

Реализация алгоритма отображения списка изменений студента

Приложение позволяет при просмотре информации о студенте просматривать список изменений совершенных в определенный момент времени (Рис. 1). Это сделано для удобства навигации по временной шкале (Рис. 2).

История изменений с момента создания до текущей даты актуализации (8)		
Изменен	Что было изменено	Дата изменения
Адрес фактического проживания	Изменен: с Город Москва, город Щербинка, театральная 14-7 на Город Москва, улица Люблинская, дом 8, кв. 57	2016-06-20
Паспорт	Изменена серия паспорта: с 4514 на 12 Изменен номер паспорта: с 956262 на 354657 Изменена дата выдачи: с 2015-01-28 на 2016-02-09 Изменена фамилия: с Умуразова на Иванова	2016-06-20
Фотография	Изменена: с image1.PNG на image1[1].PNG	2016-06-20
План обучения	Изменена специализация: с Инженер программист на Прикладная информатика и техническая физика	2016-06-14

Рис. 1

Как видно из рисунка, благодаря интерфейсу пользователь может видеть, какие атрибуты были изменены на какие значения с указанием даты изменения.

Серия паспорта:	12
Номер паспорта:	354657
Кем выдан:	отделением уфмс россии по гор москве в гор щербинка
Код подразделения:	770-143
Дата выдачи:	2016-02-09
Место прописки:	Город Москва, город Щербинка, театральная 14-7
Место регистрации:	Город Москва, город Щербинка, театральная 14-7
Место проживания (фактическое):	Город Москва, улица Люблинская, дом 8, кв. 57

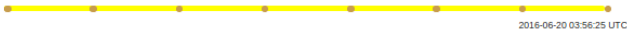


Рис. 2

В форме представления данных отображаются актуальные данные, а на так называемой, **линии событий** или **линии исторический изменений**, в рамках системы, отображаются круговыми элементами даты изменений. Кликая по ним мышкой в инерактивном режиме через браузер с помощью **Ajax** и **JavaScript** происходит асинхронное обращение к серверу с подгрузкой данных студента.

Рассмотрим этот процесс более подробно. При запросе к серверу по дате актуализации, происходит выборка из **ChangeList** (более подробно о реализации темпоральность данного алгоритма смотрите главу «Реализация темпоральности») с подгрузкой данных по **new_id** и **old_id** из таблиц соответствующих типа, указанных в поле **m_type** и соответствующей дате актуализации. Более подробно рассмотрено ниже (Рис. 3).

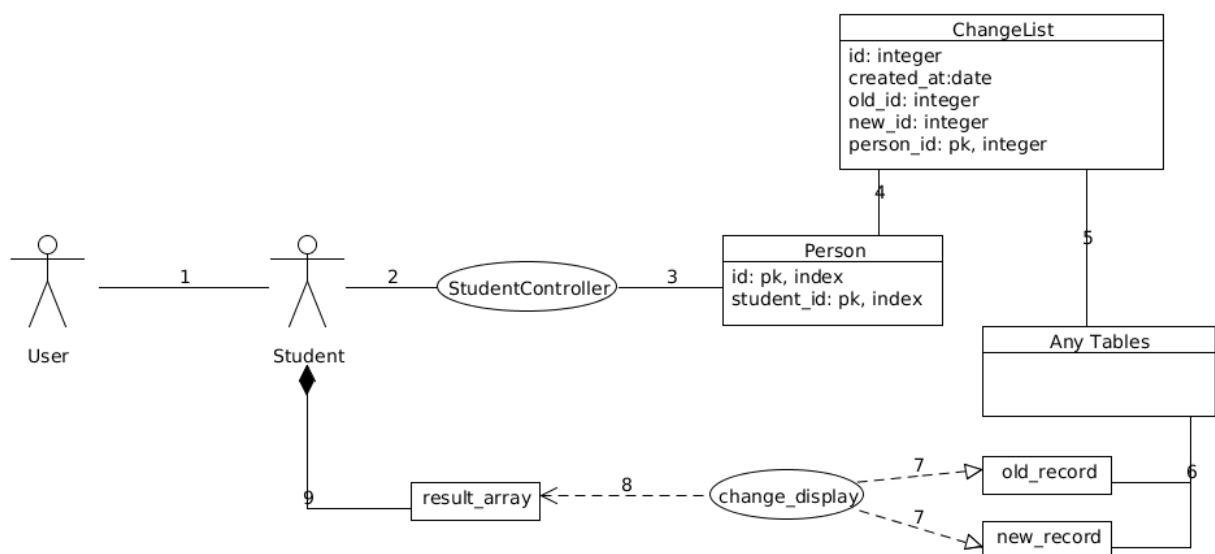


Рис. 3

Определение порядка действий:

