — установление интервалов времени, по истечении которых необходимо оценивать промежуточные результаты проекта и определять степень достижения конечного результата.

Методология разработки — это набор методов по управлению разработкой ПО, набор практических правил как нужно разрабатывать ПО.

**Модель** разработки программного обеспечения описывает, какие стадии жизненного цикла оно проходит и что происходит на каждой из них. А **методология** включает в себя набор методов по управлению разработкой: это правила, техники и принципы, которые делают её более эффективной.

Существует множество различных методологий и моделей разработки. Однако на данный момент в IT-индустрии преобладают гибкие методологии разработки, такие как **Scrum** и **Kanban**

Таблица 1. Сравнение Scrum и Kanban

|  |  |  |
| --- | --- | --- |
| Методологии | scrum | kanban |
| График | Регулярные спринты с фиксированной продолжительностью (например, 2 недели) | Непрерывный процесс |
| Подходы к релизу | В конце каждого спринта | Непрерывная поставка |
| Роли | Владелец продукта, scrum-мастер, команда разработчиков | Обязательных ролей нет |
| Ключевые показатели | Скорость | Время выполнения, время цикла, объем незавершенной работы (WIP) |
| Отношения к изменениям | В ходе спринта команды не должны вносить изменения. | Изменение может произойти в любой момент |

На основе сравнения этих методологий было принято решение воспользоваться методологией Kanban, так как он более подходит для небольших команд и проектов.

### **1.3 Техническое задание.**

Важным атрибут разработки программного проекта является грамотно сформулированное техническое задание.

Во-первых, техническое задание – это, как правило, основной документ в рамках проектной документации. Именно в ТЗ описываются все основные требования на разработку программного обеспечения, будь то создание либо простенькой программы или сайта, либо же разработка крупномасштабной информационной системы или программно-аппаратного комплекса. Причем, говоря языком ГОСТов, техническое задание может разрабатываться как в рамках эскизного проекта (это когда только описание функций и структуры системы без рассмотрения технологий реализации решения), так и в дальнейшем «перекочевать» в технический проект (более детальное описание с учетом выбранных технологий). Наше техническое задание мы подробнее рассмотрим в практической части курсовой работы. Далее мы перейдем к основным этапам разработки frontend составляющей.

## ПРАКТИЧЕСКАЯ ЧАСТЬ

### **2.1 Техническое задание**

**Назначение разработки / Общие сведения**

Проект является веб-приложением, представляющим собой социальную сеть для книжных любителей.

**Требования к программной совместимости**

Веб-приложение должно корректно работать в различных браузерах на устройствах с разным разрешением экрана (десктоп и мобильной версии)

**Требования к содержимому сайта**

Веб-приложение должно содержать следующие страницы:

* Страница регистрации и аутентификации
* Страница со списком книг
* Страница для каждой книги отдельно
* Страница автора
* Страница профиля пользователя
* Страница диалогов пользователя (мессенджер)

**Требования к функциональности приложения**

Система должна позволять пользователю следующие манипуляции:

1. Производить регистрацию и аутентификацию
2. Осуществлять коммуникацию с другими пользователями
3. Производить различные манипуляции с книгами (ставить лайки/дизлайки, оставлять отзывы)
4. Редактировать свой профиль
5. Подписываться на других пользователей

**Требования к надёжности приложения**

Приложение должно предусматривать базовую защиту от основных видов атак: межсайтового скриптинга (XSS), SQL-инъекций, CSRF-уязвимостей.

**Требования к информационной и программной совместимости**

Приложение должно быть запущено на сервере под управлением ОС семейства Linux (Debian или Ubuntu) или Microsoft Server

### **2.2 Управление процессом разработки.**

Инструмент управления проектом.

Для реализации доски Kanban был использован онлайн сервис Jira Atlassian. На рабочем столе было выделено несколько колонок с задачами для всей команды разработки:

