Регламент рабочих процессов проекта.

# Используемые инструменты и технологии.

1. ***Kanban Board*** – система отчетности и багтрэкинга (*bugtracking*).
2. ***Git*** – (VCS) система контроля версий.
3. ***GitHub*** – хостинг репозитория.
4. ***Skype*** – средство организации митингов (*meeting*).
5. ***Электронная почта*** – средство общения менеджера и разработчиков в рамках решения рабочих вопросов проекта.

# Работа с Kanban Board.

***Kanban Board*** – инструмент, отражающий актуальную информацию об активностях каждого члена команды.

1. Разделения ***Kanban Board***:
   1. *Open* – пулл предстоящих активностей. Активность на этой части доски может быть связана или не связана с конкретным разработчиком.
   2. *Planned* – пулл ближайших активностей. Главное отличие от предыдущей части доски – наличие предполагаемой даты начала и конца (данные даты не обязуют разработчика выполнить задачу строго в указанный период, но предоставляют менеджеру примерное представление о том, когда задача будет выполняться).
   3. *In progress* – пулл активностей, которые находятся в разработке в данный период времени (не более двух-трех одновременно).
   4. *In testing* – пулл активностей, которые находятся на ревью (*review*).
   5. *Done* – пулл завершенных активностей.
2. Менеджер создает пулл активностей (в соответствии с обсуждаемыми задачами на паре во вторник) и разделяет его между членами команды.
   1. Членам команды рекомендуется самим добавлять свои активности на ***Kanban Board*** для облегчения работы менеджеру и собственного понимания своих актуальных задач.
3. Разработчик должен поддерживать актуальную информацию на своей доске и размещать активности в соответствие с данным регламентом.
4. ***Kanban Board*** также позволяет эффективно взаимодействовать с почтой, поэтому, если вам необходимо оповестить кого-то об изменении какой-либо активности, рекомендуется использовать данный сервис.
5. ***Kanban Board*** также позволяет эффективную работу с документами (к активности можно приложить документ, который будет доступен членам проекта, а также сразу оповестить о приложенном документе, или прикрепить ссылку на репозиторий). Настоятельная рекомендация пользоваться данными сервисами при подобной необходимости (это не отменяет необходимости заливать документ в репозиторий проекта).

# Общение внутри команды.

1. Для общения внутри команды в рамках рабочих процессов разработчики используют ***Kanban Board*** и ***Электронную почту*** (имеется в виду отчётность разработчиков в рамках активностей перед менеджером проекта и другими членами команды. Средства для общения личного характера, пусть даже в рамках проекта, выбираются разработчиком).
2. Регулярные митинги (сбор команды для устного отчета о своих продвижениях, вопросов по работе и т.п.) проводятся 2 раза в неделю в ***skype*** в согласованное менеджером и командой время.

Заполнение дефектов проекта

# Используемые инструменты и технологии

1. ***Kanban Board*** – система отчетности и багтрэкинга (*bugtracking*).
2. ***Git*** – (VCS) система контроля версий.
3. ***Электронная почта*** – нормальной практикой считается дублирование комментариев и изменений дефектов посредством уведомлений на e-mail

Примечание: ***Git*** и его инструменты будут в основном использоваться для commit-ов и ревью, для регистрации дефектов будет использоваться система ***Kanban Board***

**Оформление дефектов**

***Баг репорт*** - это технический документ и в связи с этим акцентирую внимание, что язык описания проблемы должен быть техническим. Должна использоваться правильная терминология при использовании названий элементов пользовательского интерфейса действий пользователя и полученных результатах.

**Тема бага**

Заполнение бага начинается с формирование темы. Тема должна быть описана в таком виде [где возникла проблема] Описание проблемы

***Пример:***

[editbox “Name”] Поле недоступно для заполнения

**Описание бага**

Описание бага состоит из ***трех частей***:

* описание
* последовательность шагов, которая привела к ошибке
* скриншоты (скриншоты обязательны!)

*Примечание:* Скриншоты не нужны лишь в том случае, если вы создали баг формальным, а по сути это является запросом.

В описание также в обязательном порядке указываем все версии программ, под которыми вы работаете. Указываете название и версию VS если это необходимо или название программ и компиляторов, которые вы используете.

**Шаги**

Очень важно четко описать все шаги, с упоминаем всех вводимых данных (имени пользователя, данных для заполнения формы) и промежуточных результатов.

***Например:***

Шаги к воспроизведению  
1. Войдите в системы: Пользователь Тестер1, пароль xxxXXX  
--> Вход в систему осуществлен  
2. Кликните линк Профайл  
--> Страница Профайл открылась  
3. Введите Новое имя пользователя: Тестер2  
4. Нажмите кнопку Сохранить

***Фактический результат:*** На экране появилась ошибка. Новое имя пользователя не было сохранено.  
***Ожидаемый результат:*** Страница профайл перегрузилась. Новое значение имени пользователя сохранено.

Далее мы прикрепляем нужные **скриншоты** так, чтобы на них было всё видно и понятно разработчику. В случае необходимости подчеркиваем или выделяем нужную область на скриншоте.

Итак, мы заполнили тему, описание и приложили скриншоты. После этого определяем насколько серьезный баг и в течении какого времени его лучше устранить.

**Приоритет** (выставляется в Kanban Board в графе ‘’Серьезность”)

Рассмотрим 5 видов приоритета для установления его на баг :

1.***Блокирует*** – это значит самый высокий приоритет у бага, ставится в том случае если ваша работа, тестирование или иные действия блокируются этим багом.

2.***Критично*** – высокий уровень на рассмотрение и решение бага, также означает что найденный баг является критичным для системы и должен быть взят на разработку в ближайшее время

3.***Средняя*** – найденный баг не мешает работе, может быть взят на разработку через определенное время и впоследствии исправлен

4.***Незначительно*** – найденный баг является недоработкой

5.***Улучшение*** – баг или какое-то замечание по поводу работоспособности, скорости обработки данных, интерфейса.

**Основные ошибки при написании баг репортов**

**Недостаточность предоставленных данных**   
Не всегда одна и таже проблема проявляется при всех вводимых значениях и под любым вошедшим в систему пользователем, поэтому настоятельно рекомендуется вносить все необходимые данные в баг репорт

**Определение серьезности**  
Очень часто происходит либо завышение, либо занижение серьезности дефекта, что может привести к неправильной очередности при решении проблемы.

*Примечание:*

Частой ошибкой является, что вы заводите баг, не убедившись, что это проблема связана с ошибкой разработчика, а не с вашими ошибочными действиями.