МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОМСКОЙ ОБЛАСТИ

бюджетное профессиональное образовательное учреждение Омской области

«ОМСКИЙ АВТОТРАНСПОРТНЫЙ КОЛЛЕДЖ»

(БПОУ «Омский АТК»)

**Специальность: 09.02.03 «Программирование в компьютерных системах»**

**КУРСОВОЙ ПРОЕКТ**

**по МДК.01.01 Системное программирование**

**Тема: *Проектирование автоматизированного приложения для создания кейсов задач***

Выполнил студент группы ПКС351

Борминцев А. А.

Проверил преподаватель

Курчевский В.Е.

Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Омск 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОМСКОЙ ОБЛАСТИ

бюджетное профессиональное образовательное учреждение Омской области

«ОМСКИЙ АВТОТРАНСПОРТНЫЙ КОЛЛЕДЖ»

(БПОУ «Омский АТК»)

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ |
|  | Зам. директора |
|  | \_\_\_\_\_\_\_\_\_\_\_\_\_И.В. Сидоренко |
|  | «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_ г. |
|  |  |

##### **ЗАДАНИЕ**

**на курсовой проект**

**по МДК.01.01 Системное программирование**

студенту Борминцеву Андрею Алексеевичу

группы ПКС351 курса 4

специальность 09.02.03 «Программирование в компьютерных системах»

**Тема: *Проектирование автоматизированного приложения для создания кейсов задач***

**ВВЕДЕНИЕ**

Актуальность заданной темы, цели и задачи работы.

**1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ**

1.1 Описание предметной области

1.2 Анализ существующих аналогов

1.3 Техническое задание

**2 ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ**

2.1 Назначение разработки

2.2 Требования к проекту

2.3 Выбор программных и технических средств

2.4 Проектирование системы

**3 ОРГАНИЗАЦИОННЫЙ РАЗДЕЛ**

3.1 Организация графика разработки.

3.2 Система контроля версий.

**ЗАКЛЮЧЕНИЕ**

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

**ПРИЛОЖЕНИЕ**

Дата выдачи задания на курсовой проект «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

Срок выполнения курсового проекта «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

**Руководитель курсового проекта** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

Задание рассмотрено и одобрено на заседании ЦМК «Информатика и вычислительная техника. Математика» от «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

Протокол №\_\_\_\_\_\_ Председатель ЦМК \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Харченко В.М./

Задание получил «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_г.

**Студент**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ / \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc5819932)

[1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ 5](#_Toc5819933)

[1.1 Основные понятия 5](#_Toc5819934)

[1.2 Анализ существующих аналогов 6](#_Toc5819935)

[1.3 Техническое задание 10](#_Toc5819936)

[2 ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ 11](#_Toc5819937)

[2.1 Назначение разработки 11](#_Toc5819938)

[2.2 Требования к проекту: 12](#_Toc5819939)

[2.3 Проектирование системы 16](#_Toc5819940)

[2.4 UML Диаграммы. 21](#_Toc5819941)

[2.5 Выбор программных и технических средств 27](#_Toc5819942)

[3 ОРГАНИЗАЦИОННЫЙ РАЗДЕЛ 29](#_Toc5819943)

[3.1 Руководство пользователя 29](#_Toc5819944)

[ЗАКЛЮЧЕНИЕ 30](#_Toc5819945)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 31](#_Toc5819946)

[ПРИЛОЖЕНИЕ 32](#_Toc5819947)

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

ДП.09.02.04.ИС.О.ПЗ.01

Разраб.

Альтмаер О.Г

Провер.

ШарковаО.В..

Т. Контр.

Субботина И.В.

Утверд.

Субботина И.В.

Пояснительная записка

Разработка информационной системы страхования автотранспорта

Лит.

Листов

52

БПОУ «Омский АТК»

гр.ИС346

**ВВЕДЕНИЕ**

В настоящее время люди работают большими группами в целях решения определенной задачи. Для достижения наилучшего результата в решении большой задачи её разбивают на маленькие определённые задачи между членами группы. Для передачи данной информации можно использовать разные накопители информации начиная от бумаги и заканчивая специальными приложениями социального взаимодействия. Открытых и бесплатных программ для создания кейсов задач, подходящих к любому предприятию мало и чаще они разрабатываются отдельно для каждого предприятия. Именно эти факторы подтолкнули меня к проектированию данного приложения для передачи кейс задач.

Цель проекта: спроектировать приложения для создания кейс задач.

Задачи, которые следует выполнить во время исполнения проекта:

1) Собрать информацию по аналогичной теме и провести функциональный анализ этой информации,

2) Составить требования к проекту,

3) Спроектировать базу данных и разработать будущий дизайн приложения,

4) Реализовать задуманное.

1. **ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ**
   1. **Основные понятия**

С#— объектно-ориентированный язык программирования. Разработан в 1998—2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft .NET Framework. Впоследствии был стандартизирован как ECMA-334 и ISO/IEC 23270.

UML – Это система обозначений, которую можно применять для объектно – ориентированного анализа и проектирования. Его можно использовать для визуализации, спецификации, конструирования и документирования программных систем.

SQL – Декларативный язык программирования, применяемый для создания, модификаций и управления данными в реляционной базе данных, управляемой соответствующей системой управления базами данных.

* 1. **Анализ существующих аналогов**

1.2.1 Jira

Jira – Мощный онлайн сервис позволяющий командам – разработчикам планировать проекты, назначать исполнителей задач, планировать определенные цели и объёмы работ, собирать задачи в журнал оставшейся работы необходимой выполнить команде, выставлять приоритеты и крайние сроки. (рис. 1.1).