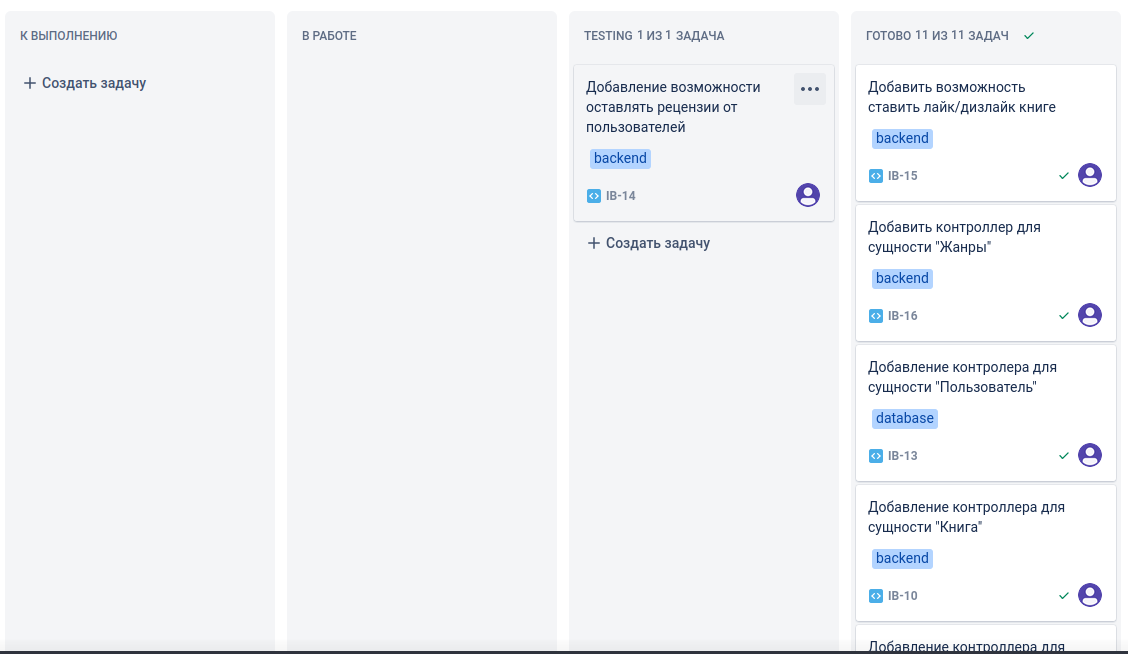


Рис.1Kanban доска.

Для визуализации примерного плана разработки frontend-составляющей проекта была создана диаграмма Ганта.

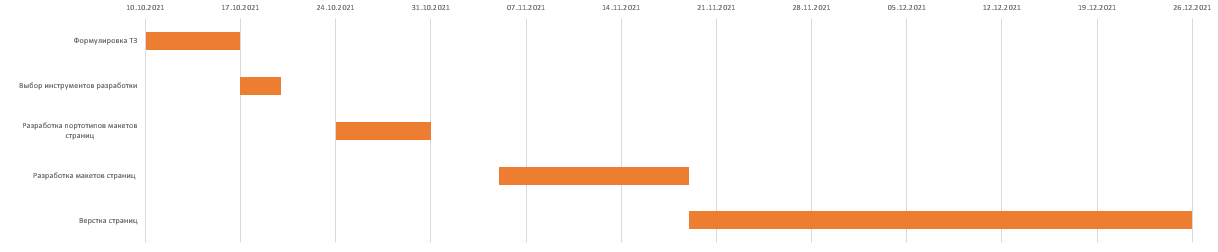
****

Рис. 2 Диаграмма Ганта.

Рассмотрим подробнее сервис. Который мы использовали при разработке:

Atlassian (Jira Software):

Jira — это набор agile-решений для управления работой, который обеспечивает совместную работу между всеми командами, начиная с разработки идеи продукта и заканчивая его поставкой клиенту, а также позволяет вам вместе добиваться наилучших результатов. Jira предлагает ряд продуктов и вариантов развертывания, специально разработанных для команд разработчиков ПО, ИТ-команд, бизнес-команд, операционных команд и т. д.

«Задача» Jira обозначает отдельную рабочую задачу любого типа или размера, которая отслеживается от создания до завершения. К примеру, задачей может быть функциональная возможность, которую разрабатывает команда разработчиков, или задача, стоящая перед командой маркетологов, или контракт, который необходимо составить команде юристов.

Основные наборы возможностей Jira Software были разработаны специально для работы по методикам agile, в том числе scrum и kanban. Методика Kanban отлично подходит для разработки социальной сети книжных любителей в рамках курсовой работы:

Методология Kanban была выбрана не просто так, дело в том, что в зависимости от выбора методологии зависит и сам процесс разработки. В случае с методологией Scrum, постановка задач производилась бы на отдельные промежутки времени «sprint», по итогам которых завершается некоторый из этапов разработки. Данная схема больше пригодна для более крупных команд, чем в нашем случае, также данную методологию лучше применить для коммерческих разработок, где существуют четкие дедлайны, и каждый член команды ответственен строго за свою задачу. В случае же разработки нашего проекта, это нам не подходит, так как команда состоит всего из трех человек, а также работа осуществляется в рамках курсовой работы, поэтому и постановка задач будет отлична от методологии Scrum. В нашем случае мы не разрабатываем коммерческий проект, соответственно нет необходимости в спринтах, наша цель получить некоторую часть программного продукта к концу общего срока разработки. Также обязанности между участниками проекта могут быть не такими четкими, как в методологии scrum. Поэтому вариант, в котором мы используем Kanban доску для постановки задач нам подходит больше всего.

