# Техническое задание на разработку Web-проекта.

# Введение

## Цель

Целью проекта является предоставление конечному пользователю удобного и современного сервиса для организации и проведения мероприятий, в том числе удаленных встреч через Интернет средствами аудио и видео связи.

## Сфера применения

* Организация и проведение удаленных собраний внутри компании или между компаниями
* Организация и проведение различных образовательных курсов, лекций
* Организация и проведение удаленного обучения на всех уровнях образования (школа, университет)
* Организация различных мероприятий
* И т.д.

## Основные возможности системы

* Визуализация предмета встречи, возможность наглядного изложения материала.
* Аудио и видео связь.
* Удобная организация встреч и наличие личного календаря для каждого пользователя.

# Общее описание

## Основные объекты системы

* Пользователи
  + Доступ к объектам и возможностям системы имеют только зарегистрированные пользователи
  + Наличие личного календаря
  + Возможность отправки сообщения одному или нескольким другим пользователям
* Группы
  + Создаются пользователями
  + Могут быть открытыми или закрытыми (тип определяется способом попадания пользователя в группу)
  + Внутри группы для всех ее участников могут быть доступны основные элементы группы[[1]](#footnote-1):
    - Файлы
    - Информация о проведенных встречах
    - Сообщения
    - Информация о предстоящих встречах (расписание)
  + Внутри группы пользователь может[[2]](#footnote-2):
    - Добавить, просмотреть или скачать файлы
    - Создать новую встречу
    - Участвовать в опросе на организацию новой встречи
    - Участвовать в удаленных аудио или видео встречах
    - Обмениваться сообщениями с другими участниками группы
    - Редактировать расписание встреч
* Встречи
  + Существуют только в рамках групп
  + Создаются пользователями
  + Состоят из нескольких этапов (каждый этап рассматривается как отдельный объект системы):
    - Организация
    - Проведение
    - Подведение итогов
  + Проводятся в виде:
    - Удаленная встреча средствами аудио и видео связи
    - Реальная встреча в назначенном месте. В таком случае:
  + Для каждой встречи можно определить основные элементы:
    - Название
    - Дата и время
    - Место (в случае реальном встречи)
    - Цели
    - Условия проведения
    - Дополнительные комментарии
  + Информация о встрече, ее основных элементах и итогах сохраняются в группе с возможностью просмотра и редактирования
* Организация встреч
  + Состоит из основных этапов:
    - Определение основных элементов встреч
    - Подготовка доски (опционально)
    - Настройка уровня доступа к возможности изменения элементов встречи другими участниками группы
    - Подготовка материалов для проведения встречи (подробнее в пункте «проведение встречи»)
    - Автоматическое оповещение пользователей о предстоящей встрече
    - Голосование пользователей за время встречи (опционально)
  + Назначение времени и даты встречи может осуществляться посредством голосования всеми участниками за те или иные варианты, предложенные создателем встречи. В таком случае система должна указывать оптимальную[[3]](#footnote-3) дату и время для всех участников, но окончательное решение должен принимать создателей встречи.
* Проведение встречи (в случае удаленной аудио или видео связи)
  + Основные элементы встречи:
    - Участники
    - «Доска» (служит для отображения различного материала, рассматривается как отдельный объект системы)
  + Во время встречи у участников должен оставаться доступ к основным возможностям и элементам системы (кроме присоединения к другой встрече)
* Подведение итогов
  + Выражается в виде заметок о проведенной встрече
  + Помечаются цели, которые были достигнуты в ходе встречи
  + Доступ к созданию и редактированию итогов определяется создателем встречи
* «Доска»
  + Выражается в виде элемента интерфейса, схожим по внешнему виду с маркерной доской или любым иным объектом, предоставляющий общий доступ к графической информации (объявление, рекламный плакат и т.д.)
  + Служит для отображения любой графической информации:
    - Изображения
    - Нарисованные пользователем в реальном времени рисунки
    - Текст
    - Другие файлы в виде ссылок
  + Может состоять из многих страниц
  + Доступ к добавлению и изменению элементов на «доске» определяется создателем встречи
  + Создатель встречи может заранее добавить элементы на «доску»

# Дополнительные объекты и возможности

## Система рекомендаций

**Цель**

Целью разрабатываемой системы является предоставление информации о группах, которые потенциально интересны пользователю.