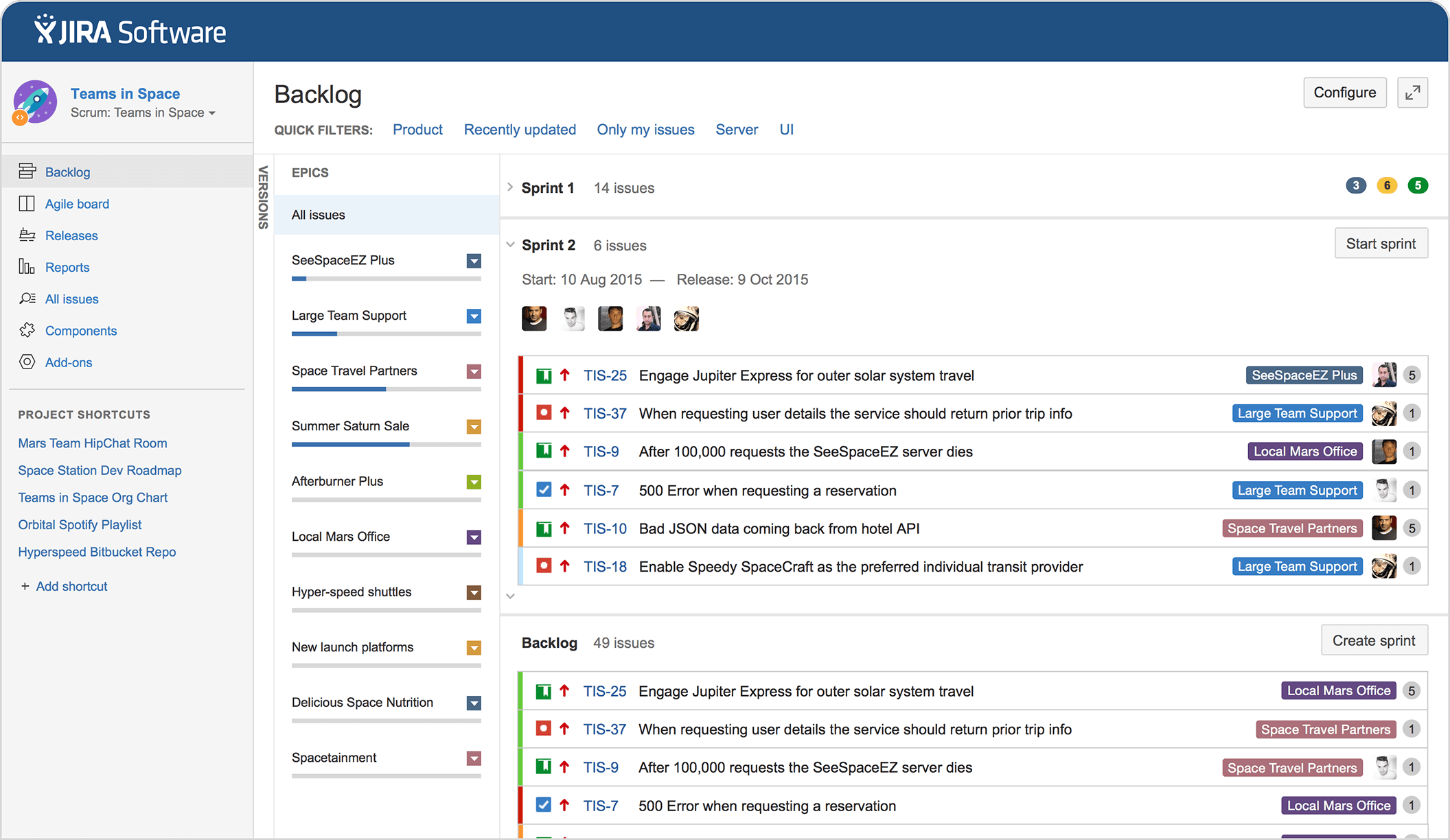


Рисунок 1.1 – Главное окно программы Jira

Преимущества данной программы: Приятный внешний вид, большое количество настроек фильтрации, перетаскивание задач в бэклоге и спринте, детальные отчёты, простота в управление проектами.

Минусы данной программы: Нет возможности назначения нескольких исполнителей для одной задачи, настройки фильтрации могут заблокировать уже используемые настройки, трудный порог вхождения из-за большого количества настроек.

1.2.2 Redbooth

Redbooth–Сервис позволяющий отслеживать ошибки, планировать проекты и активности, управлять ресурсами и ставить задачи участникам проекта, устанавливать крайние сроки и ключевые моменты, следить за расходами проекта. Также сервис предлагает делиться файлами и синхронизацию с Google Drive, Gmail и Outlook. (рис 1.2).

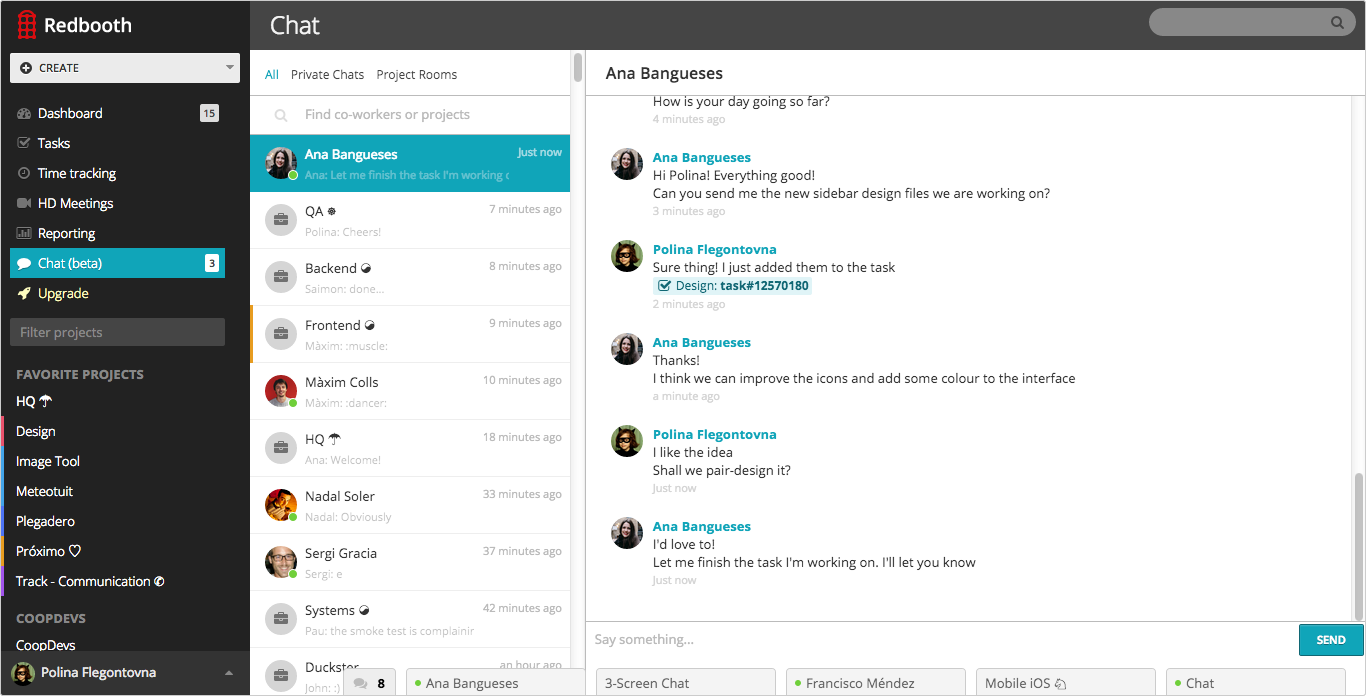


Рисунок 1.2 – Главное окно программы Redbooth

Преимущества данной программы: Простой и интуитивный интерфейс, возможность чата, интеграция с другими сервисами, возможность создания отчетов.

Минусы данной программы: Нет выставления процента выполнения задач и слежения за прогрессом проекта в целом, нет возможности выставления приоритетов задач.

1.2.3 GanttPro

GanttPro – Диаграмма Ганта, позволяющая планировать и управлять проектами онлайн, визуализировать процессы, создавать задачи и назначать их участникам, выставлять конечные сроки и процент завершения отдельных задач и проекта в целом, добавлять ключевые моменты, а также делиться созданным графиком Ганта с командой и с клиентами с правом просмотра и редактирования. (рис. 1.3).

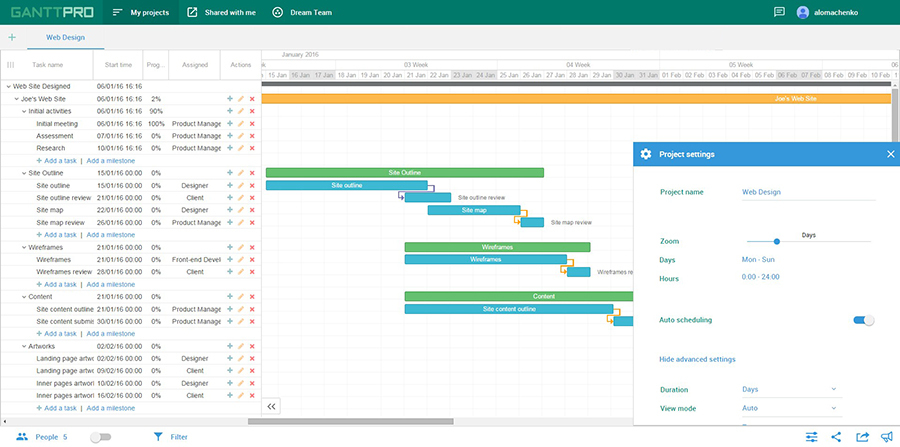


Рисунок 1.3 – Главное окно программы GanttPro

Преимущества данного приложения: Наличие возможности автоматического планирования проектов, перетаскивание задач, возможность выставлять собственные рабочие дни и часы, автоматический пересчет длительностей задач и проектов, выставление зависимостей между задачами проекта, назначение прав доступа.

Минусы данного приложения: Нет возможности выставить приоритеты задач, отсутствуют уведомления, нет интеграции с Jira и Slack.

1.2.4 Basecamp

Basecamp – Один из самых популярных сервисов для совместной работы над проектами, позволяющий делиться документами, вести обсуждения с командой, создавать to – do листы и добавлять комментарии к задачам, высылать и принимать электронную почту (рис. 1.4).

