## Алгоритм добавление город в форме ред. профиля соискателя

1. блоки с информацией по городам
2. над всеми блоками выбор Страны и кнопка Добавить город
3. в каждом блоке
   1. город
   2. метро города
   3. Улица
   4. Поле "Другое"
   5. Дни недели

## Алгоритм регистрации пользователя

1. Переходим на /user/register
   1. Тут выводится форма 1-го этапа регистрации в зависимости от параметра “p”
2. После заполнения и проверки формы перегружаемся на /user/register
   1. Проверяем поля на правильность заполнения
   2. Если ошибки - выводим view и сообщаем об ошибках
   3. Если всё правильно:
      1. Записываем пользователя в БД
         1. Устанавливаем ему isblocked - 2 - ожидает активации
      2. Формируем код активации
      3. Формируем письмо активации
      4. Отсылаем пользователю письмо с кодом активации аккаунта
      5. Вызываем view с сообщением об успешности 1-го этапа регистрации и необходимости зайти на почту
3. С почты пользователь переходит на /user/activate
4. Проверяется код активации:
   1. Если неправильный - сообщаем и предлагаем перейти на форму регистрации
   2. Если всё хорошо
      1. Устанавливаем пользователю isblocked = 3 - завершение регистрации
      2. отдаём ему форму 2-го этапа регистрации с доп. полями и уже заполненными первоначалными (из 1-го этапа)
5. После окончания заполнения формы 2-го этапа и перегрузки формы
   1. Проверяем правильность заполненности основных полей
   2. Если ошибки - снова выводим форму и сообщаем об ошибках
   3. Если всё хорошо
      1. устанавливаем пользователю isblocked = 0 - активный
      2. Сообщаем об успешности регистрации на сайте и предлагаем перейти на свой профиль
         1. При этом осуществляется авторизация и уже выводится меню пользователя
6. Если пользователь не закончил 2-й этап регистрации, то когда он снова зайдёт на сайт и введёт свой логин и пароль – его перекинет на страницу 2-го этапа регистрации, чтобы завершить

## Отклик на вакансию

1. Проверяем, если соикатель авторизирован скрываем кнопку отклика
   1. Если он всёравно откликнулся перекидываем на страницу авторизации
2. На странице вакансии проверяем:
   1. Есть ли отклик у пользователя на вакансию
   2. Проходит ли он по возрасту и полу
   3. Пишем соответствующее сообщение или отображаем кнопку отклика
3. Соискатель нажимает кнопку отклика, проверяем совпадение по возрасту и полу, если не подходит:
   1. Во всплывающем окне пишем по какому параметру несовпадение
4. Если всё ок:
   1. Пишем отклик в таблицу vacation\_stat со статусом 0 (новый уведомлять)
   2. Отправляем email на адрес работодателя с уведомлением о новом отклике
5. После этого у работодателя в сообщениях появляется уведомление, что есть новый отклик на вакансию
6. Из письма работодателя он может:
   1. Просмотреть профиль соискателя
   2. Подтвердить отклик
   3. Отклонить отклик
   4. Отметить отклик как просмотренный
   5. Перейти на страницу откликов
7. Когда работодатель переходит в раздел просмотра откликов – всем откликам со статусом 0 устанавливается статус 1 (новый не уведомлять)
   1. Также все новые отклики выделяются от остальных
8. Если работодатель переходит на просмотр профиля соискателя – у отклика проставляется статус 2 (просмотрен)
9. Если работодатель отклоняет отклик – проставляем статус 3 (отклонен) и более этот отклик не отображается у работодателя
10. После установки статуса 2 (просмотрен) работодатель может снова вернуть статус 1 (новый не уведомлять) в этом случае, отклик снова помечается как новый и выделяется среди остальных
11. Если работодатель выставляет статус 4 (принято работодателем) – в этом случае соискателю приходит уведомление на почту и в личный кабинет о том, что его заявка принята
    1. Возле подтвержденного отклика соискателя проставляется фраза *Принята, ожидает подтверждения соискателем*
12. После этого соискатель может перейти на просмотр этой вакансии и подтвердить, что он готов работать, после чего у отклика выставляется статус 5 (обоюдное согласие, начало работы)
    1. После чего высылается уведомление на email работодателя
13. С момента обоюдного подтверждение соискателем и работодателем, начинается отсчёт 7 дней после чего работодатель и соискатель могут выставить друг-другу рейтинги и отзывы
    1. Если вакансия временная и заканчивается раньше 7 дней – выставляем друг-другу рейтинги и отзывы после даты окончания вакансии
    2. Когда истекает 7 дней от даты начала вакансии – высылаем соискателю и работодателю уведомление на почту и в личный кабинет о том, то можно выставить друг-другу рейтинги и отзывы
14. Работодатель может менять статус вакансии:
    1. 0 – открыта (открыта для откликов)
    2. 1 – завершена (закрыта для откликов)
15. Если статус вакансии не 0 – то, соискатель не может оставлять отклики на данную вакансию
16. На странице выставления рейтинга и отзыва:
    1. Соискатель устанавливает по работодателю следующие характеристики:
       1. Соблюдение сроков оплаты
       2. Размер оплаты
       3. Четкость постановки задач
       4. Четкость требований
       5. Контактность
    2. Работодатель выставляет оценки по следующим характеристикам:
       1. Качество выполненной работы
       2. Контактность
       3. Пунтуальность
    3. Также работодатель и соискатель могут оставить отзыв друг-другу, относительно текущей вакансии. Т.е. 1 вакансия – 1 отзыв