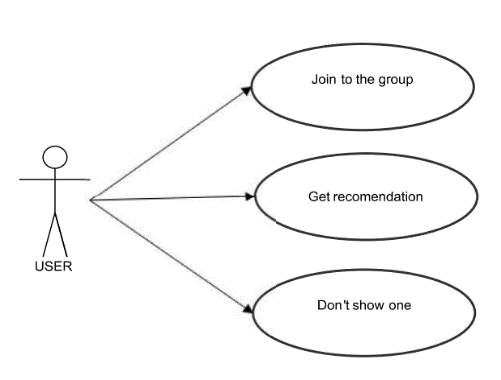
**Функциональные требования.**

В качестве рекомендаций отображаются только открытые группы.

Пользователь должен получать информацию (предложение вступить) о группах, которые соответствуют его потенциальным интересам. Пользователь получает список групп, в порядке возрастания “интересности”.

Может перейти на страницу группы

Может отменить показ данной группы (кнопка “больше не показывать”)

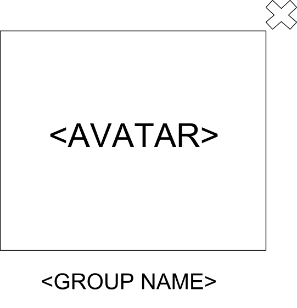


**Параметры реализации.**

Пользователю должна предоставляться 1 группа для рекомендаций, получаемая из списка рекомендации.

Визуальное отображение должно содержать следующие элементы:

* Название группы
* Аватар
* Кнопку “больше не показывать”



**Принцип реализации:**

Реализация возможна на основе [Apache Mahout](http://mahout.apache.org/). Библиотеки машинного обучения.

Рекомендательная система нуждается во входных данных на которых будут основываться рекомендации. Эти данные принимают форму предпочтений на языке Mahout.

Предпочтение состоит из ID пользователя и ID предмета, и обычно числа, выражающего степень предпочтения этого пользователя этому предмету (оценку).

В данном случае оценка представляет собой степень “причастности” пользователя к группе. (Или же просто отражаем участие)

В случае реализации оценок метрикой, для определения правдоподобия является логарифмическое правдоподобие.

В качестве метрики для определения степени подобия пользователя используется коэффицент Танимото.

*Рекомендации основываем на сходстве пользователей. (*[*Метод k ближайших соседей*](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_k_%D0%B1%D0%BB%D0%B8%D0%B6%D0%B0%D0%B9%D1%88%D0%B8%D1%85_%D1%81%D0%BE%D1%81%D0%B5%D0%B4%D0%B5%D0%B9)*)*

**Критерии, по которым можем производить сравнение:**

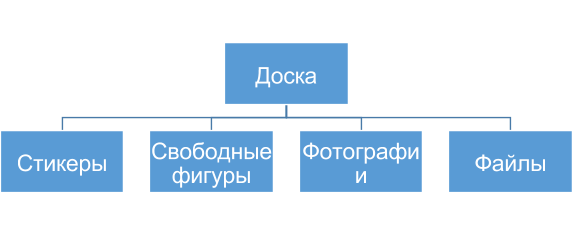
* *Участие в группах пользователей, которые состоят в группах пользователя*
* *Соответствие по тегам*
* *Размеру группы (большие группы имеют больший вес)*
* *Плотности контента (частота проведения митинга, временной ряд по экспоненте)*
* *Возможно коммерциализации*

## Todo list/meeting notes for board

**Цель**

Предоставить пользователям удобный способ подведения итогов встречи.

**Структура доски:**



**Требования**

**TODO list:**

Достаточно дать администратору возможность перед началом встречи дать возможность создать данный список. Вероятно, он будет показываться под самой доской на протяжении всей встречи.

**Meeting notes:**

Необходимо предоставить возможность администратору выбрать наиболее важные элементы доски. Для этого достаточно будет добавить кнопку favorite(☆) для каждого элемента доски.

При этом появляется небольшое диалоговое окно, позволяющее выбрать пункт, к которому он относится, или даже этого не делать. После окончания встречи, каждый выбранный элемент выстраивается в список, в котором можно переставлять\модифицировать\добавлять новые элементы и пункты списка. При этом элементы, отмеченные ☆ для которых мы не выбрали пункт (подразумевается, что это будет происходить в случае, если требования к нашему списку изменились в ходе встречи) будут находится где-то в стороне, чтобы мы могли модифицировать наш список для соответствия новым требованиям.

**Средства**

<http://fabricjs.com/> Данная библиотека решает задачу внешнего вида и поведения доски.