Реализация методологии в нашем проекте выглядит следующим образом: Никитой были добавлены метки для разделения на разные типы задач: backend, database, frontend, devops. В тот момент, когда мы добавляем новую задачу на доску, карточке присваивается исполнитель, срок выполнения и метка задачи. Только что созданные карточки отправляются в колонку «К выполнению». Как только член команды начинает выполнять задачу из первой колонки нужно перенести задачу в колонку «В работе». Для задач связанных с backend составляющей Никита создал колонку Testing, куда перемещались реализованные задачи для их тестирования. Как только задача проходила тестирование и исправления, она перемещалась в последнюю колонку «Готово». Также был реализован принцип дочерних и родительских задач. То есть, когда мы ставим задачу мы определяем ее как дочернею или родительскую для того, чтобы реализовывать задачи в правильном порядке. Дочерняя задача может быть выполнена, только когда выполнена родительская задача.

### **2.3 Основные этапы разработки frontend – составляющей:**

Frontend-разработчики занимаются версткой — создают структуру гипертекстового документа на основе HTML-разметки, как правило, используя таблицы стилей (CSS) и клиентские сценарии (JS), чтобы элементы дизайна выглядели аналогично макету. Простыми словами frontend-специалисты «оживляют» проекты дизайнеров: добавляют анимацию для удобства и улучшения восприятия, делают макеты интерактивными. Благодаря верстке, веб-страницы можно открыть в браузере и «пощупать» элементы мышкой.

Вместе с разметкой страниц сайта, готовых к интеграции (программированию), frontend-разработчики предоставляют дополнительный функционал (виджеты, календари, стилизованные элементы, заменяющие нативные карты, диаграммы и т. п.), а также Style Guide с набором всех функциональных блоков дизайна, размещенных на одной странице верстки. Style Guide позволяет показать, как нужные элементы дизайна, будучи вырванными из контекста, должны выглядеть на странице.

Теперь рассмотрим этапы разработки:

**Этап 1: Разработка макетов страниц:**

Был составлен приблизительный макет приложения дизайнерами команды, на основе которых необходимо передать в View нужные объекты и данные, по которым в будущем будет верстаться фронт.

Страница профиля:

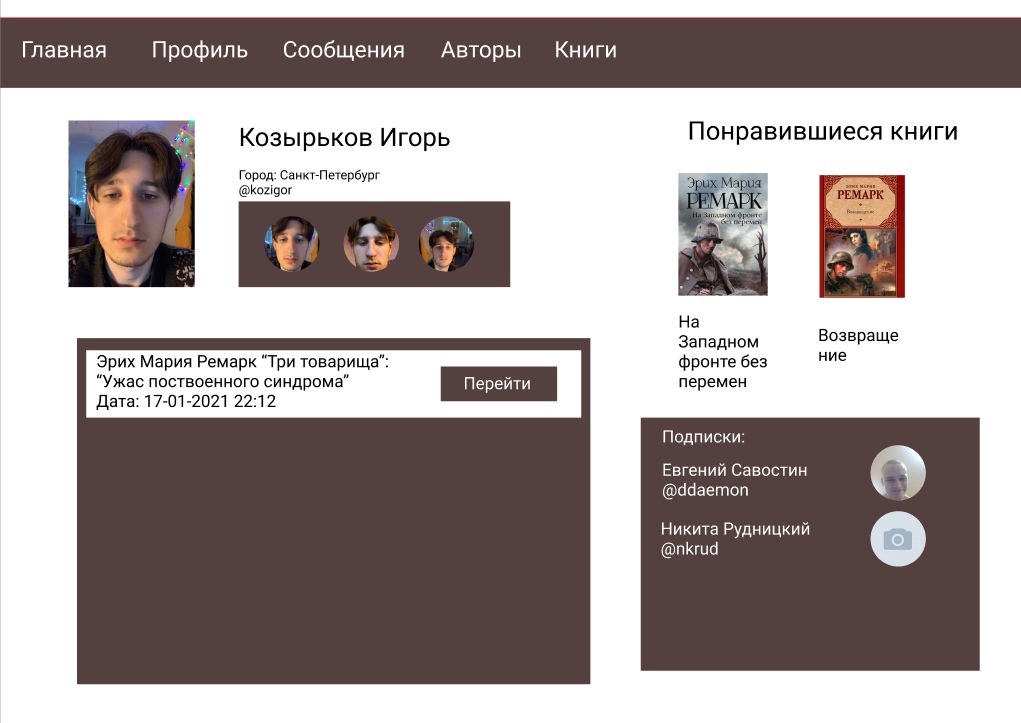


Рис.3 Макет страницы профиля.

