**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики и информатики**

**Кафедра информационных систем управления**

**Отчет**

**По проектированию программных систем**

Выполнил студент группы № 12

*Шишлянников Иван Викторович*

**Минск 2020**

Упражнение 1.2. Создать проект и распределить задачи

1. Создать проект типа Kanban в выбранной системе управления проектами

согласно:

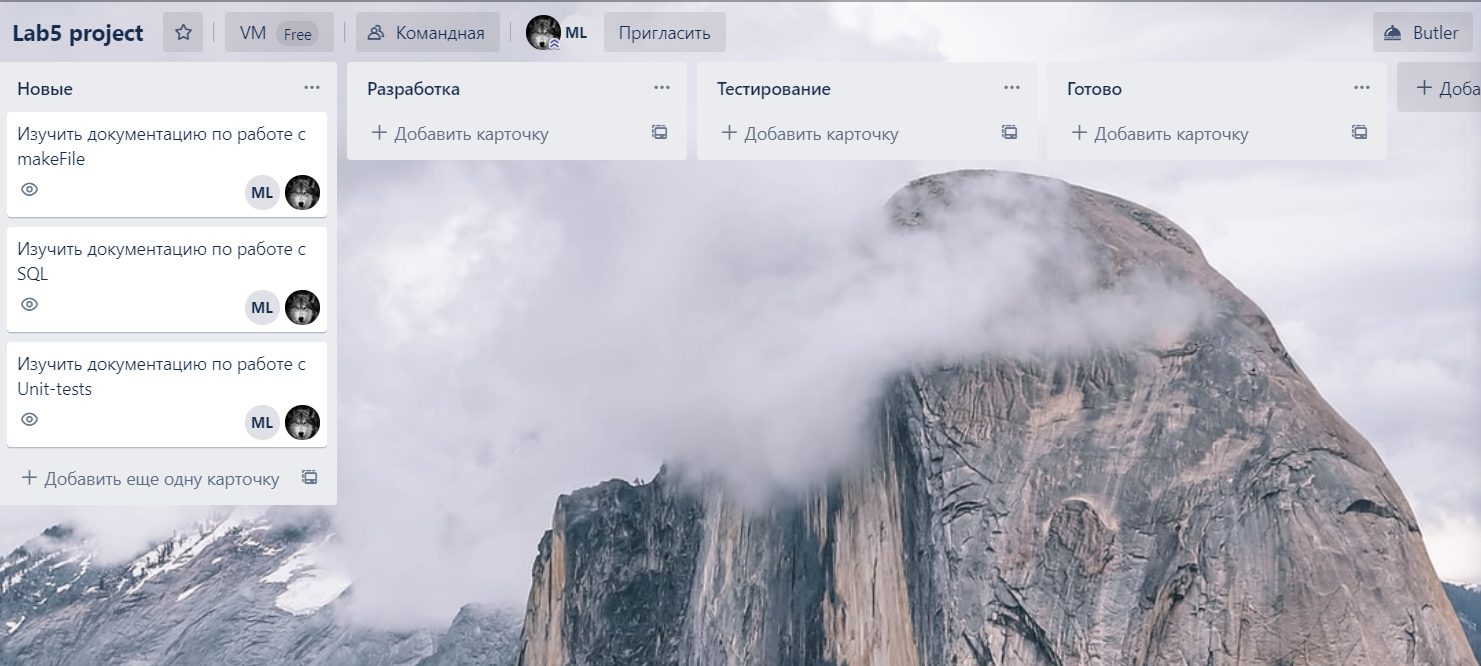
◦ Порядок действий при создании проекта в Trello

2. Добавить в проект участников.

3. Добавить списки и задачи.

4. Распределить задачи.

5. Использовать проект для управления командой.



ЗАДАНИЕ 2. ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ

1. Познакомиться с документом UML and C.pdf.

2. Реализовать диаграмму вариантов использования для иллюстрации

функциональных требований согласно варианту команды.

3. Используя диаграмму классов, создать диаграмму файлов консольного

приложения, иллюстрирующую файлы, типы, атрибуты, функции и отношения.

Для разработки использовать десктопные приложения:

• VisualParadigm Community Edition (http://visual-paradigm.com/)

• StarUML 2 (http://staruml.io/)

или онлайн-сервисы:

• Creately (http://creately.com/)

• GenMyModel (<http://www.genmymodel.com/>)

ЗАДАНИЕ 3. ПРОЕКТИРОВАНИЕ АРХИТЕКТУРЫ БАЗЫ

ДАННЫХ И НАПОЛНЕНИЕ ДАННЫМИ

Для решения необходимо использовать http://www.vertabelo.com для

проектирования архитектуры базы данных.

Упражнение 3.1. Изучить функционал Vertabelo

Ознакомьтесь с видео https://youtu.be/hU-A08K08-Y.

