Санкт-Петербургский Государственный Политехнический Университет

Институт информационных технологий и управления

Кафедра Компьютерных Систем и Программных Технологий

**Курсовой проект**

**Отчет по дисциплине «Проектирование архитектур ПО»**

**По теме «Портал государственных услуг»**

**Работу выполнила студентка группы № 13541/3**

Фомина М. А. \_\_\_\_\_\_\_\_\_\_\_

**Работу принял преподаватель:**

Зозуля А.В. \_\_\_\_\_\_\_\_\_\_\_

.

Санкт-Петербург

2017

## Этапы курсового проекта

1. Постановка задачи
   * Описание назначения проектируемой системы
   * Функциональные требования (текстовое описание Участников и их Интересов)
   * Описание бизнес-процессов (этапы, Участники, вовлеченные Сущности)
2. Разработка вариантов использования
   * Диаграммы прецедентов для всех ролей
   * Подробное текстовое описание всех вариантов использования с альтернативами
3. Разработка статической объектной модели предметной области
   * Моделирование предметной области при помощи диаграммы классов
4. Разработка динамической объектной модели предметной области
   * Моделирование предметной области при помощи диаграмм последовательностей
5. Проектирование слоя бизнес-логики
   * Сравнительный анализ и выбор архитектурного шаблона уровня бизнес-логики
6. Реализация слоя бизнес-логики (Java, NetBeans)
   * Применение паттернов проектирования
   * Реализация unit-тестирования всех вариантов использования (JUnit)
   * Использование вместо слоя хранения шаблона "Репозиторий"
7. Проектирование слоя источников данных
   * Выбор архитектурного шаблона уровня доступа к данным
   * Источники данных: реляционная БД, внешний сервис (REST API, xml/json/yaml файл)
8. Реализация слоя источников данных
   * Корректировка unit-тестов
9. Проектирование сервисного слоя и слоя представления GUI (Swing)
   * Клиенты бизнес-логики: GUI, внешний сервис (xml/json/yaml по HTTP)
10. Реализация слоев представления, сервисного слоя
    * Реализация внешнего API при помощи [Simple high-level embedded HTTP server](https://docs.oracle.com/javase/8/docs/jre/api/net/httpserver/spec/com/sun/net/httpserver/package-summary.html)
11. Комплексное тестирование системы
12. Написание пояснительной записки (включает все разделы, указанные выше, а также выводы)

**Содержание пояснительной записки**

1. Постановка задачи
2. Варианты использования
3. Моделирование
4. Слой бизнес-логики
5. Слой источников данных
6. Сервисный слой
7. Слой представления
8. Интеграция
9. Тестирование
10. Выводы

**1. Постановка задачи**

**1.1. Описание назначения проектируемой системы**

Система, проектируемая в ходе выполнения курсового проекта, является системой автоматизации для оказания государственных услуг, а именно, процессов, связанных с записью ребёнка в школу; записью на приём к врачу; отправкой отзывов об образовательных и медицинских учреждениях, а также о конкретных врачах.

**1.2. Функциональные требования (текстовое описание участников и их интересов)**

**Гражданин.**

1. Гражданин может заполнить информацию о своих детях и о себе.
2. Гражданин может отправлять заявки на запись ребёнка в образовательное учреждение.
3. Гражданин может отменить заявку на запись ребёнка в образовательное учреждение.
4. Гражданин может записываться на приём к врачу.
5. Гражданин может отменить свою запись на приём к врачу.
6. Гражданин может оставить вопрос, отзыв или жалобу об образовательных и медицинских учреждениях.

**Представитель учебного учреждения.**

1. Добавляет информацию об учебном учреждении.
2. Добавляет информацию о количестве свободных мест для приёма школьников.
3. Одобряет заявки по приёму школьников.
4. Назначает время для прихода в школу гражданина для сдачи документов и окончательного оформления школьника.
5. После оформления документов окончательно подтверждает, что ребёнок будет учиться в школе. Если же документы не были своевременно оформлены, то имеет право отклонить заявку.
6. Отвечает на вопросы и жалобы граждан.

**Представитель медицинского учреждения.**

1. Добавляет информацию о медицинском учреждении и врачах.
2. Добавляет информацию о свободных талонах.
3. Отменяет талоны, выданные на приём к врачу, если врач заболел.
4. Отвечает на вопросы и жалобы граждан.