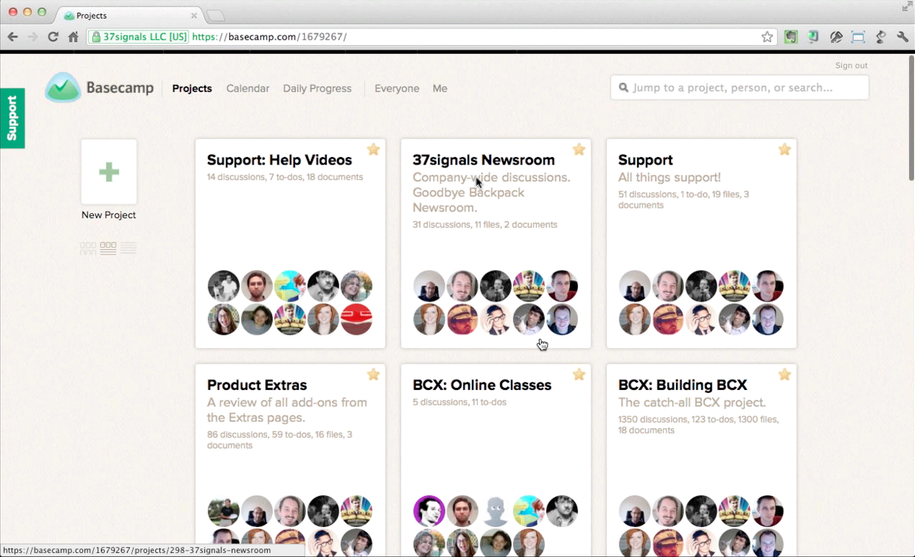


Рисунок 1.4 – Главная окно программы Basecamp

Преимущества данного приложения: Прост в использовании, наличие мгновенных уведомлений об обновлениях.

Минусы данного приложения: Нет возможности для тактического и стратегического планирования, не хватает особенностей для отслеживания ошибок, невозможно оценить полную картину проекта.

* 1. **Техническое задание**

Программа была разработана с помощью IDE Microsoft Visual Studio 2017 на языке c#, а для редактирования картинок и элементов дизайна использовался Adobe Photoshop. Система управления Базами данных была выбрана MS SQL и была спроектирована с помощью специальных служб в Visual Studio.

Меню программы состоит из разделов: Авторизация, Коллеги, Сообщения, Проект.

Раздел Авторизация состоит из разделов: Вход, Регистрация.

Раздел Проект состоит из разделов: Создать проект, Процесс выполнения, Редактирование проектов.

Для создания графики будет использоваться WPF, которой присущи свои пространства имён для создания насыщенной графики на языке C#.

1. **ТЕХНОЛОГИЧЕСКИЙ РАЗДЕЛ**
   1. **Назначение разработки**

Цель: Приложение разрабатывается с целью автоматизации процесса распределения задач между группами сотрудников, предоставить возможность вести удаленную коммуникацию между собой. Также разработка данного приложения послужит мне хорошей практикой в разработке приложений на платформе .Net.

Задачи:

1) Создать базу данных,

2) Разработать интерфейс и дизайн приложения,

3) Разработать функциональную часть приложения,

4) Разработка регистрации новых пользователей и авторизации старых пользователей или рабочего персонала, т.е. администрации,

5) Воплотить мой проект в реальность.

* 1. **Требования к проекту**
     1. Требование к безопасности

Для того, чтобы обеспечить безопасность паролей пользователям, я использовал метод шифрования MD5(рис. 2.1).

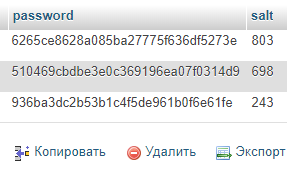


Рисунок 2.1 – Метод шифрования MD5

* + 1. Требование к графическому интерфейсу

2.2.2.1. Окно входа в приложении.

На рисунке 2.2 изображено окно входа в программу. На ней будут располагаться два раздела: Вход, Регистрация.

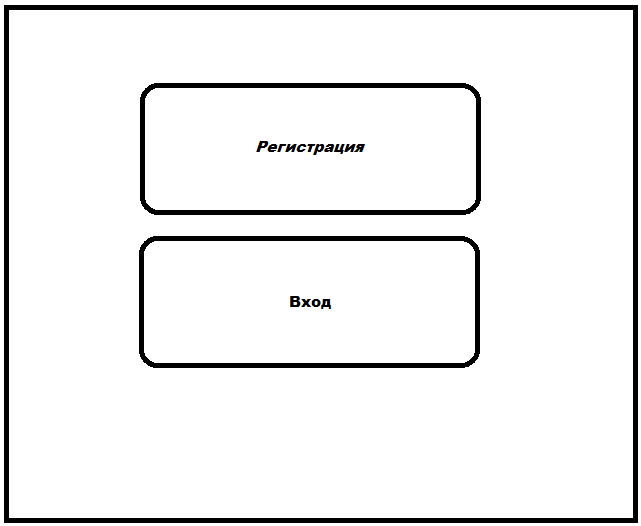


Рисунок 2.2 – Окно авторизации

2.2.2.2. Форма авторизации пользователя

На рисунке 2.3 представлена форма авторизации пользователя для входа в программу.

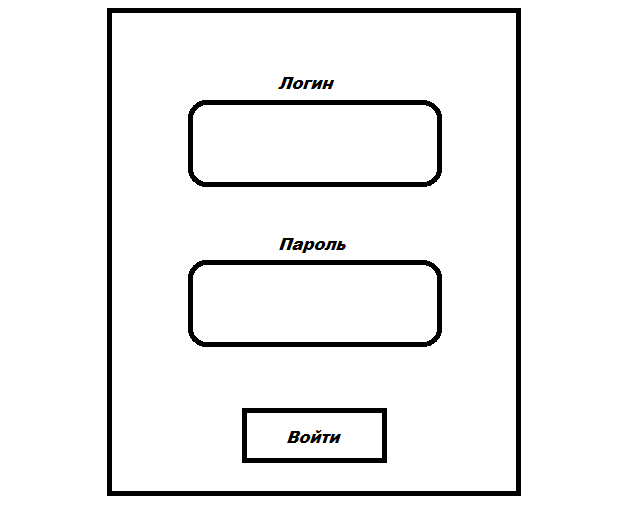


Рисунок 2.3 – Форма авторизации

2.2.2.3. Форма регистрации пользователя

На рисунке 2.4 представлена форма для регистрации пользователя.

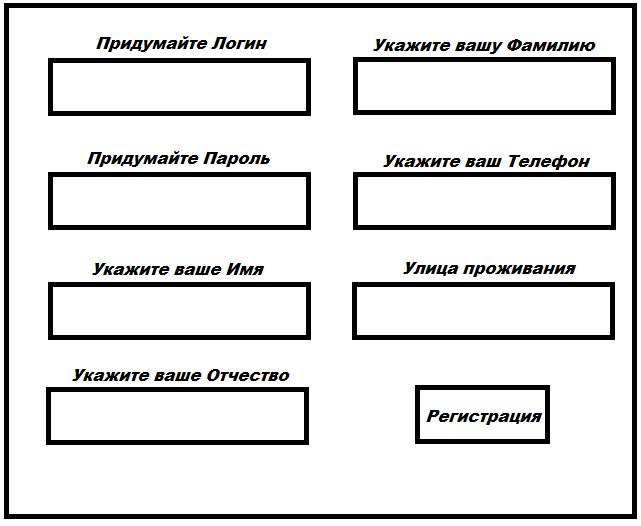


Рисунок 2.4 – Форма регистрации

2.2.2.4 Главное окно приложения

На рисунке 2.5 представлена главная форма, на которой располагается весь основной функционал.

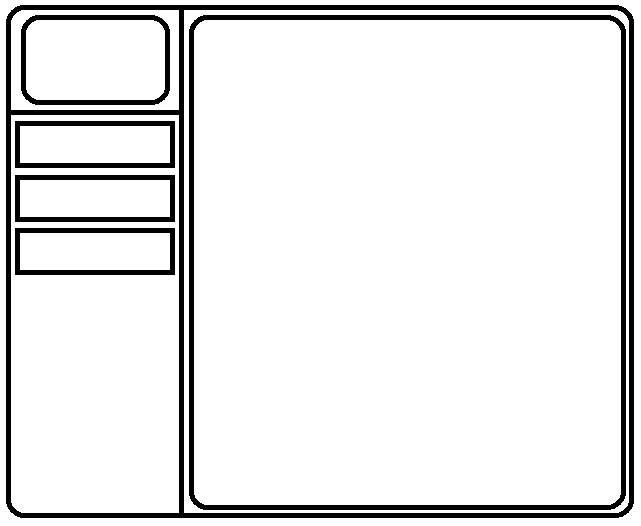


Рисунок 2.5 – Главное форма приложения

* + 1. Требование к функциональной части

Обеспечить рабочую систему c авторизацией и регистрацией пользователя. Обеспечить фильтрацию определенных записей в таких разделах: Коллеги, Сообщения, Проекты.