Страница книги:

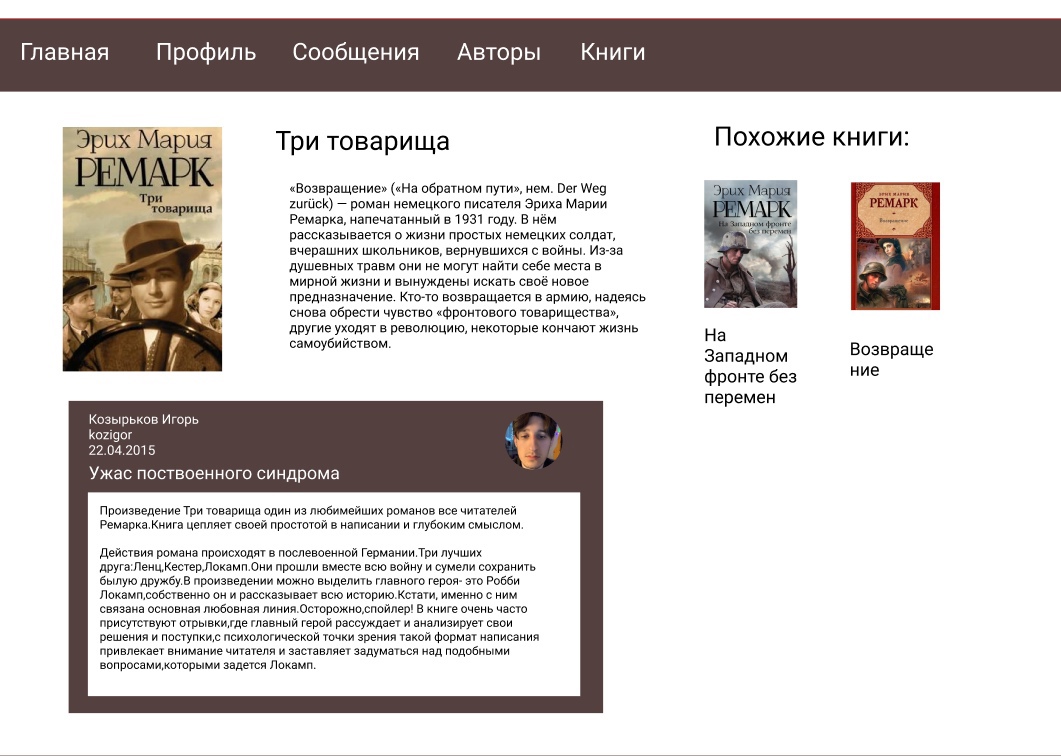


Рис.4 Макет страницы Книги.

Страница автора:

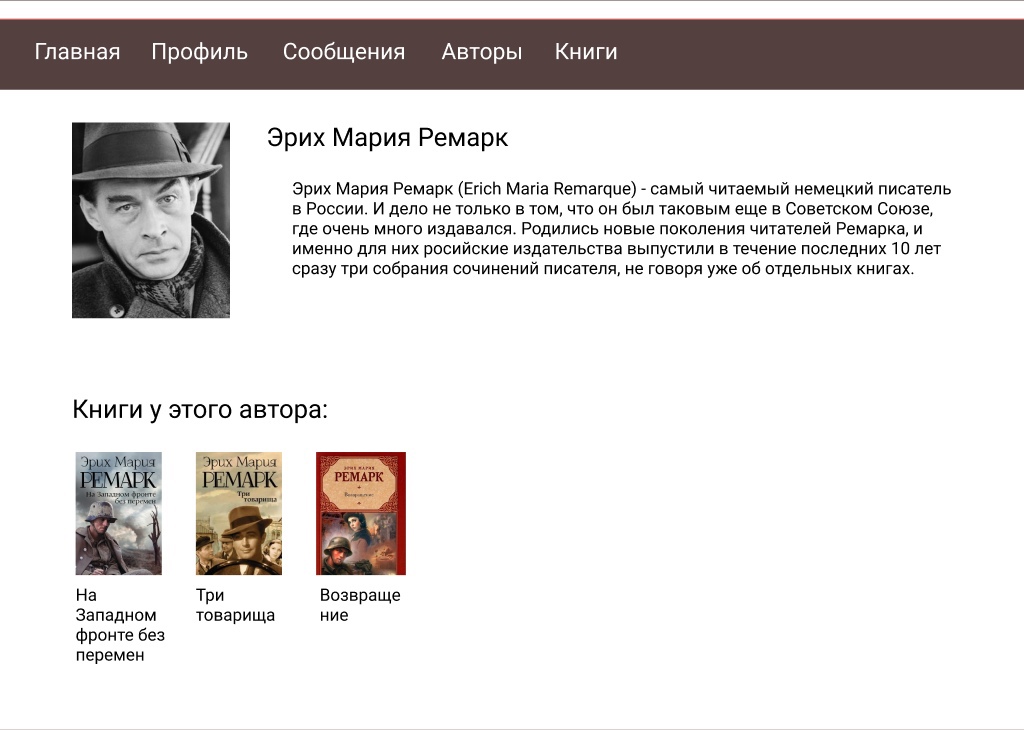


Рис.5 Макет страницы автора.