**1.3. Описание бизнес-процессов (этапы, участники, вовлеченные Сущности)**

**Участники:** гражданин, представитель учебного учреждения, представитель медицинского учреждения.

**Сущности:** талон, отзыв, заявка на запись в учебное учреждение, учреждение, ребёнок, врач.

**Бизнес-процессы:**

**1.3.1. Запись ребёнка в школу**

1. Представители учебных учреждений добавляют информацию об учреждениях и о свободных местах.
2. Гражданин выбирает образовательное учреждение, в которое хочет отправить ребёнка, и подаёт заявку в него. Гражданин может подать какое угодно количество заявок в разные учреждения.
3. Представитель учебного учреждения подтверждает заявку.
4. Гражданин подтверждает, что действительно выбирает это учреждение, или отклоняет свою заявку на запись.
5. Представитель учебного учреждения назначает время для прихода в школу гражданина для сдачи документов и окончательного оформления школьника.
6. Представитель учебного учреждения окончательно подтверждает, что ребёнок будет учиться в школе. При подтверждении все заявки в другие учреждения, отправленные гражданином, автоматически отменяются.
7. Гражданин добавляет отзыв о работе учебного учреждения и приёме.
8. Представитель учебного учреждения отвечает на отзыв.

**1.3.2. Запись на приём к врачу**

1. Представители медицинских учреждений добавляет информацию о талонах.
2. Гражданин записывается на приём к врачу, выбирая поликлинику, врача и время приёма.
3. Представитель медицинского учреждения может отменить выданные талоны в непредвиденной ситуации, к примеру, если врач заболел. Тогда гражданам должны прийти уведомления о том, что приём отменён.
4. Гражданин может отменить свою запись к врачу. Тогда его талон будет доступен для других людей.
5. Представитель медицинского учреждения подтверждает, что гражданин посетил доктора.
6. Гражданин добавляет отзыв о медицинском учреждении.
7. Представитель медицинского учреждения отвечает на отзыв.