* + 1. Требование к техническим характеристикам.

Характеристики компьютера:

|  |  |
| --- | --- |
| Параметры: | Характеристики: |
| Форм-фактор сервера | 1U (rack) |
| Модель процессоров | Xeon E5-2620 v4 |
| Объем установленной оперативной памяти | 4 Гб |
| Тип установленных накопителей | HDD |
| Максимально возможный суммарный объем накопителей | 40 Тб |
| Мощность системы питания | 450 Вт |
| Сетевой адаптер | HP 331i |
| Скорость передачи данных Ethernet | 10/100/1000 Мбит/с |

* 1. **Проектирование системы**

На протяжении подготовки приложения выявлялось множество вариантов пути развития, но в тоже время их реализация оставляла желать лучшего. В итоге получилось то, что получилось: плюсов оказалось в несколько раз меньше, чем минусов. Однако это не означает, что цель была провалена. Основополагающим фактором является текст и визуальное оформление приятное глазу и привлекающее должное внимание. Оценку эксперимента можно будет провести после осмотра сайта должностными лицами.

Разработка Базы данных состоит из трех этапов: организация, создание логической структуры и проектирование.

В первом этапе следует определиться с тем, что именно будет присутствовать на приложении и на этом основании формировать таблицы в Базе данных, т.е. определиться с их темой, количеством.

На втором этапе стоит разработать организацию и связью между таблицами. В данном случаи связи играют огромную роль и очень важно правильно их организовать в целях избежания различных аномалий.

Заключительный этап подразумевает преобразование идей в физическую реализацию, происходит создание самой базы данных.

В данном приложении имеются четыре таблицы: (рис. 2.6)

* Авторизация;
* Пользователь;
* Группа;
* Проект.

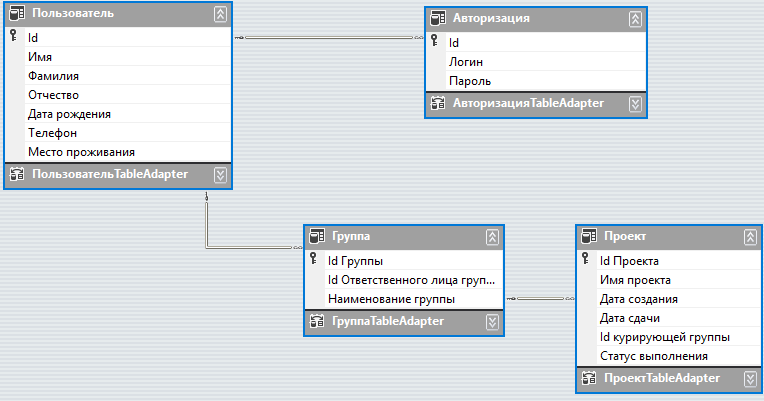


Рисунок 2.6 – Структура связей с БД

Таблица Авторизация для входа в приложения (рис 2.7):

1. Id (PK) - Уникальный идентификатор пользователя,
2. Логин – Имя учётной записи,
3. Пароль – Пароль для входа в учётную запись.

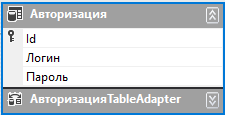


Рисунок 2.7 – Таблица Авторизация

Таблица Группа (рис. 2.8):

1) Id Группы (PK) – Уникальный идентификатор группы,

2) Id Ответственного лица группы – Идентификатор ответственного лица группы,

3) Наименование группы – Определенное название группы.

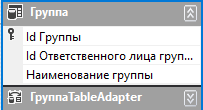


Рисунок 2.8 – Таблица Группа

Таблица Пользователь (рис 2.9):

1. Id – Уникальный идентификатор пользователя,
2. Имя – Имя пользователя,
3. Фамилия – Фамилия пользователя,
4. Отчество – Отчество пользователя,
5. Дата рождения – Дата рождения пользователя,
6. Телефон – Телефон пользователя,
7. Место проживания – Координаты проживания пользователя.

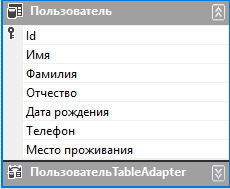


Рисунок 2.9 – Таблица Пользователь

Таблица Проект (рис. 2.10)

1) Id Проекта – Уникальный идентификатор проекта,

2) Имя проекта – Наименование проекта,

3) Дата создания – Дата начала реализации проекта,

4) Дата сдачи – Конечные сроки сдачи проекта,

5) Id курирующей группы – Идентификатор отвечающей за проект группы,

6) Статус выполнения – Описание состояния проекта.

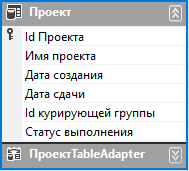


Рисунок 2.10 – Таблица Проект

* 1. **UML Диаграммы.**
     1. Диаграмма Прецедентов

Данная диаграмма показывает возможные взаимодействия пользователя с приложением (рис. 2.11).

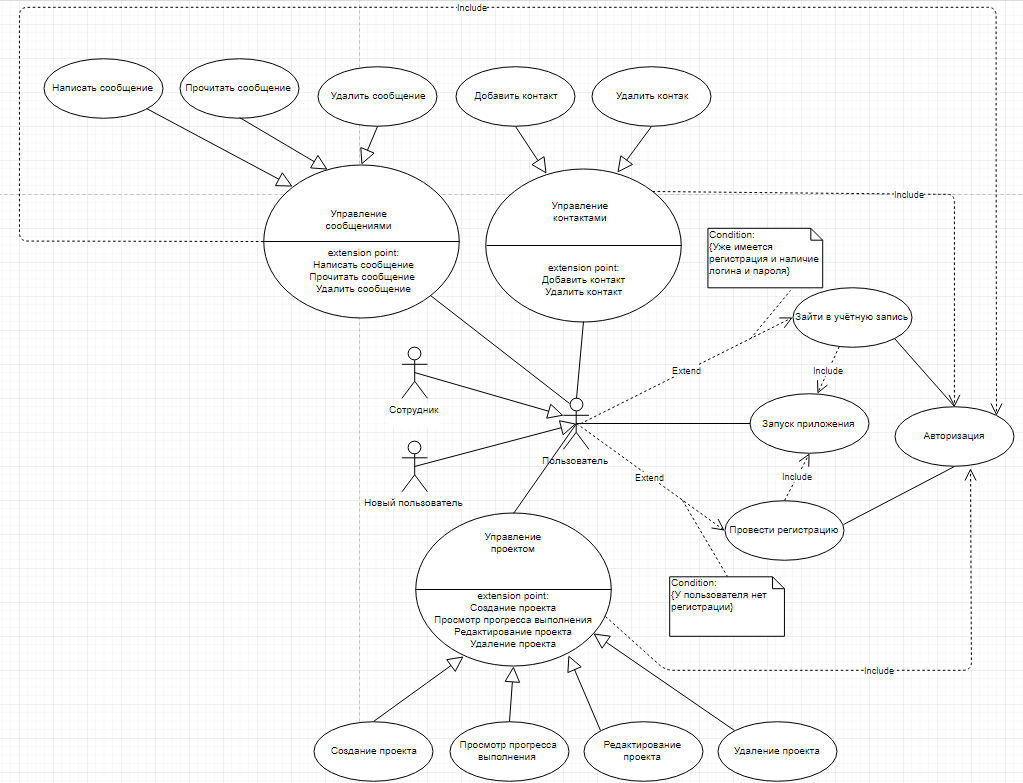


Рисунок 2.11 – Диаграмма Прецедентов

* + 1. Диаграмма Классов

Данная диаграмма показывает статическую структуру моделируемой системы (рис. 2.12).