## Переписка

1. Открываем страницу чатов:
   1. Выбираем все чаты пользователя с пагинацией
   2. Сортируем по последнему сообщению
   3. Пользователь может перейти на один из чатов
2. После перехода на один из чатов
   1. Открывается текст переписки
   2. Когда пользователь отправляет сообщение, оно post через Ajax на сервер и получаем ответ, после чего добавляется в общий чат в конец
3. Постоянно крутится скрипт проверки, который проверяет наличие новых сообщений на сервере. Как только появляются новые сообщение на сервере
   1. Добавляется разделитель, в котором написано *Новые* *сообщения*
   2. Под ним отображаются сообщения
4. Новые сообщения также проверяются при отправки своего
5. Новые сообщения становятся прочитанными
   1. как только браузер получает фокус
   2. сразу после получения, новые сообщения выводятся как новые, и отправляются на сервер для получения статуса прочитанного
   3. Также при переходе из списка диалогов в чат – новые сообщение выводятся под чертой *Новые сообщения* и отправляются на сервер для смены статуса на *посмотренные*
6. При переходе на страницу работодателя соискатель может начать диалог с работодателем нажав кнопку *Отправить сообщение*
   1. После этого он переходит на страницу чата с этим работодателем:
      1. Открывается интерфейс чата
      2. Предлагается выбрать тему
         1. название одной из вакансий на которую есть отклик
         2. либо вписать тему вручную в поле ввода
   2. Если пользователь пишет сообщение и пытается его оправить без темы – система просит установить тему
   3. Как только тема установлена и пользователь отправляет первое сообщение
      1. Создаётся диалог в таблице диалогов и вносится первое сообщение

## Загрузка фото на сайт

1. Открываем окно для загрузки
2. Пишем ограничения на загрузку
3. Если ограничения не выполнены – в это окно выводим сообщение об ошибке
4. Если всё гут – грузим изображение в /images/applic/tmp
   1. Даём ему имя текущий timestamp
   2. Масштабируем до квадрата 1000х1000
5. Выдаем это фото пользователю на обрезку
6. Пользователь размечает фото на обрезку
7. Обрезаем фото и кладём его в каталог /images/applic
   1. 350х350 кладём в каталог /images/md
   2. 70х70 кладём в каталог /images/xs
8. Имя фото кладём записываем в поле photo для соискателя
9. Для работодателя сохраняем лого в поле logo

## Загрузка заказ карты промму (загрузка фото)

1. Создаём универсальную форму загрузки фото
2. Вызываем UploadManager
   1. Указываем форму для отправки данных на сервер – туда будет созданно hidden поле с названием ID картинки
   2. Указываем ему имя формы
   3. Указываем контейнер с изображениями
   4. Указываем шаблон загруженной фоты
   5. Указываем ему тип изображений
   6. Миниатюры
      1. Указываем будет ли оставляться оригинал (вместо размера пишем ему *orig*)
      2. Указываем какая миниатюра будет открываться при клике на thumbnail
      3. Указываем какая миниатюра (по порядку) будет отдаваться как thumbnail
   7. Устанавливаем кликабельность thumbnail-a
3. Пишем ограничения на загрузку