**2. Разработка вариантов использования**

**2.1. Диаграмма прецедентов для всех ролей**

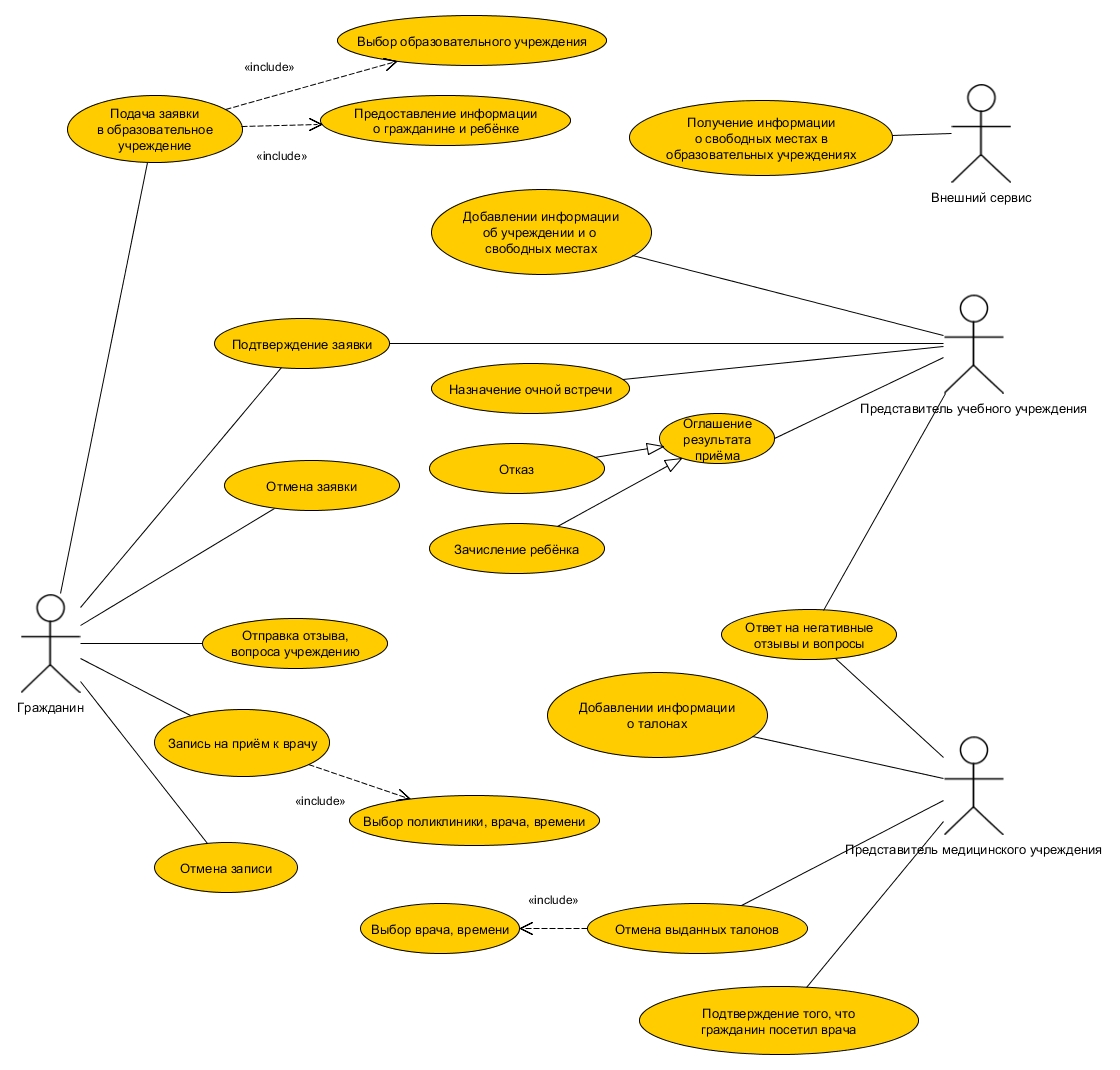


Рис. 1. Диаграмма прецедентов для всех ролей

**2.2. Подробное текстовое описание всех вариантов использования с альтернативами**

**2.2.1. Запись ребёнка в школу**

1. Представитель учебного учреждения добавляет информацию об учреждениях и о свободных местах.
2. Гражданин выбирает образовательное учреждение, в которое хочет отправить ребёнка.
3. Гражданин подаёт заявку в образовательное учреждение.
4. Представитель учебного учреждения подтверждает заявку.
5. Гражданин подтверждает, что действительно выбирает это учреждение.
6. Представитель учебного учреждения назначает время для прихода в школу гражданина для сдачи документов и окончательного оформления школьника.
7. Представитель учебного учреждения окончательно подтверждает, что ребёнок будет учиться в школе
8. Гражданин добавляет отзыв о работе учебного учреждения и приёме.
9. Представитель учебного учреждения отвечает на отзыв.

**Альтернатива.**

5а. Гражданин отклоняет свою заявку на запись. После этого переход на шаг 2.

**Альтернатива.**

6а. Гражданин отклоняет свою заявку на запись. После этого переход на шаг 2.

**Альтернатива.**

7а. Представитель учебного учреждения отклоняет заявку, по причине не прихода родителя на очную встречу и не приносу необходимых документов. После этого переход на шаг 2.

**2.2.2. Запись на приём к врачу**

1. Представитель медицинского учреждения добавляет информацию о талонах.
2. Гражданин выбирает поликлинику, врача, время.
3. Гражданин записывается на приём к врачу.
4. Представитель учебного учреждения подтверждает, что гражданин посетил врача по талону.
5. Гражданин пишет отзыв о медицинском учреждении.
6. Представитель медицинского учреждения отвечает на отзыв.

**Альтернатива.**

4а. Гражданин отменяет свою запись к врачу. Тогда переход к шагу 2. Талон доступен другим гражданам для записи.

**Альтернатива.**

4а. Представитель медицинского учреждения отменяет выданные талоны. Тогда переход к шагу 2.

**Альтернатива.**

4а. Представитель медицинского учреждения не подтверждает посещение врача по талону.