Упражнение 3.2. Спроектировать базу данных

Спроектируйте базу данных для хранения данных о пользователях и других

7

сущностях согласно варианту задания.

База данных должна содержать таблицу пользователей и таблицы под каждый

тип сущности приложения.

Например, для банковского приложения создаются таблицы:

BANK\_ACCOUNTS, BANK\_CLIENTS, BANK\_USERS, BANK\_CONFIG. В банковскогом

приложении сведения о счетах всех пользователей хранятся в таблице

BANK\_ACCOUNTS. Данные клиентов хранятся в таблице BANK\_CLIENTS. Данные о

пользователях, имеющих доступ в приложение, — в таблице BANK\_USERS.

Основная конфигурационная информация банка должна храниться в таблице

BANK\_CONFIG. К конфигурационной информации относится:

• размер процентов для того или иного типа счет;

• период начислений процентов;

• сведения о штрафе в случае овердрафт;

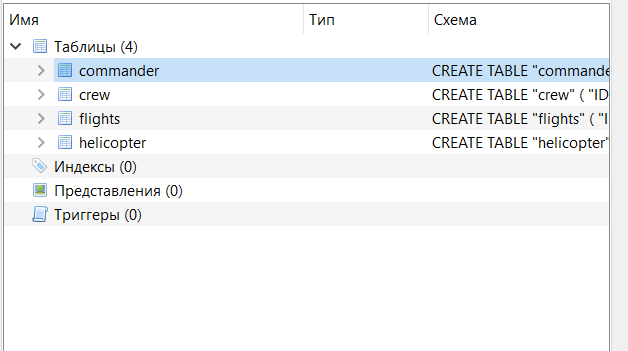
• максимальное количество транзакций для CheckingAccount;

• сроки для OverdraftAccount.

Упражнение 3.3. Сгенерировать sql-скрипт для sqlite

Сгенерируйте файл SQL-скрипта для БД SQLite. Импортируйте файл в

Valentina Studio или SQLBrowser и заполните таблицы БД данными.



ЗАДАНИЕ 4. НЕПРЕРЫВНАЯ СБОРКА ПРОЕКТА И

ТЕСТИРОВАНИЕ

1. Изучите основные материалы по использованию travis-ci для сборки и

тестирования консольного проекта.

2. Продемонстрируйте результаты сборки и тестов (не менее 3 тестов на

каждый файл проекта).

ЗАДАНИЕ 5. РАЗРАБОТКА ПРИЛОЖЕНИЯ

Упражнение 5.1. Подключить репозиторий группы

Подключить репозиторий на github согласно ссылке для Вашей группы.

В процессе выполнения разработки проиллюстрировать работу с несколькими

ветками в репозитории проекта, добавление и приём коммитов, слияние и

перемещение веток и т. д.

Упражнение 5.2. Документирование проекта

Изучить документацию https://guides.github.com/features/wikis/ и

документировать проект в Readme и wiki репозитория согласно следующим

требованиям:

1. Файл Readme и wiki оформить с помощью синтаксиса Markdown.

2. Структура файла Readme должна быть следующей:

◦ Project Name: в данном блоке указать название проекта.

◦ Description: Краткое описание проекта и его функциональности в 3-5

предложениях.

◦ Installation: Последовательность шагов, как установить приложение

локально.

8

◦ Usage: Рекомендации как использовать приложение после установки.

Может содержать скриншоты.

◦ Contributing: Сведения об авторах проекта и какие задачи реализовывали.

3. Структура страниц wiki должна быть следующей:

◦ Главная страница: содержит краткое описание задачи и ссылки на другие

материалы и разделы.

◦ Функциональные требования: описание функциональных требований,

диаграммы Use case, текстовые сценарии.

◦ Диаграмма файлов приложения: диаграмма файлов и описание.

◦ Дополнительная спецификация: ограничения, требования к

безопасности, надежности и другое.

◦ Схема базы данных: страница содержит схему базы данных в виде

изображения и ссылку на sql-файл.

◦ Презентация проекта: ссылка на презентацию проекта, в которой должно

быть отражено распределение задач в команде, требования к приложению,

схема базы данных, как была организовано работа с репозиторием и

проектом и др.

Упражнение 5.3. Разработать приложение

Разработать консольное приложение на языке Си, которое позволяет пройти

аутентификацию и авторизацию и выполнять операции согласно варианту задания.

Проект должен состоять из нескольких .c файлов. Структура проекта должна

соответствовать модели КИС, содержать папки bin, build, includes, src.

Документацию проекта, включая постановку задачи и другие материалы

представить в вики проекта.

Проиллюстрировать различные виды запросов к базе данных, обеспечить

хранение данных в различных форматах, включая изображения.