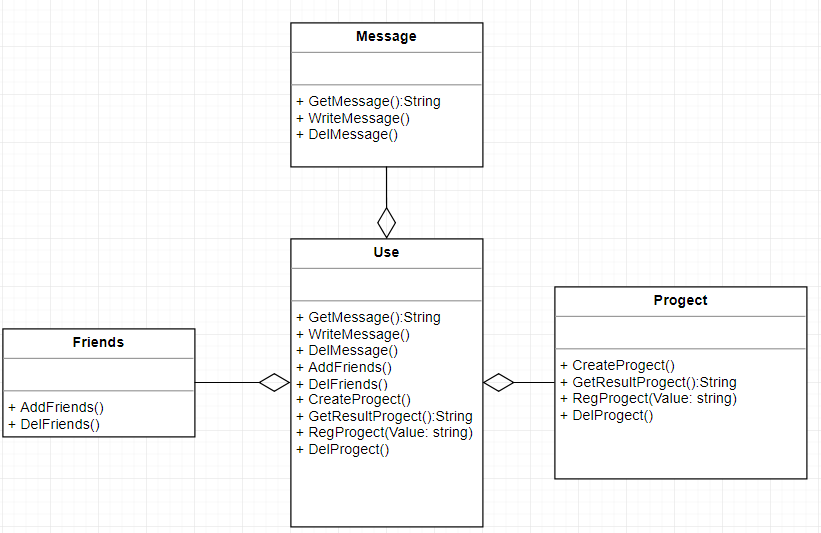


Рисунок 2.12 – Диаграмма Классов

* + 1. Диаграмма Состояний

Диаграмма состояний показывает, как программа переходит из одного положения в другое (рис. 2.13).

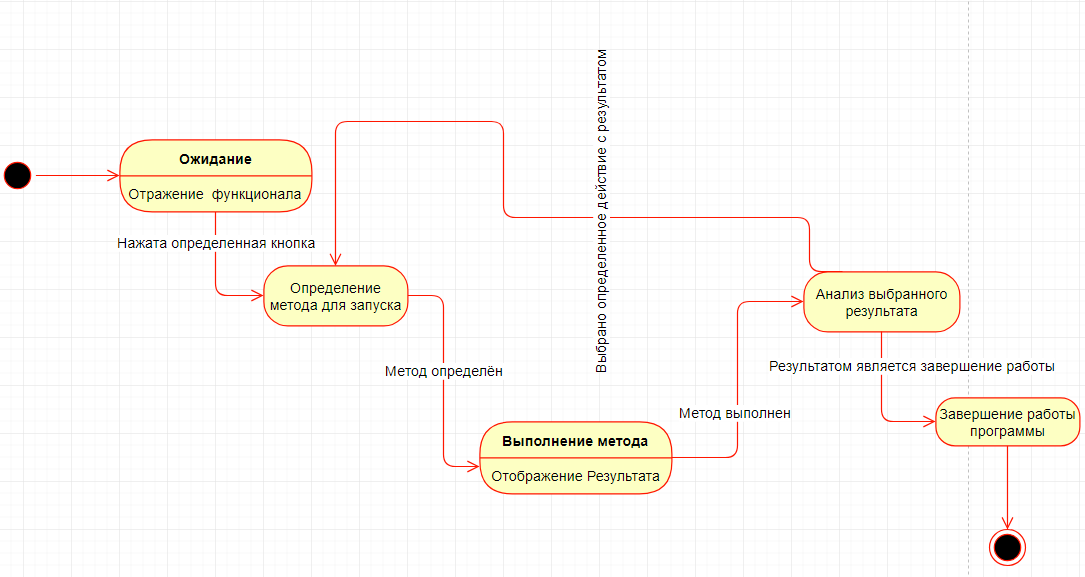


Рисунок 2.13 – Диаграмма Состояний

* + 1. Диаграмма Деятельности

Диаграмма деятельности предоставляет алгоритм действий, выполняемых в системе (рис. 2.14).

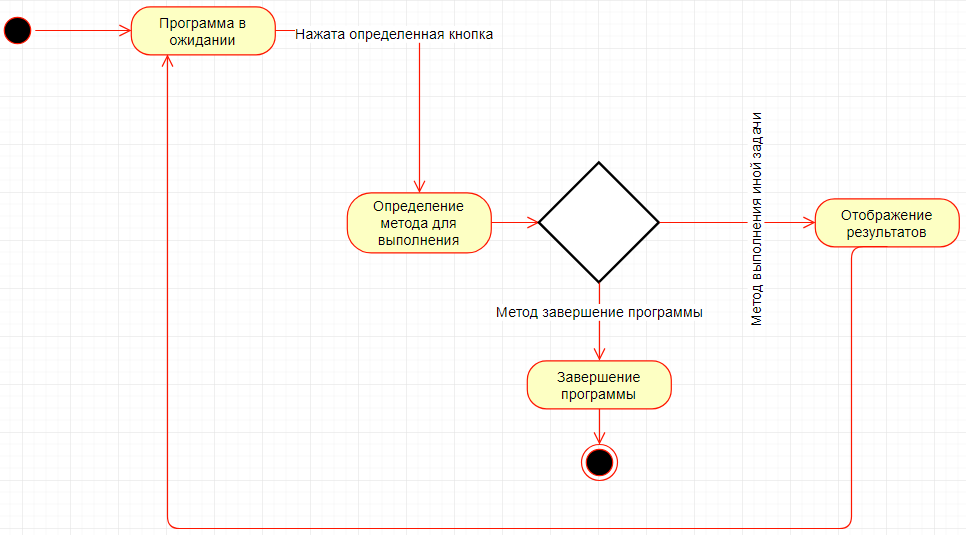


Рисунок 2.14 – Диаграмма Деятельности

* + 1. Диаграмма Последовательности

Диаграмма последовательности показывает взаимодействие объектов в системе (рис. 2.15).

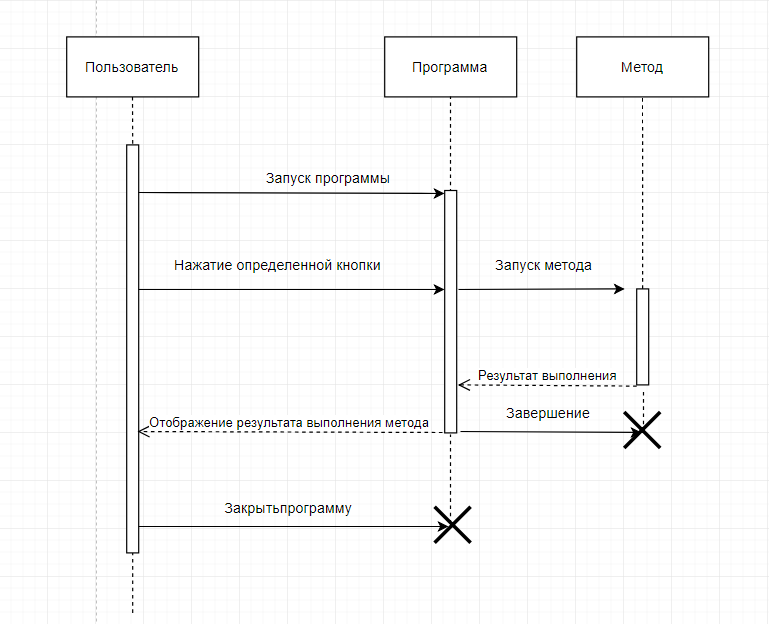


Рисунок 2.15 – Диаграмма Последовательности

* + 1. Диаграмма развёртывания

Диаграмма развёртывания показывает топологию системы (рис. 2.16).

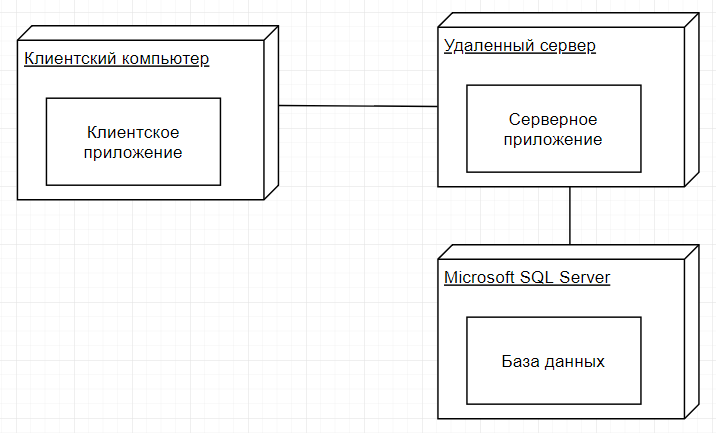


Рисунок 2.16 – Диаграмма Развёртывания

* + 1. IDEF диаграмма

IDEF диаграмма описывает бизнес процессы (рис. 2.17).