## Защита от нежелательного контента

**Цель**

Применительно к разрабатываемому нами приложению: фильтрация контента, предназначенного для *публичного просмотра*. Фильтрации должны подвергаться следующие категории:

Профиль (вся информация о себе, предоставленная на личной странице пользователя): *имя*, *изображение* (тоже потребное изображение должно быть), *раздел дополнительной информации о пользователе(details).*

Группы (вся информация о группе, предоставленная на странице группы и информация, которая предоставляется группой для публичного просмотра): *название группы*, *изображение* *группы*, *раздел дополнительной информации о группе(info), названия встреч, организуемых группой(meets), текстовая информация и файлы на доске группы.* Я считаю это важным,

поскольку информация на доске группы открыта для всех пользователей, в том числе и для незарегистрированных (ведь им позволяется её просматривать).

**Функциональные требования:**

Система фильтрации состоит из 2 уровней защиты:

Автоматизированная система защиты: имеем фильтр непосредственно текстовой информации, ссылок на внешние ресурсы.

Проверка контента модератором: модератор реагирует на жалобы пользователей. Автоматизированная система защиты представляет собой фильтр, через который проходит вся

текстовая информация, прежде чем она оказывается в базе данных, на доске и т.п. Так, если система обнаружила нецензурное слово, она оставляет первую и последнюю буквы, а остальные заменяет знаком \*. В случае обнаружения ссылки на запрещенные источники информации, выводится «ресурс заблокирован» + знак предупреждения.

Второй уровень фильтрации контента – «ручная» проверка. Администратор группы является её модератором и сам вправе решать какой контент удалить из группы. Тем не менее каждый пользователь имеет возможность пожаловаться модератору на *группу* и *пользователя.* Без «ручной» проверки не обойтись в тех случаях, когда ресурсы загружаются с локальной машины пользователя. Проверка контента модератором осуществляется с определенной периодичностью, и когда группа или пользователь набирает определённое число жалоб, происходит удаление из системы.

**Реализация:**

Фильтр текста я планирую реализовать самостоятельно, в виде модуля .jar, который легко

подключить к проекту. Его можно будет использовать следующим образом: myFilter.clearInput(userInputText);

Кроме того, в этом фильтре можно будет выбрать шаблон фильтрации, то есть каким образом будет преобразовываться нежелательное слово: удаление, замена всех символов звездочками или каким-либо другим знаком и т.д. И, естественно, этот фильтр будет пополняемым, это значит, что можно добавить какие-либо новые выражения в базу данных этого фильтра. Ссылки на внешние ресурсы тоже можно проверять через этот фильтр чисто на непотребное содержание, в виду тенденции давать доменное имя сайтам, характеризующее их содержание.

В помощь для реализации фильтра внешних ресурсов вот этот API: [https://antizapret.info/api.php.](https://antizapret.info/api.php) Фильтр посылает ajax-запрос на указанный url, получает данный в формате json и сверяет ссылки пользователей сети с запрещенными, блокируя не прошедшие проверку. Чтобы трафик большой не шел, часто запросы к этому сервису производится не будут, а, например, раз в неделю в 3 часа ночи, полученные результаты сохраняются в локальном хранилище системы.

*Как можно осуществить жалобу модератору и какие у модератора права доступа к ресурсам?*

В профиле группы есть кнопка *«пожаловаться на группу».* Пользователь ни неё кликает, и жалоба уходит модератору, а именно сначала она фиксируется в базе данных в специально отведенной таблице:

|  |  |  |
| --- | --- | --- |
| complaint\_group\_id | group\_id | user\_id |
|  |  |  |
|  |  |  |

complaint\_id – идентификатор жалобы

group\_id – идентификатор группы, на которую пришла жалоба user\_id – идентификатор пожаловавшегося пользователя

Жалоба на пользователя происходит несколько иным образом: каждое сообщение, отправленное пользователем, сопровождается небольшой кнопочкой «пожаловаться». Это нужно для того, чтобы модератор смог на основании каких-то данных определить, стоит удалять пользователя или нет. Если сделать просто кнопку на профиле пользователя «пожаловаться», то модератор не сможет оценить справедливость/несправедливость жалобы, поскольку, естественно, доступ к переписке пользователя ему предоставлять не будет, а придет только отдельное сообщение, на которое пожаловались. В этом случае таблица в базе данных будет выглядеть так:

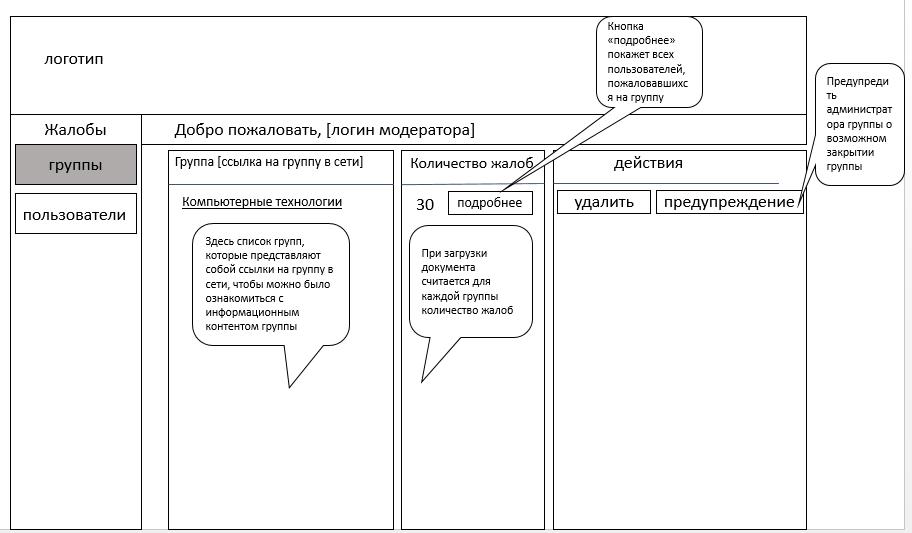
|  |  |  |
| --- | --- | --- |
| complaint\_user\_id | user\_id | message\_text |
|  |  |  |
|  |  |  |

Теперь о том, как выглядит личный кабинет модератора и какие они права. Прежде всего, модераторы хранятся в своей таблице:

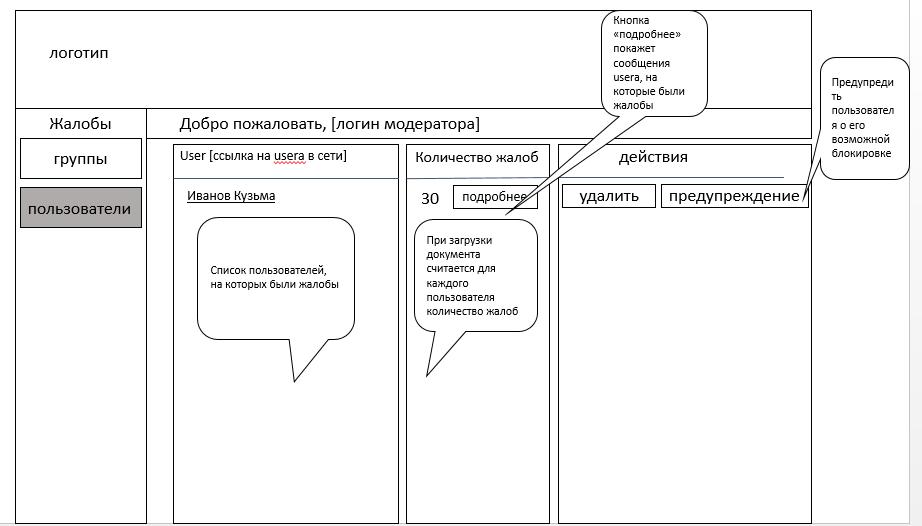
|  |  |  |  |
| --- | --- | --- | --- |
| moderator\_id | moderator\_login | moderator\_password | moderator\_mail |
|  |  |  |  |
|  |  |  |  |

Для модераторов есть свой url входа в личный кабинет, то есть вход производится не через социальную сеть. Модератор попадает в личный кабинет через систему авторизации, реализованную по всем правилам безопасности. Личный кабинет выглядит так:

*При активной вкладке «группы»*



*При активной вкладке «Пользователи»*



Предполагается предоставить модератору послать администратору группы или пользователю сообщение о надвигающемся удалении с испытательным сроком. Ну и понятное дело удаление группы/пользователя совершается sql-запросом к соответствующим таблицам системы.

**Таким образом, нужно как-то наладить систему общения между модераторами и пользователями (администраторами групп). Об этом еще подумать надо.**

## Аудио и видео связь.

## Цель

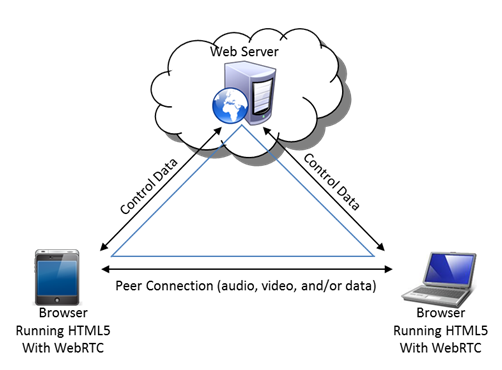
Предоставить пользователям возможность общаться во время встречи непосредственно из самой системы.