## Место работы для вакансий

1. Создаём блоки с городами
   1. 1 блок 1 город
      1. Если есть – список метро
   2. Каждый блок с городом имеет несколько локаций
   3. Каждая локация имеет своё название и адрес
      1. Так же у каждой локации своё время работы по дням недели

## Алгоритм добавление вкладок на вакансию

1. Проверяем что это владелец вакансии
2. Добавляем 4 вкладки
3. На каком этапе появляется возможность участвовать в обсуждении?
4. Обсуждение в виде форума?
5. Можно ли принять из отклоненных?

## Алгоритм добавление фото для соискателя

1. Какое кол-во фото можно добавлять? - 50
2. Будут ли они выводится все сразу или по одной? – аватарка + маленькие
3. Как редактировать? Можно сделать как фильтр в анкетах - да
4. Нужно ли сделать смену положения фото или только выбор главной фото? – пока нет
5. Основное фото хранится в RESUME доп фото хранятся в USER\_PHOTO
6. Доп фото привязываются к резюме
7. При выводе фото на странице профиля
   1. Выводится основное фото как сейчас
   2. Остальные фото выводятся маленькими иконками (≈ 50x50)
   3. При нажатии на любом фото – открывается просмотрщик, например, LightBox с возможностью листать на другие фоты
8. При редактировании профиля
   1. Выводим сверху 3 последних фото
   2. При нажатии спойлера – разворачиваются все остальные фото
   3. С фото можно выполнит операции
      1. Сделать главным
      2. Удалить
   4. Можно загрузить новое фото
   5. При загрузке нового фото, оно становится главным (выбираем max npp+1)
   6. При загрузке нового фото – используем обрезку

Алгоритм добавление вкладок на вакансию

* 1. Проверяем что это владелец вакансии
     1. Добавляем 4 вкладки
  2. Принимать участие в обсуждениях можно только после принятия
  3. Обсуждение в виде отзывов
  4. Статус отклоненных и отложенных можно поменять
  5. Вкладки работают через ajax, но меняют URL адрес
  6. Постраничный вывод во вкладках

* 1. На вкладке Откликнувшиеся:
     1. Полный перечень с возможностью отметить отложенным
     2. Утвердить
     3. Отклонить
  2. Утвержденные остаются на тойже вкладке
  3. Отклоненные или отложенные перемещаются на свои вкладки
  4. Алгоритм утверждения заявки как и на соотв. Странице работодателя

## Алгоритм работы с городами

1. Загружаем страницу с городами и локациями
2. Возле каждого города и локации ставим карандаш для входа в режим редактирования
3. При входе в режим редактирования, убираем данные с инфо данного блока и загружаем вместо блока форму с данными по AJAX
4. Каждый блок с данными можно перегружать через AJAX (далее BF)
   1. Если при обновлении данных происходит сбой – сообщаем пользователю, что ему надо перегрузить страницу
5. Пользователь работает с данными и после сохранения – отправляем данные через AJAX на сервер
   1. Если произошел сбой – пишем пользователю об ошибке и предлагаем перегрузить блок с данными (произвести BF)
   2. Если данные сохранены успешно – возвращаем ему сохранённые данные, как при обычном обновлении блока с данными и обновляем блок
   3. Если пользователь отменяет редактирование данных – делаем BF
6. Если пользователь удаляет блок с данными – производим удаление
   1. Если удаление успешно – удаляем блок из интерфейса
   2. Если сбой – предлагаем пользователю сделать BF
7. При добавлении блока с данными отдаём пользователю форму на добавление
   1. При сохранении если всё хорошо – возвращаем данные для функции BF
   2. Если сбой – предлагаем пользователю перезагрузить страницу
8. При публикации новой вакансии – сразу создаём новую вакансию с заголовком поумолчанию
   1. Сообщаем пользователю, что вакансия создана и неактивна, необходимо заполнить данные вакансии и активировать её
   2. Блок с городами и локациями будет привязываться уже к этой вакансии
   3. В конце формы публикации вакансии располагаем кнопку *Сохранить данные* и *Сохранить и активировать*, что означает просто сохранить и сохранить и активировать вакансию соответственно
   4. В списке вакансий, вакансии со статусом неактивным – надо исключить