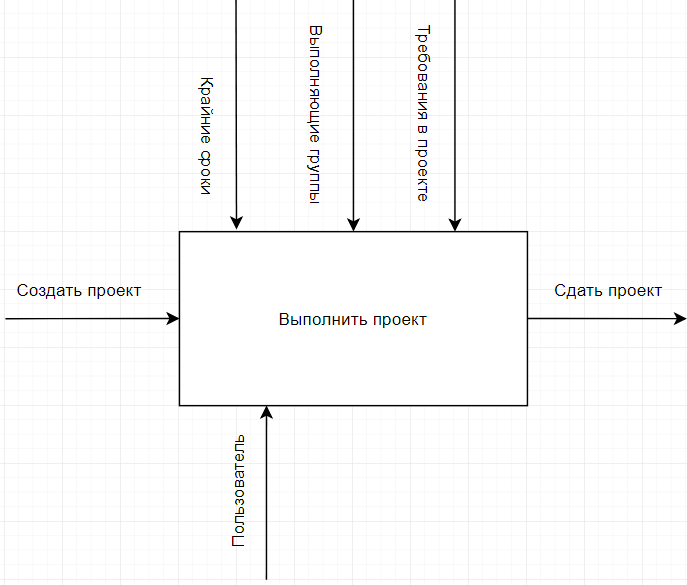


Рисунок 2.17 – IDEF диаграмма

* 1. **Выбор программных и технических средств**

Adobe Photoshop - многофункциональный графический редактор, разработанный и распространяемый фирмой Adobe Systems. В основном работает с растровыми изображениями, однако имеет некоторые векторные инструменты. Продукт является лидером рынка в области коммерческих средств редактирования растровых изображений и наиболее известным продуктом фирм-мы Adobe. В настоящее время Photoshop доступен на платформах macOS, Windows, в мобильных системах iOS, Windows Phone и Android. Также существует версия Photoshop Express для Windows Phone 8 и 8.1. Сейчас в США идёт бета-тестирование стриминговой версии продукта для Chrome OS. Ранние версии редактора были портированы под SGI IRIX, но официальная поддержка была прекращена, начиная с третьей версии продукта. Для версий 8.0 и CS6 возможен запуск под Linux с помощью альтернативы Windows API — Wine.

Несмотря на то, что изначально программа была разработана как редактор изображений для [полиграфии](https://ru.wikipedia.org/wiki/Полиграфия), в данное время она широко используется и в [веб-дизайне](https://ru.wikipedia.org/wiki/Веб-дизайн). Вместе с более ранними версиями Photoshop распространялась специальная программа для этих целей — [Adobe ImageReady](https://ru.wikipedia.org/wiki/Adobe_ImageReady) (для анимации gif-файлов), которая была исключена из поставки Photoshop CS3 за счёт интеграции её функций в основной графический редактор, а также включения в линейку программных продуктов [Adobe Fireworks](https://ru.wikipedia.org/wiki/Adobe_Fireworks), перешедшего в собственность Adobe после приобретения компании [Macromedia](https://ru.wikipedia.org/wiki/Macromedia).

Microsoft Visual Studio – Это интегрированная среда разработки (IDE) от Microsoft. Он используется для разработки компьютерных программ, а также веб-сайтов, веб-приложений, веб-сервисов и мобильных приложений. Visual Studio использует платформы разработки программного обеспечения Microsoft, такие как Windows API , Windows Forms , Windows Presentation Foundation , Windows Store и Microsoft Silverlight . Он может производить как собственный код, так и управляемый код.

Visual Studio включает в себя редактор кода, поддерживающий IntelliSense (компонент завершения кода), а также рефакторинг кода. Интегрированный отладчик работает как отладчик уровня источника и отладчик уровня машины. Другие встроенные инструменты включают профилировщик кода, конструктор форм для создания приложений с графическим интерфейсом, веб-дизайнер, дизайнер классов и конструктор схем базы данных. Он принимает плагины, которые расширяют функциональность практически на каждом уровне, включая добавление поддержки систем контроля версий (таких как Subversion и Git).) и добавление новых наборов инструментов, таких как редакторы и визуальные дизайнеры для языков, специфичных для предметной области, или наборов инструментов для других аспектов жизненного цикла разработки программного обеспечения (например, клиент Team Foundation Server: Team Explorer).

Visual Studio поддерживает 36 различных языков программирования и позволяет редактору кода и отладчику поддерживать (в разной степени) практически любой язык программирования при условии, что существует специальная языковая служба. Встроенные языки включают C, C ++, C ++ / CLI, Visual Basic .NET, C #, F #, XML, XSLT, HTML и CSS. Поддержка других языков, таких как Python, Ruby, Node.js.

1. **ОРГАНИЗАЦИОННЫЙ РАЗДЕЛ**
   1. **Организация графика разработки**

Trello — это одна из самых популярных систем управления проектами в режиме онлайн, которая пользуется особенным спросом среди небольших компаний и стартапов. Она позволяет эффективно организовывать работу по японской методологии канбан-досок.

Она создана Fog Creek Software в 2011 году на базе MongoDB, Backbone.js и Node.js. Главные достоинства, которые позволили Trello добиться популярности — это: Простой интерфейс, бесплатный доступ, интеграция с другими инструментами для онлайн работы.

Для организации задач используется доска с карточками, которые распределяются по типам. Как правило, задачи разбиваются на:

1. Запланированные,
2. Текущие,
3. Выполненные.

Это самая элементарная структура, возможности по модернизации, которой ограничены лишь вашим воображением.

Далее пойдут скриншоты, включающие в себя начало работы (рис. 3.1), ситуацию в процессе (рис. 3.2), конец первой (рис. 3.3), второй (рис. 3.4) частей и финальный этап (рис. 3.5).