**Этап 2: Верстка (HTML,CSS):** На данном этапе нам необходимо с помощью HTML и CSS реализовать, те макеты, которые были разработаны в прошлом этапе в виде готовых страниц.

Страница регистрации и входа:

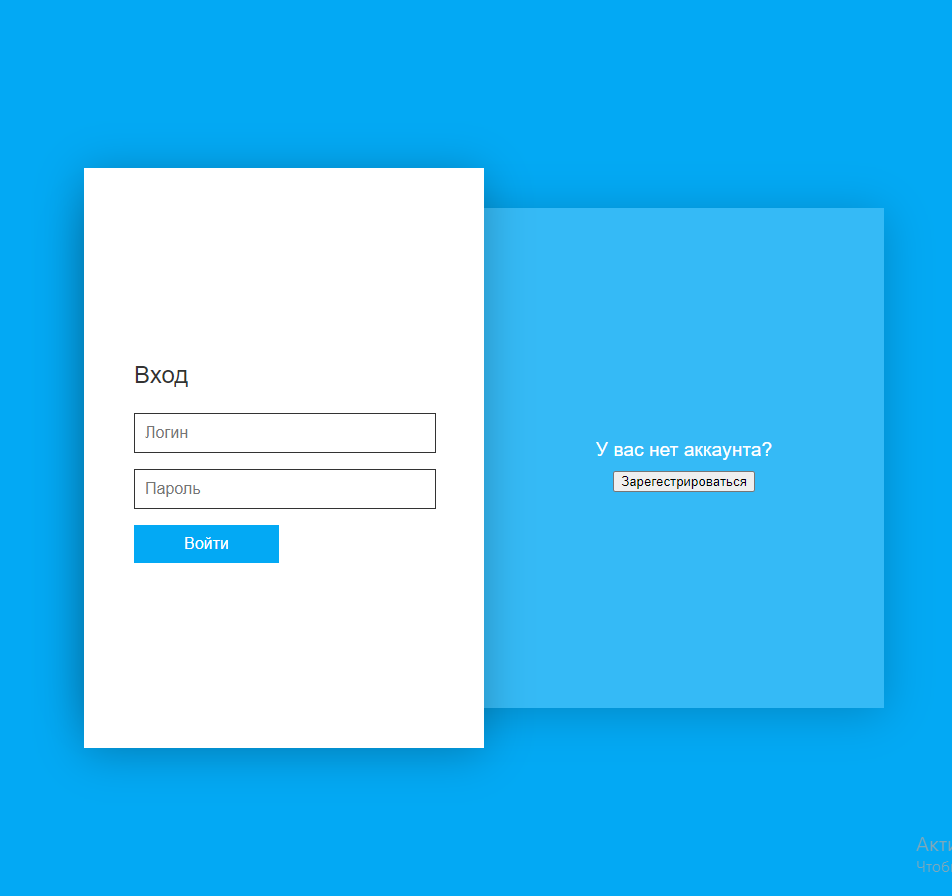


Рис.6 Свёрстанная страница авторизации/регистрации.