**3. Разработка статической объектной модели предметной области**

**3.1. Моделирование предметной области при помощи диаграммы классов**

**Участники:** гражданин, представитель учебного учреждения, представитель медицинского учреждения.

**Сущности:** талон, отзыв, заявка на запись в учебное учреждение, учреждение, ребёнок, врач.

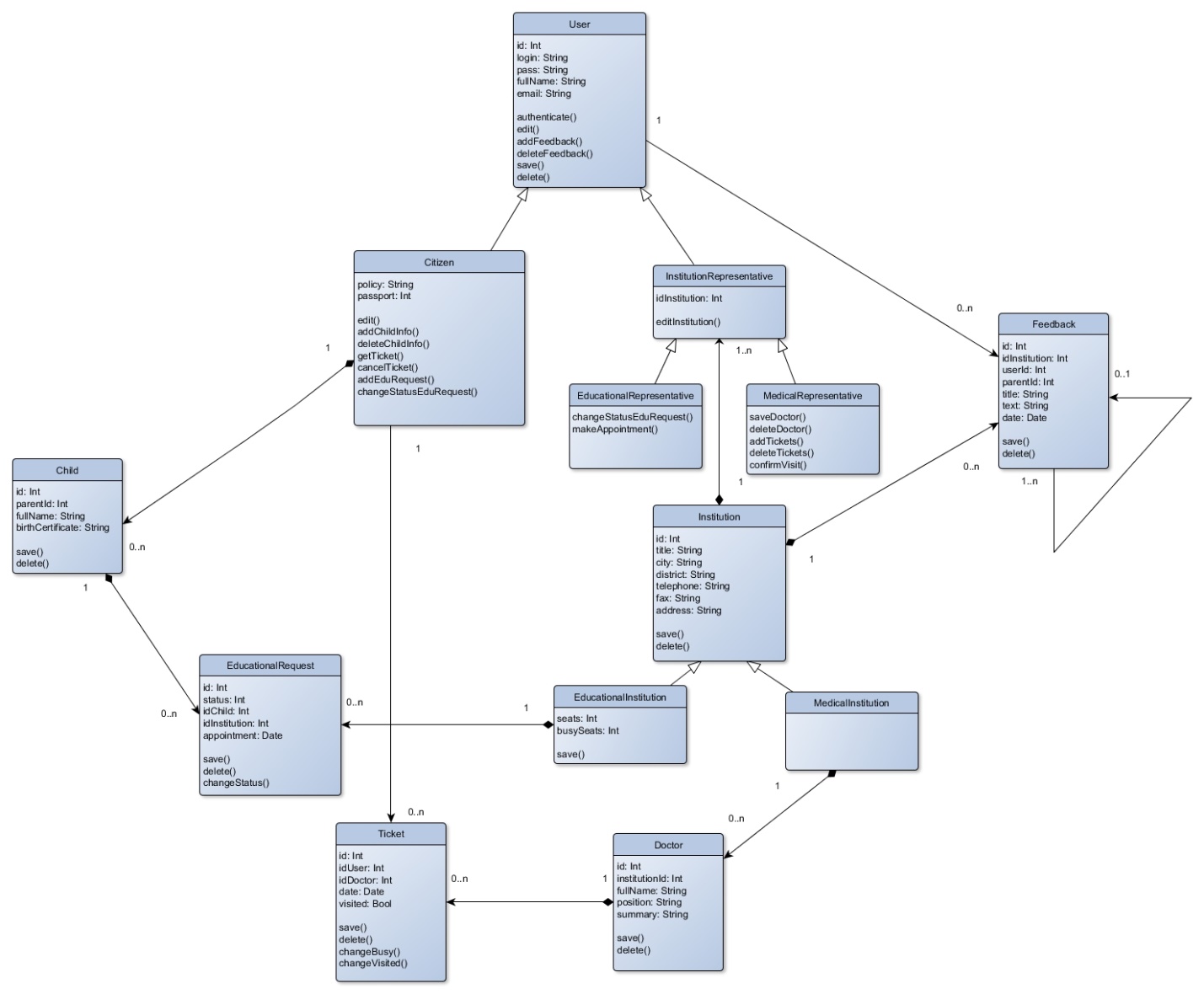
****

Рис. 2. Диаграмма классов

**4. Разработка динамической объектной модели предметной области**

**4.1. Запись ребёнка в школу**

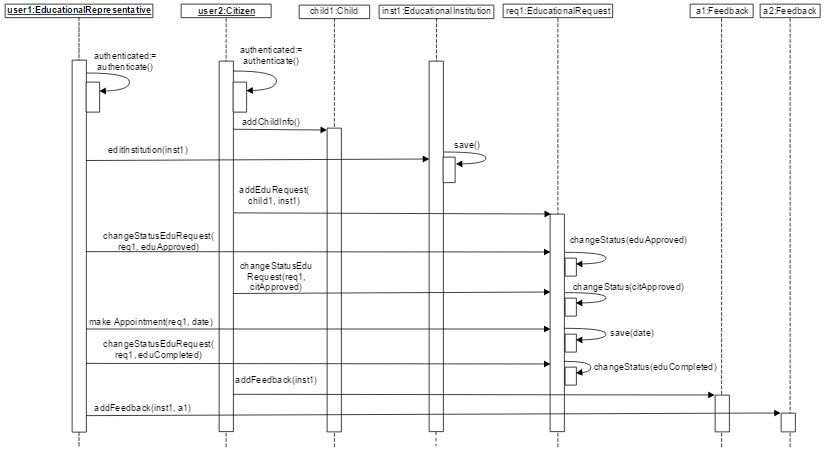


Рис. 3. Диаграмма последовательности для бизнес-процесса записи ребёнка в школу

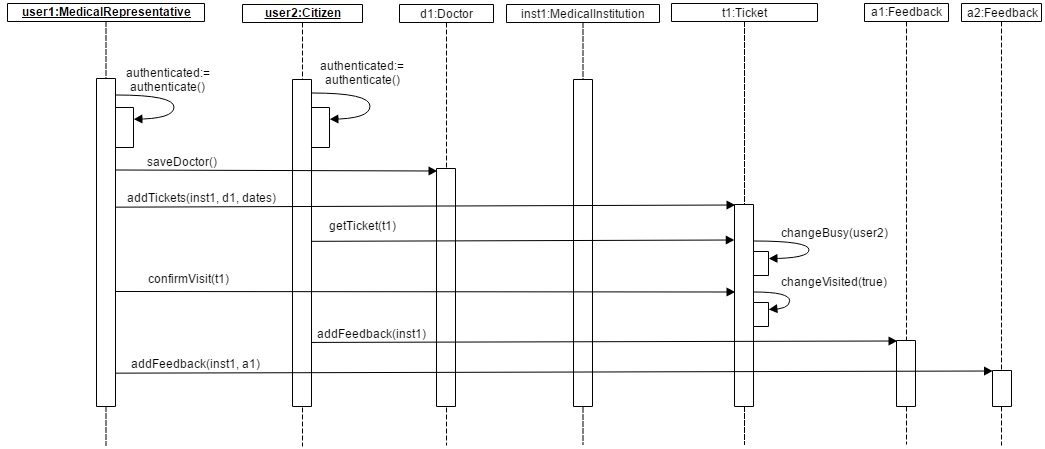


Рис. 4. Диаграмма последовательности для бизнес-процесса записи на приём к врачу

1. **Проектирование слоя бизнес-логики**

В качестве шаблона, который будет использован при создании слоя бизнес-логики была выбрана модель предметной области. Каждый объект наделяется функциями, соответствующими его природе. Образуется сеть взаимосвязанных объектов. Функции тесно сочетаются с данными.

**Список используемых источников**

* + 1. <http://www.yworks.com/products/yed/download#ReleaseNotes> – Редактор yEd
    2. <https://www.gliffy.com/> - Online-редактор gliffy (для диаграмм-последовательностей)

