**План разработки**

1. Технологический процесс

Процесс разработки состоит из следующих основных этапов:

1. Определение технологического стека
2. Разработка общей архитектуры
3. Формирование технического задания
4. Декомпозиция на отдельные задачи
5. Составление календарного плана выполнения задач
6. Выделение подзадач и определение сроков их выполнения
7. Определение исполнителей
8. Выполнение выделеных подзадач
9. Верификация результатов
10. Повторение шагов 6-9, пока проект не будет приведен в соответствие с техническим заданием

Все участники команды в равной степени принимают участие в разработке на всех ее этапах. Роли распределяются в соответствии со знаниями и умениями каждого учатника, а также исходя из соображений рациональности расхода ресурсов (в частности, времени). При возникноверии спорных ситуаций решающее слово остается за управляющим проектом.

1. процесс разработки

Задачи может создавать любой участник команды. При постановке задачи создается новая карточка в Trello в списке To Do. В названии карточки необходимо кратко описать задачу. В поле Description изложено более подробное описание задачи(если это требуется). Назначается крайний срок, до наступления которого задача должна быть выполнена. Одновременно заводится аналогичный Isuue в GiHub. Issue необходимо привязать к созданной в Trello карточке используя Addon;

Если у задачи из списка To Do отсутствует исполнитель, любой участник команды может указать себя в качестве исполнителя задачи. Если задача долго находится в списке To Do без указанного исполнителя - менеджер проекта самостоятельно назначает исполнителя;

Когда исполнитель начинает работать над задачей, он должен перенести ее из списка To Do в список In Progress. В своем fork’е основного репозитория на аккаунте GitHub необходимо сделать отдельную ветку(от ветки master) со следующим названием: “Issue[Номер Issue на основном репозитории на GitHub]” (например: Issue1);

Когда исполнитель задачи закончил работу над задачей, он помещает ее в список Testing. В сервисе GitHub необходимо сделать Pull Request своей ветки(напр.: Issue1) в ветку master основного репозитория;

Группа верификации и остальные разработчики должны выполнить верификацию кода и написать комментарии, если были обнаружены ошибки. Для исправления мелких ошибок исполнитель может обновлять свой Pull Request и отвечать на соответствующие комментарии(напр.: Исправлено). При наличии существенных ошибок или недоработок, карточка задачи переносится обратно в список In Progress. После исправления, разработчик обновляет свой Pull Request и переводит карточку задачи обратно в список Testing;

Если у группы верификации нет претензий или комментарий к Pull Request’у, то они самостоятельно переводят карточку задачи в список Done и выполняют Merge Pull Request’а.

1. Коммуникация

Для коммуникации между участниками команды может использоваться любой доступный способ связи. Основным считается общий чат вконтакте. Вся информация, имеющая значение не только в контексте одной задачи, должна передаваться через него. Обязательным условием является возможность каждого участника команды связаться с любым другим участником по телефону.

Общее обсуждение возникающих проблем, решенных задач, полученных результатов, возникших новых задач, сроков их выполнения, исполнителей и т.п. должно происходить в общем чате каждый четверг в 20.00. Для понимания общей картины каждому участнику обязательно отчитыватья о результате работы (даже если он нулевой или отрицательный).

Во избежание трудностей поиска важной информации не рекомендуется использование общего чата для обсуждения вопросов, не связанных с данным проектом, или, во всяком случае, с учебой.

Передача результата для верификации осуществляется через репозиторий на GitHub, в котором должна храниться актуальная версия.