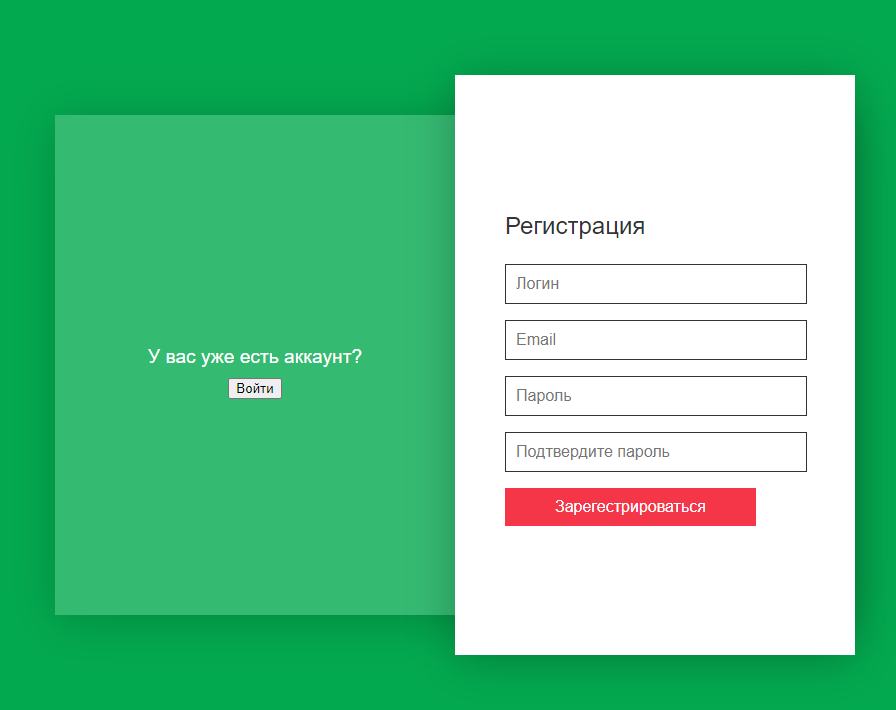


Рис.7 Рис.6 Свёрстанная страница авторизации/регистрации.

Страница книги:

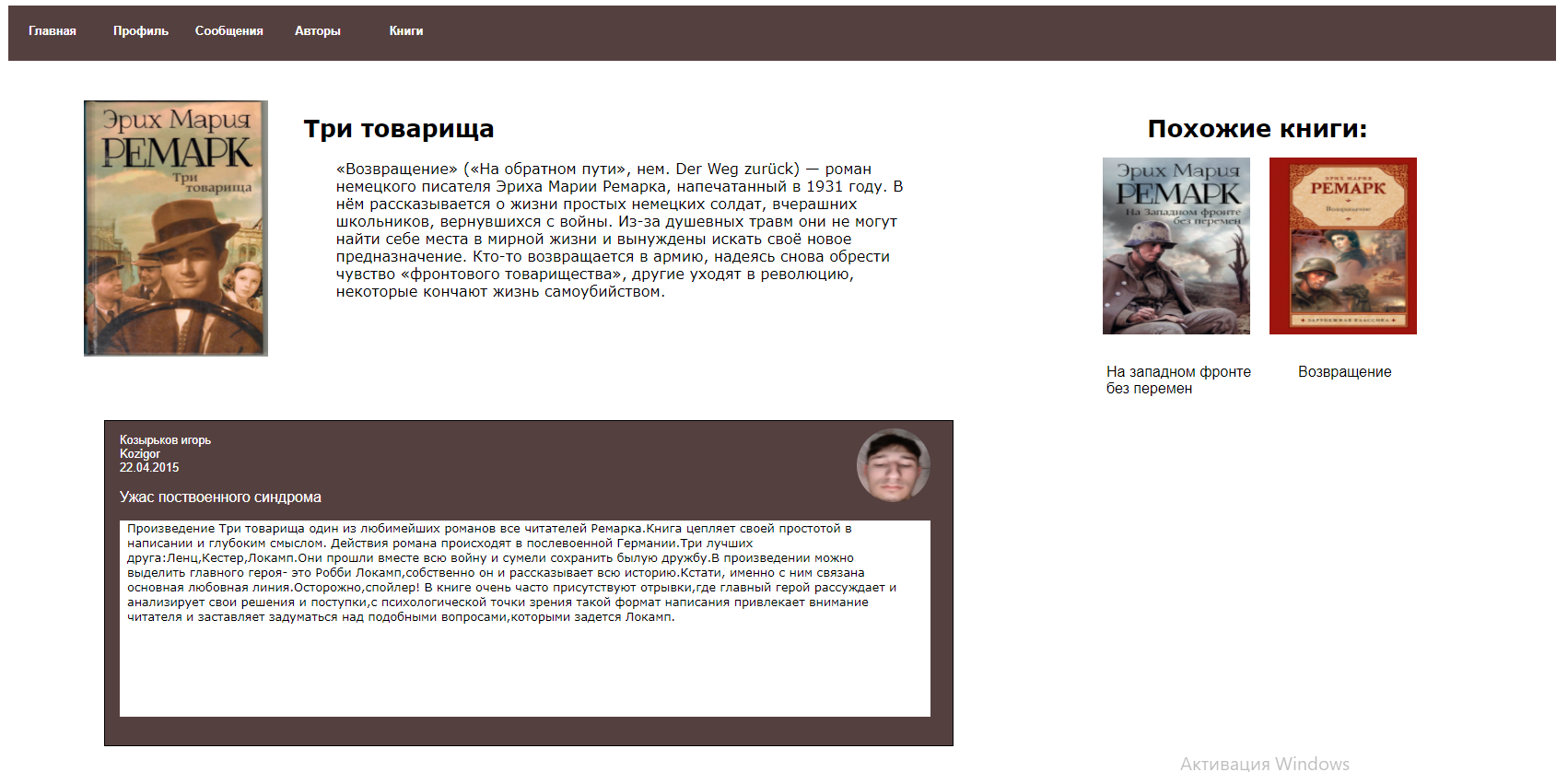


Рис.8 Свёрстанная страница книги.