## <http://tiger.ftk.spbstu.ru/trac/edu-arh-2017>

## Литература

* Конспект лекций ([слайды](http://kspt.ftk.spbstu.ru/course/architecture))
* UML. Основы. 2-е издание. М. Фаулер, К. Скотт
* Предметно-ориентирование проектирование. Э. Эдванс
* [Архитектура корпоративных программных приложений. М. Фаулер](http://books.google.ru/books?id=MdMM1ZrICnsC)
* Приемы объектно-ориентированного проектирования. Э. Гамма, Р. Хелм и др.
* Шаблоны интеграции корпоративных приложений. Г. Хоп, Б. Вульф и др.

## On-line каталоги типовых решений

* <http://citforum.ru/SE/project/pattern/> (RU)
* <http://ooad.asf.ru/Patterns.aspx> (RU)
* <http://www.oodesign.com/> (EN)
* <http://design-pattern.ru/> (RU)
* <http://thierryroussel.free.fr/java/books/martinfowler/www.martinfowler.com/isa/> (EN)
* <http://martinfowler.com/eaaCatalog/> (EN)
* <http://www.enterpriseintegrationpatterns.com/toc.html> (EN)

## Проектные хостинги

* <http://bitbucket.org/>
* <http://github.com/>

## UML-редакторы

### online

* [CodeUML online](http://www.codeuml.com/)
* [yUML online](http://yuml.me/)
* [Online Diagram](https://www.draw.io/)
* [gliffy](https://www.gliffy.com/)

### desktop

* [Dia](http://live.gnome.org/Dia)
* [yEd](http://www.yworks.com/en/products_yed_about.html)
* [PlantUML](http://plantuml.sourceforge.net/)