1. Пользователь через интерактивный режим обращается к студенту для по вопросу значения его атрибутов согласно дате актуализации (в **GUI** представлено в виде Рис. 2).
2. Клиент обращается к серверу через **Ajax** с заданными параметрами. На сервере срабатывает необходимый обработчик. В данном случае за работу с студентом отвечает **StudentController**.
3. Контроллер подгружает **person_id** через который связан студент с **ChangeList**.
4. По заданным параметрам подгружаются все изменения для данного

студента, сгруппированные по дате актуализации.

5. Так как элементы **ChangeList** содержат **old_id** и **new_id** (связь с предыдущим значением атрибута и замененного), то подгружаются все элементы из таблиц соответствующие полю **m_type**.
6. В результате согласно каждой записи в **ChangeList** имеем по две записи из других таблиц, согласно заданным параметрам.
7. Так как данные в **Ruby on Rails** представлены в виде моделей, то было принято решение в каждой модели данные реализовать метод класса **change_display**, который способен сравнивать значения моделей одного класса и возвращать результат в виде массива.
8. Результирующий массив, каждой сравниваемой модели, помещается в общий массив, тем самым в конечном итоге составляется общий список всех изменений для всех измененных атрибутов.
9. Полученные данные возвращаются на сторону клиента в виде **json** данных, которые в свою очередь обрабатываются и устанавливаются в необходимые поля формы предоставления информации о студенте.

Для повторения заметим что данные подгружаются асинхронно, что очень удобно для пользователя системы при взаимодействии. При успешном переключении между состоянием информации о студенте, активный элемент на **линии событий** подсвечивается красным цветом (Рис. 4 и Рис. 5)

Поиск

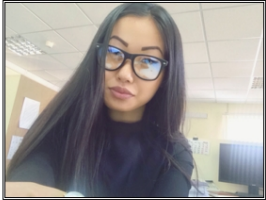
Студенты

Направления/ Специальности

Группы/ Подразделения

Учебные промежутки

Общая (на 2016-06-20) информация



Редактировать

Имя:	Алина
Фамилия:	Иванова
Отчество:	Станиславовна
Дата рождения:	1994-12-14
Место рождения:	г.Москва
Пол:	Женский
Номер студенческого:	123652
Группа	45612
Подразделение:	Московский государственный машиностроительный университет
Специальность:	Инженер программист
Направление:	Прикладная информатика и техническая физика
Учебный семестр	Осень/Лето
Учебный год	2016/2017
Когда выдан:	2016-06-14

Серия паспорта:	12
Номер паспорта:	354657
Кем выдан:	отделением уфмс росси по гор москве в гор щербинка
Код подразделения:	770-143
Дата выдачи:	2016-02-09
Место прописки:	Город Москва, город Щербинка, театральная 14-7
Место регистрации:	Город Москва, город Щербинка, театральная 14-7
Место проживания (фактическое):	Город Москва, улица Люблинская, дом 8, кв. 57

История изменений с момента создания до текущей даты актуализации (6)

Рис. 4 До запроса по дате актуализации

Поиск

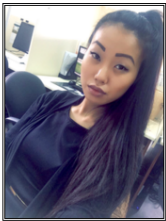
Студенты

Направления/ Специальности

Группы/ Подразделения

Учебные промежутки

Общая (на 2016-06-20) информация



Редактировать

Имя:	Алина
Фамилия:	Уразкова
Отчество:	Станиславовна
Дата рождения:	1994-12-14
Место рождения:	г.Москва
Пол:	Женский
Номер студенческого:	123652
Группа	2016-06-14
Подразделение:	4514
Специальность:	956262
Направление:	отделением уфмс росси по гор москве в гор щербинка
Учебный семестр	770-143
Учебный год	2015-01-28
Когда выдан:	2016-06-14

Серия паспорта:	Город Москва, город Щербинка, театральная 14-7
Номер паспорта:	Город Москва, город Щербинка, театральная 14-7
Кем выдан:	Город Москва, город Щербинка, театральная 14-7
Код подразделения:	770-143
Дата выдачи:	2016-02-09
Место прописки:	Город Москва, город Щербинка, театральная 14-7
Место регистрации:	Город Москва, город Щербинка, театральная 14-7
Место проживания (фактическое):	Город Москва, город Щербинка, театральная 14-7

История изменений с момента создания до текущей даты актуализации (6)

Рис. 5 После запроса по дате актуализации

На данных картинках изменены такие паспортные данные как серия , номер паспорта, фамилия и адрес фактического проживания. К примеру это может быть если человек заключает брачный союз, тогда могут измениться данные атрибуты.