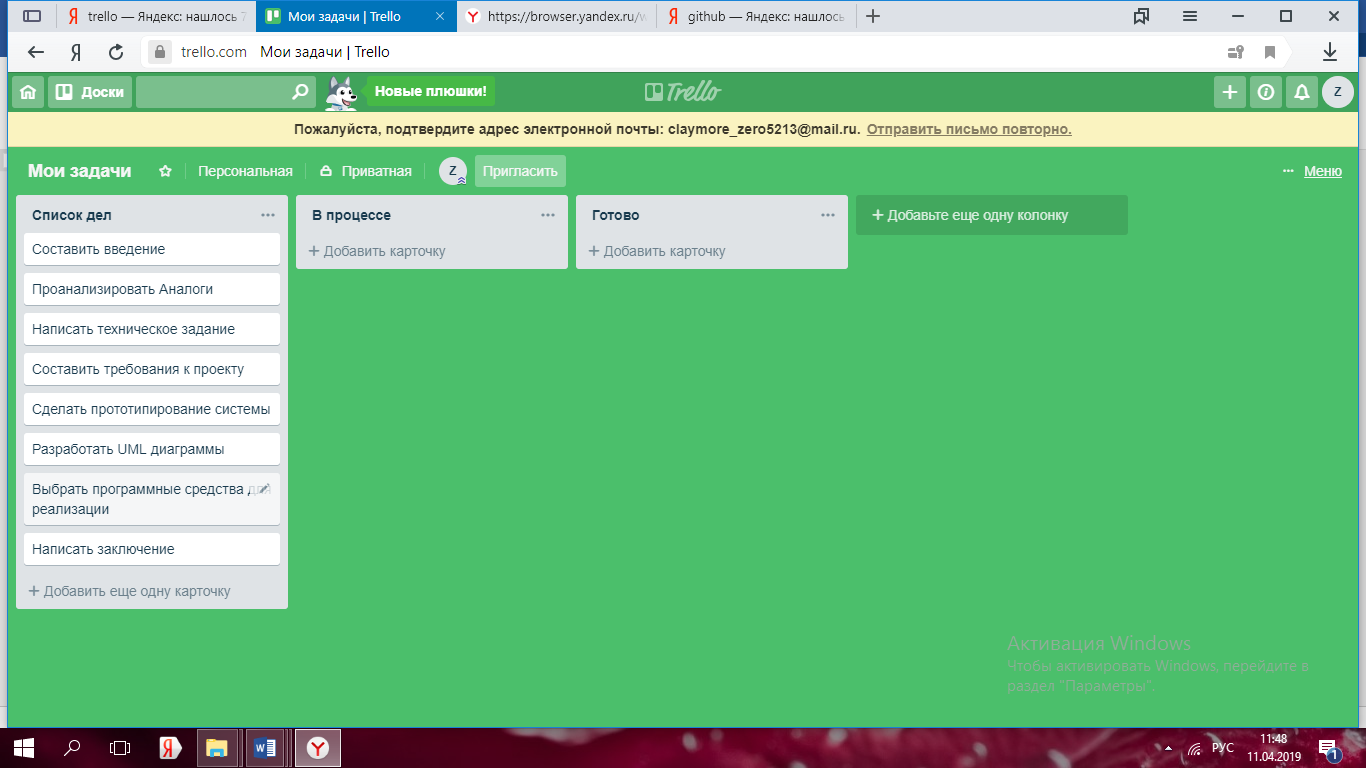


Рисунок 3.1 – Начало проектирование

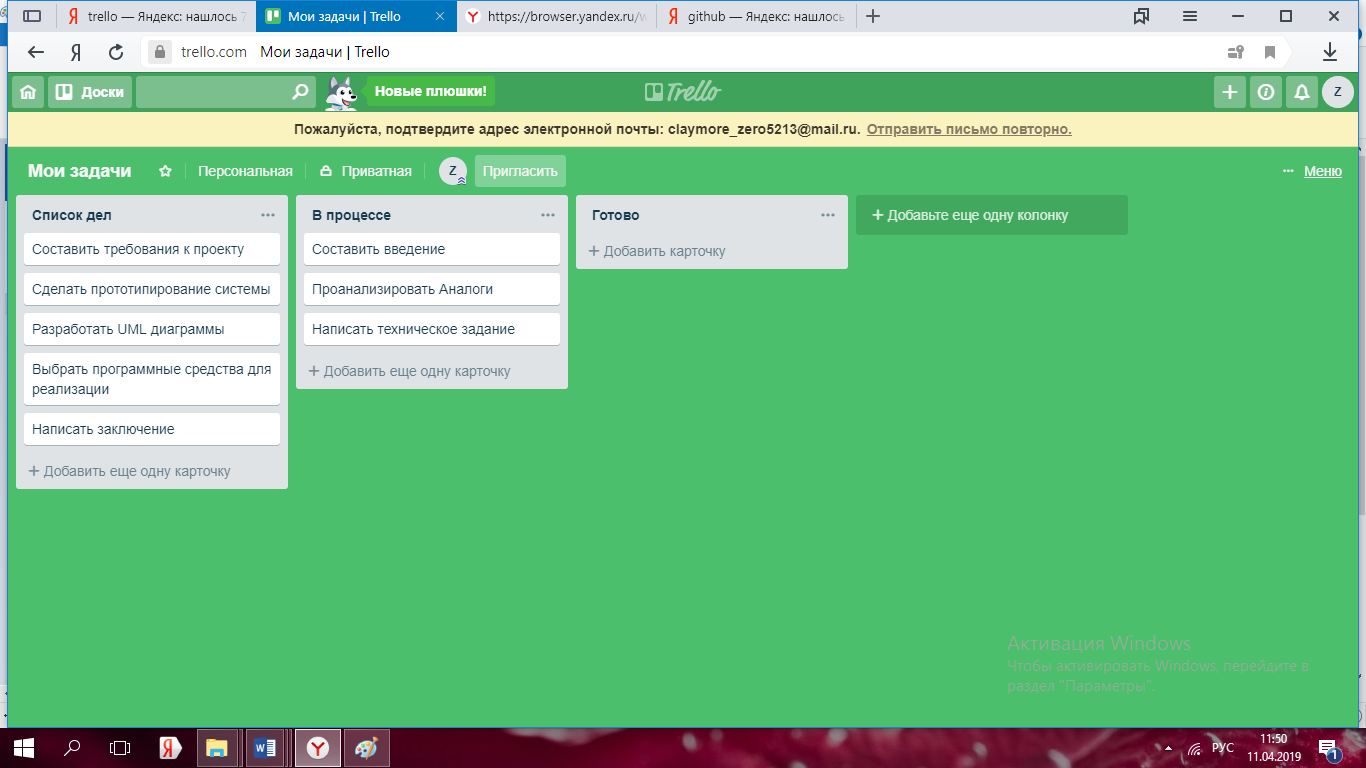


Рисунок 3.2 – В процессе

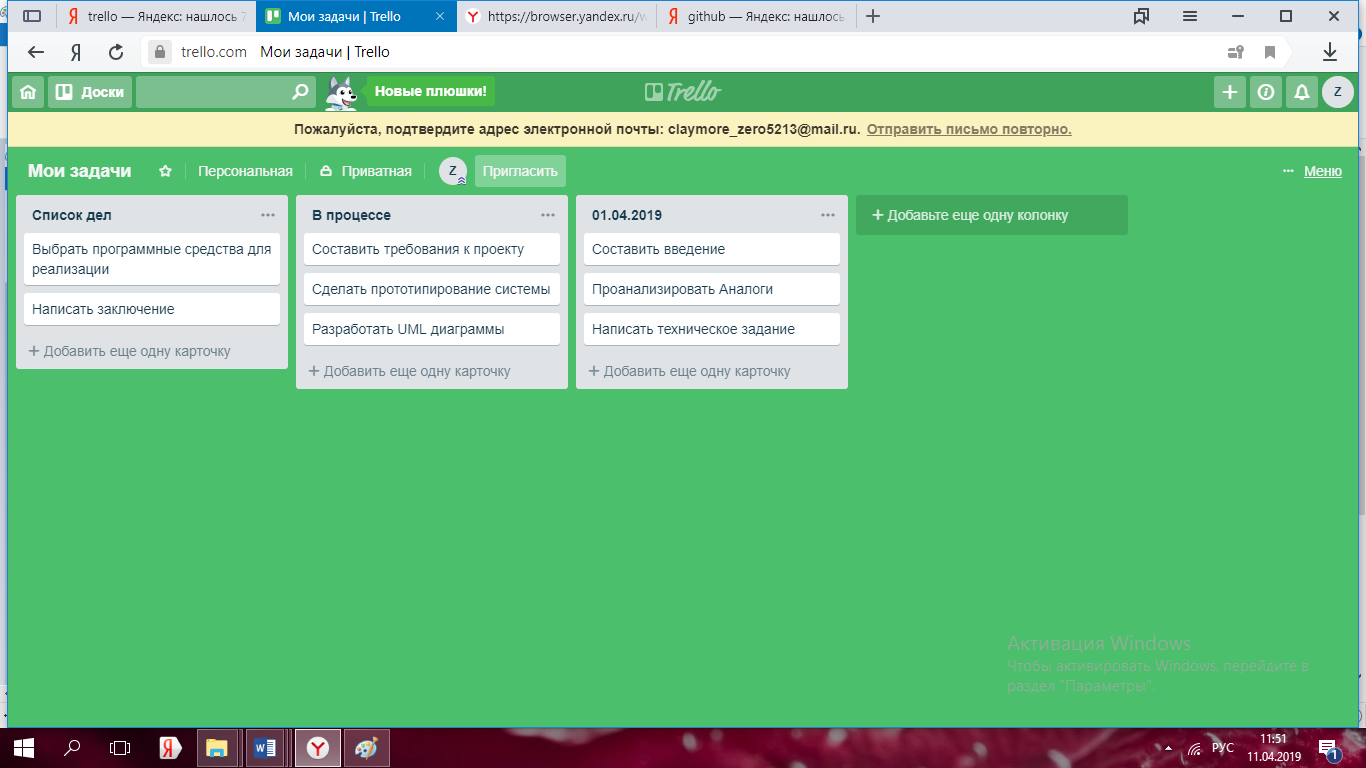


Рисунок 3.3 – Конец первой части

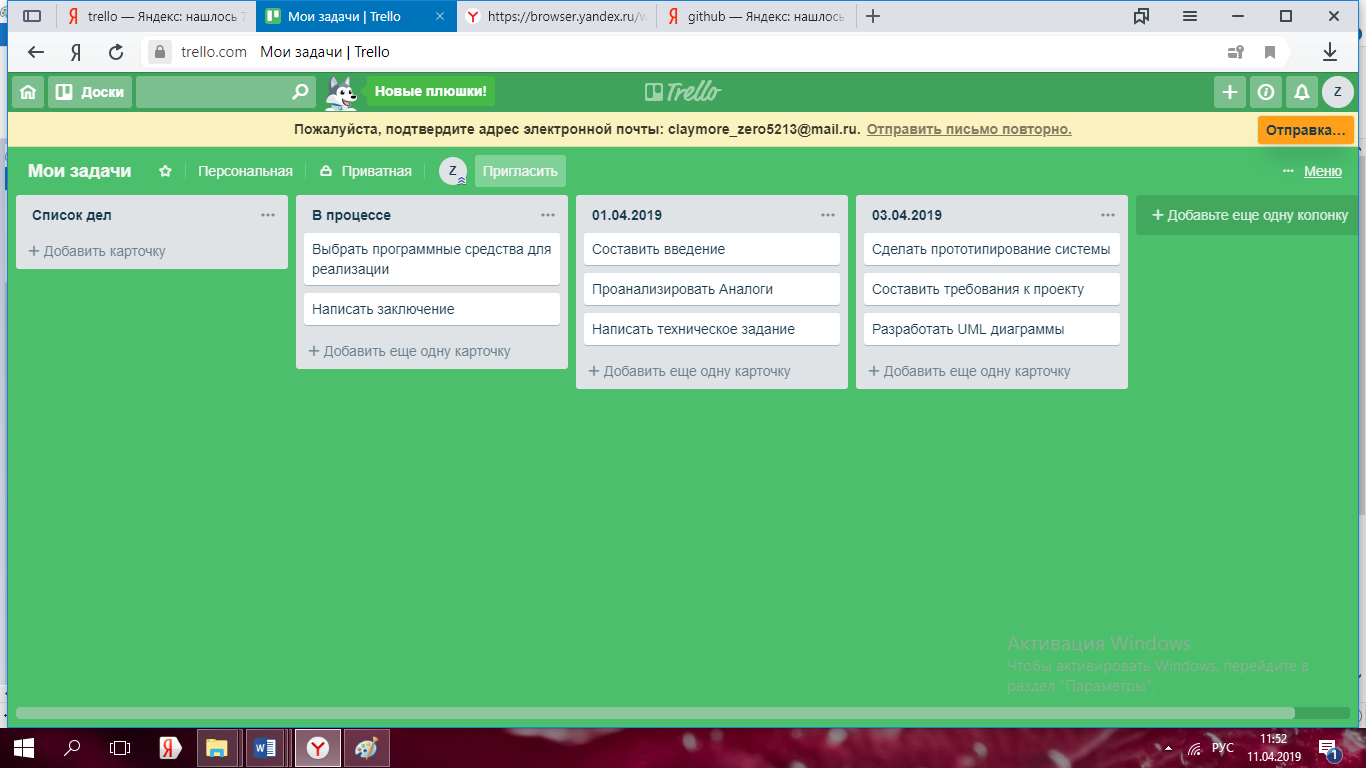


Рисунок 3.4 – Конец второй части

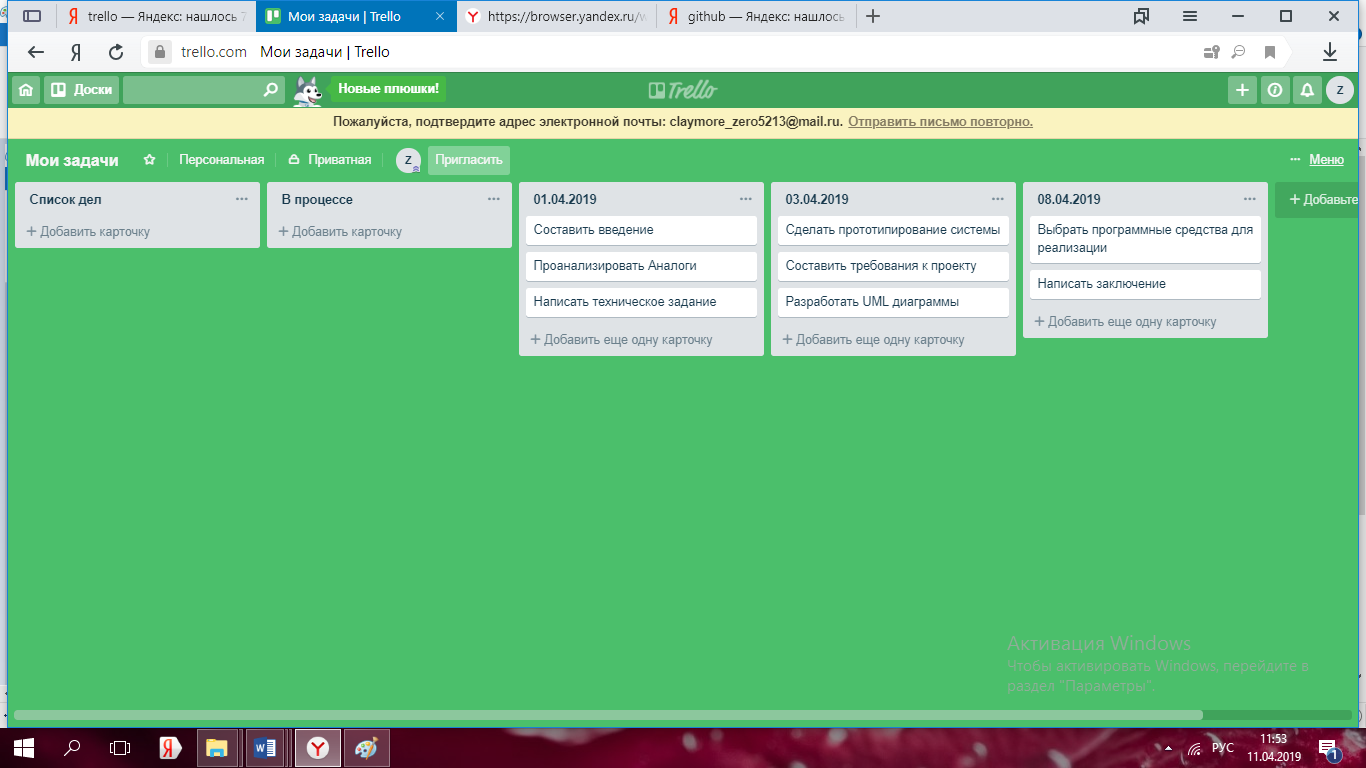


Рисунок 3.5 – Завершение работы

**ЗАКЛЮЧЕНИЕ**

В курсовом проекте был разработан интернет магазин с возможностями заказов товаров, регистрации и авторизации пользователей, просмотров отзывов, оставленных предыдущими клиентами.

По сравнению с другими сайтами-аналогами, данный сайт обладает своей уникальностью – внешним видом. Хоть он и не идеален, однако это не мешает ему бросаться в глаза даже больше других, тем самым привлекая внимание пользователей, которые в будущем могут стать клиентами сайта.

В процессе работы над проектом были получены практические навыки в исследовании предметной области, описания проектного решения, построения моделей.

Выполнение задач оценивается положительно, однако не все задачи были выполнены так, как было запланировано из-за меняющихся на протяжении работы условий.

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**

1. ГОСТ 2.004-88 ЕСКД. Общие требования к выполнению конструкторских и технологических документов на печатающих устройствах вывода ЭВМ;
2. Сайт компании Bic [Электронный ресурс] / Режим доступа:

https://www.bicworld.com/en. Дата обращения – (04.06.2018).

1. Сайт компании Derwent [Электронный ресурс] / Режим Доступа:

https://www.derwentart.com/en/gb/home Дата обращения - (04.06.2018).

1. Сайт компании Erich Krause [Электронный ресурс] / Режим доступа:

http://www.erichkrause.com/. Дата обращения – (03.06.2018).

1. Сайт компании Pilot [Электронный ресурс] / Режим доступа:

http://www.pilot.co.jp/. Дата обращения – (01.06.2018).

**ПРИЛОЖЕНИЕ**