**Этап 3: Сборка front и back составляющих воедино**

**Структура взаимодействия backend и frontend**

Серверные приложения:

В этом случае HTTP-запросы отправляются напрямую на сервер приложения, а сервер отвечает HTML-страницей.

Между получением запроса и ответом сервер обычно ищет по запросу информацию в базе данных и встраивает ее в шаблон (ERB, Blade, EJS, Handlebars).

Когда страница загружена в браузере, HTML определяет, что будет показано, CSS — как это будет выглядеть, а JS — всякие особые взаимодействия.

Связь с использованием AJAX:

Другой тип архитектуры использует для связи AJAX (Asynchronous JavaScript and XML). Это означает, что JavaScript, загруженный в браузере, отправляет HTTP-запрос (XHR, XML HTTP Request) изнутри страницы и (так сложилось исторически) получает XML-ответ. Сейчас для ответов также можно использовать формат JSON.

Это значит, что у вашего сервера должна быть конечная точка, которая отвечает на запросы JSON- или XML-кодом. Два примера протоколов, используемых для этого — REST и SOAP.

Клиентские (одностраничные) приложения:

AJAX позволяет вам загружать данные без обновления страницы. Больше всего это используется в таких фреймворках, как Angular и Ember. После сборки такие приложения отправляются в браузер, и любой последующий рендеринг выполняется на стороне клиента (в браузере).

Такой frontend общается с бэкендом через HTTP, используя JSON- или XML-ответы.

### **2.4 Тестирование**

Для реализации цикла тестирования к Jira был подключен плагин Zephyr Squad, который предоставляет большое количество возможностей для QA. Для некоторых задач были составлены тест-кейсы

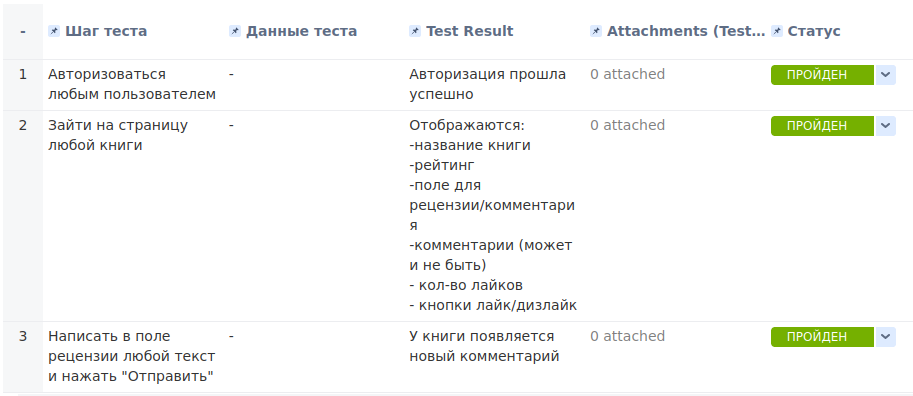


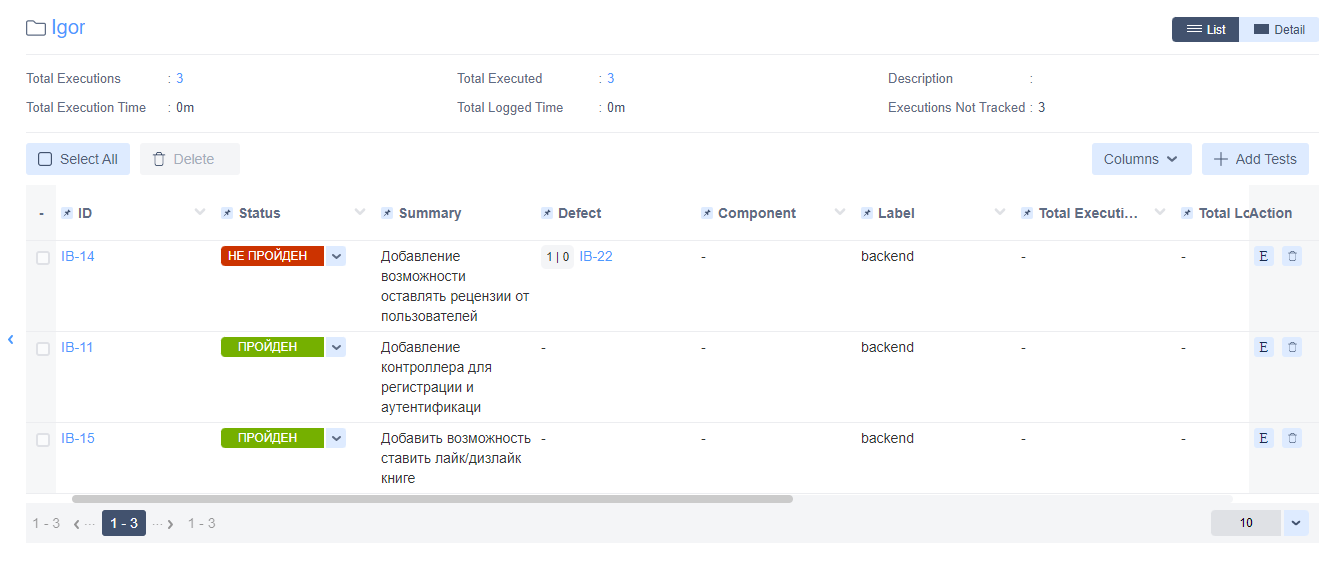
Рис.9.1 тест-кейс для проверки комментирования



