## (план работы)

* суть и цель продукта
* функциональные требования
* нефункциональные требования
* предполагаемые технологии

## Суть продукта

Электронная система для регистрации участников на мероприятия факультета.

## Функциональные требования

1. По ссылке, размещённой на стороннем ресурсе (например, на сайте ВШЭ или в соцсетях), открывается веб-страница мероприятий системы.
   1. На странице представлен список мероприятий, интересующих пользователя. Для каждого из мероприятий указаны: (список уточнить)
      1. Название
      2. Описание
      3. Место проведения
      4. Время проведения
      5. Требования к участникам (возраст, наличие реквизита и т.д.)
      6. Количество доступных мест
   2. Каждое мероприятие также имеет отметку о возможности регистрации (чекбокс).
      1. Мероприятия без доступных мест недоступны для регистрации (вся информация о нём и его чекбокс в состоянии disabled), все остальные - доступны.
      2. При выборе мероприятия становятся недоступными для выбора те, что совпадают с ним по времени (вся информация о них и их чекбоксы в состоянии disabled)
   3. Страница также имеет строку поиска.
      1. Поиск введённого пользователем текста осуществляется по всем информационным полям мероприятий.
   4. Страница также имеет возможность сортировки списка мероприятий:
      1. По времени проведения
      2. По количеству доступных мест
   5. Страница также имеет кнопку “Зарегистрироваться”.
      1. Кнопка становится активной при выборе минимум одного мероприятия.
2. При нажатии кнопки “Зарегистрироваться” открывается веб-страница регистрации.
   1. Страница содержит список выбранных мероприятий.
   2. Страница также содержит кнопку “Назад”.
      1. По нажатию кнопки открывается страница мероприятий для возможности редактирования выбора пользователем.
      2. При возвращении для редактирования уже выбранные пользователем мероприятия остаются отмеченными, в отличие от первоначальной загрузки страницы.
   3. Страница также содержит поля, которые пользователю требуется заполнить: (необходимые поля тоже уточнить офк)
      1. ФИО
      2. Место обучения
      3. Номер телефона
      4. Адрес электронной почты
   4. Страница также содержит кнопку “Отправить заявку”.
      1. После заполнения всех требуемых полей кнопка становится доступной пользователю.
3. После отправления заявки открывается страница с сообщением об успешной регистрации, предлагающая вернуться обратно на исходный ресурс.
4. Информация о заявках хранится и обрабатывается в базе данных системы.
   1. Отправленные пользователями заявки обрабатываются системой и учитываются следующим образом:
      1. Подсчитывается количество занятых мест
      2. Формируются списки имён участников для оформления необходимых документов (списков на вход, сертификатов об участии и т.д.)
      3. Формируются списки контактов для рассылки информации о мероприятии.
5. (вот это всё мы не обсуждали, кстати, но по логике ведь эта часть тоже должна быть реализована?) Администраторы системы имеют возможность управлять списком имеющихся мероприятий:
   1. Создавать, указывая всю информацию, перечисленную в пункте 1.1
   2. Редактировать, изменяя имеющиеся поля информации
   3. Удалять
   4. Группировать, позволяя пользователям получать доступ к разным группам мероприятий, а не к единому списку

## Нефункциональные требования

1. (скорость работы?)
2. (количество обслуживаемых одновременно клиентов?)
3. (максимальный объём базы данных?)

## Предполагаемые технологии

1. Серверная часть - Python посредством фреймворка Django
2. База данных - PostgreSQL
3. Браузерная часть - HTML+CSS

Причины выбора: доступность для разработки, масштабируемость, относительная лёгкость в освоении, популярность у разработчиков, из которой следует наличие множества готовых шаблонов и решений проблем, наличие предыдущего опыта с подобным стеком.