## Функциональные требования

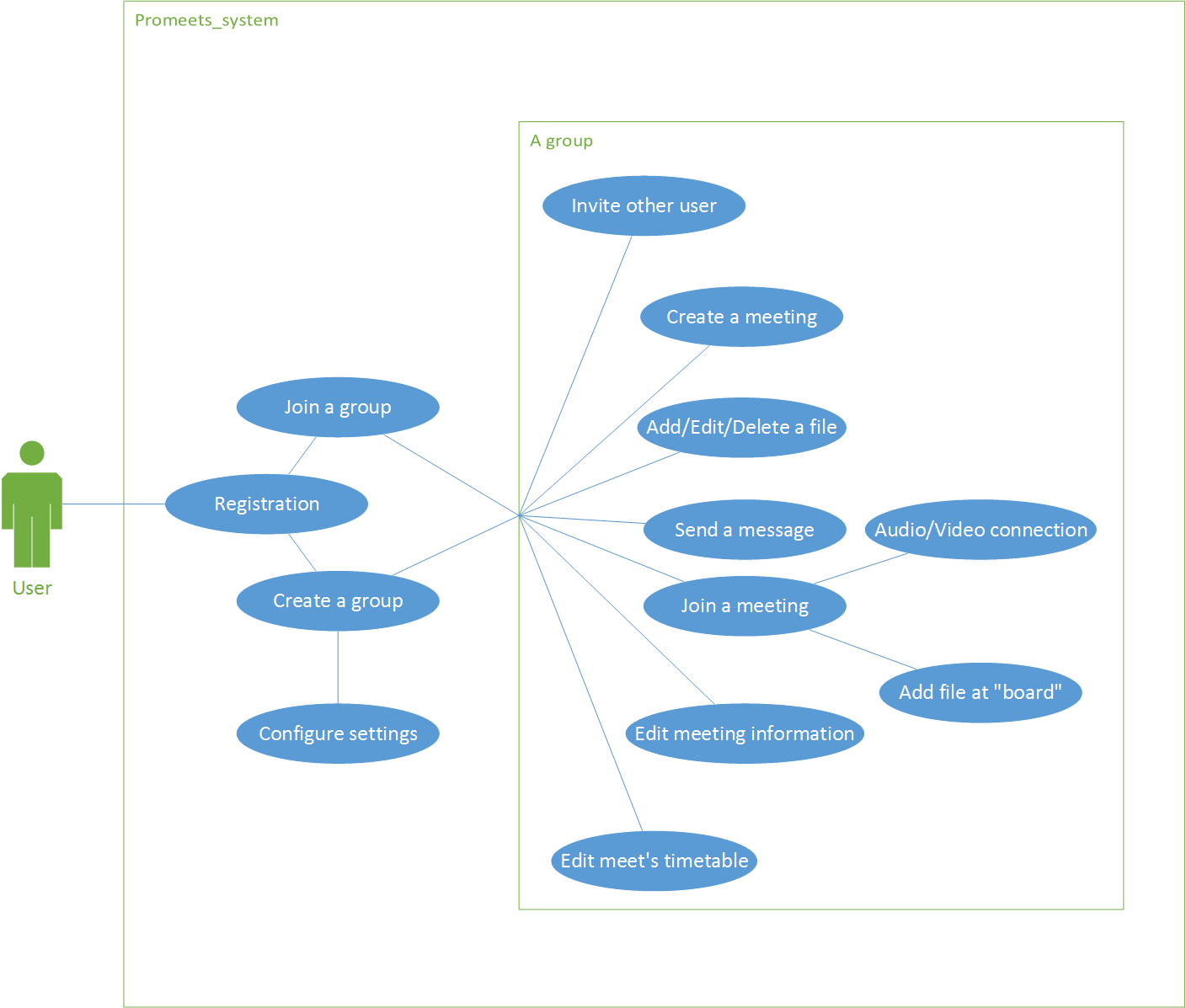
* Качественная аудио и видео связь между всему всеми участниками встречи.
* Ограничение на количество участников – 10.
* Возможность отключения видео или аудио.
* Возможность уменьшения или увеличения размера окошка с видео другого участника.
* Возможность отключить видео или аудио другого участника. («Приглушить»)

## Реализация

Технология «WebRTC», поддерживаемая большинством современных браузеров. (Основана на типе сети “Peer to peer” и использования сервера для установления связи с другим пользователем)



# Promeets use cases



1. Действия внутри группы ограничиваются пользователем, создавшим ее. [↑](#footnote-ref-1)
2. См. сноска 1. [↑](#footnote-ref-2)
3. Зависит от системы голосования. [↑](#footnote-ref-3)