Рис.9.2 Рисунок 5.2 - тест-кейс для проверки авторизации и регистрации



Рис 9.3 Рисунок 5.3 - тест-кейс для проверки лайков и дизлайков книги



9.4 – отметки о прохождении тестов

# **ЗАКЛЮЧЕНИЕ**

Появление новых средств массовой информации и новых технологий сузило книжную сферу, однако именно их и надо использовать для расширения интереса к чтению книг. Специалисты по книжному маркетингу в этих виртуальных пространствах могут при помощи эффективных маркетинговых технологий рекламировать и продавать каждую книгу в отдельности. Наличие большого количества компаний, которые изо дня в день посредством новейших технологий, соответствующих социально-культурным условиям, присоединяются к средствам массовой информации, продвигающие книги в массы и не дающие им уйти в забвение, доказывает тот факт, что путём слияния с этими группами можно заниматься увеличением интереса к книгам. Наш проект как раз смотрит в данном направлении, мы хотим, чтобы не смотря на активный рост различных источников информации, которые могут заменить книги, люди читали и интересовались ими как можно больше.

В результате выполнения курсовой работы были выполнены следующие задачи:

Была изучена литература по таким темам, как: «Социальная сеть» и «Управление программными проектами». Были проанализированы различные методологии и выбрана подходящая под наш проект. В результате анализа и выполнения проекта по разработке социальной сети книжных любителей были выявлены особенности разработки программного продукта. В ходе разработки было разработано техническое задание, макет приложения, часть frontend, backend – составляющих, а также проведено регрессионное – тестирование для выявления багов.

# **ЛИТЕРАТУРА**

1. Гультяев А.К. Microsoft Project. Управление проектами: Русифицированная версия. – М.: Корона принт, 2003.
2. Дитхелм Г. Управление проектами. Т. 1-2: Пер.с нем. – Спб: Бизнес-пресса, 2003.
3. Евсеев Д.А. Особенности обучения управляющих проектами навыкам работы в среде Primavera //Вестник ПМСофт. Журнал по управлению проектами для профессионалов. №1, 2005. — С.34-37.
4. Карпова В.С. Опыт использования Primavera на фазе инициации проекта. //Вестник ПМСофт. Журнал по управлению проектами для профессионалов. №1, 2005. — С.26-30.
5. Королев Д. Эффективное управление проектами. – М.: ОЛМА-ПРЕСС Инвест: Ин-т экономических стратегий, 2003.
6. Разработка технического задания (ТЗ) по ГОСТ - написание и оформление техзадания, стоимость - URL: <http://www.swrit.ru/tekhnicheskoe-zadanie.html>
7. Культин Н.Б. Инструменты управления проектами: Project Expert и Microsoft Project. – СПб.: БХВ-Петербург, 2009. – 160 с. Ил.
8. Культин Н.Б. Управление проектами: инструментальные средства. – СПб.: Поли-техника, 2002. – 216 с. Ил..
9. Медынский В.Г. Инновационный менеджмент: Учебник. – М.: ИНФРА-М, 2004.
10. Использование Jira Software | Официальное руководство - URL: <https://www.atlassian.com/ru/software/jira/guides>
11. Блинов И. Н. Java. Промышленное программирование - Минск : УниверсалПресс, 2007. -704 с.
12. Блинов И. Н. Java. Методы программирования / И. Н. Блинов, В. С. Романчик. - Минск : Издательство Четыре Четверти, 2013 - 896 с.