Для удобства взаимодействия в форму представления изменений была подключена постраничная навигация. Так же для людей с пониженным зрением была реализована просмотра фотографии в режиме галереи (Рис. 6).

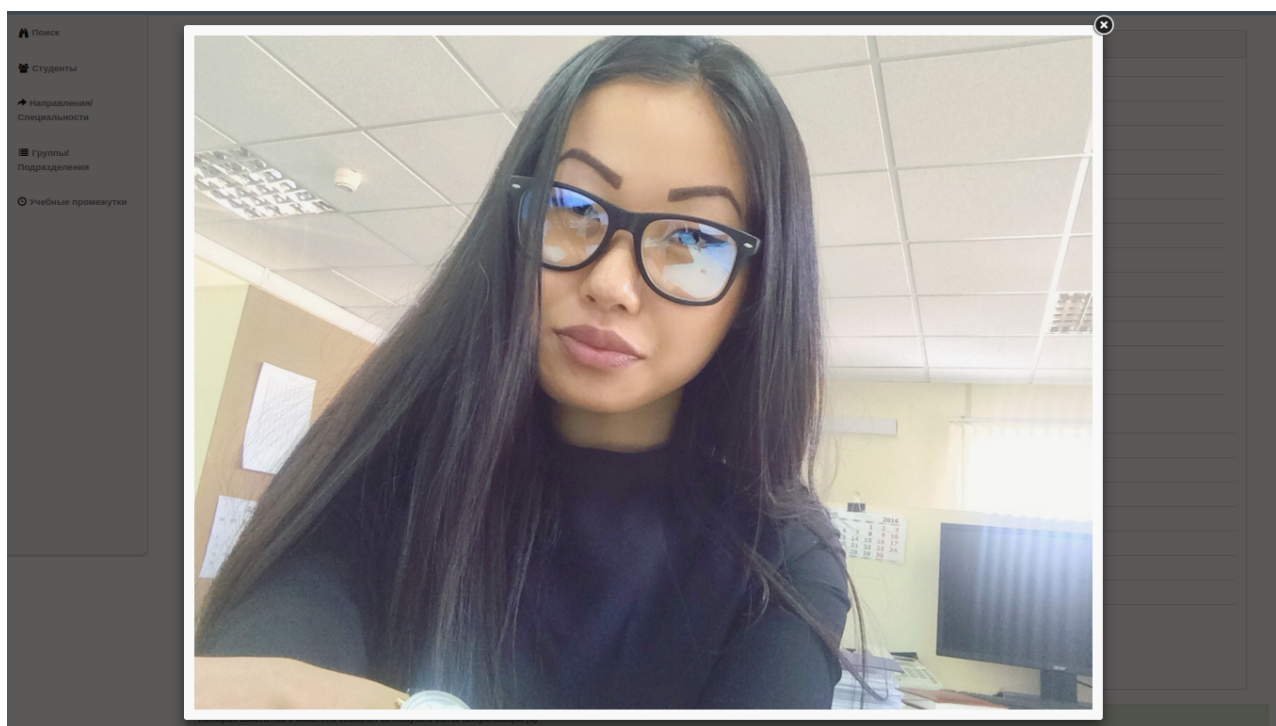


Рис. 6 Режим галереи

Ниже для примера приведена реализация **change_display** - метод класса характеризующий адрес (**Address**) в системе (Рис. 7):

```
class Address < ActiveRecord::Base
  belongs_to :person

  ADDRESS_TYPE = {1=>'Адрес прописки', 2 => 'Адрес регистрации', 3 => 'Адрес фактического проживания'}

  public

  def change_display(address)
    result = [ADDRESS_TYPE[self.a_type]]
    result << ['Изменен',self.address, address.address] if self.address != address.address
    result << address.created_at
    return result
  end
end
```

Рис. 7 Реализация **change_display**.

Пример дан для ознакомления общего подхода к решению задачи. В других моделях данных реализация **change_display** зависит от полей, которые они содержат, но алгоритм реализации таков.

Тем самым список история изменений (Рис. 1) является вспомогательным инструментом для пользователя системы при взаимодействия с **линией событий** (Рис. 2) .