МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

«Новосибирский национальный исследовательский государственный университет»

(Новосибирский государственный университет)

Структурное подразделение Новосибирского государственного университета –

Высший колледж информатики Университета

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

**ОТЧЕТ по производственной практике по профессиональному модулю**

**ПМ03 «Участие в интеграции программных модулей»**

**Тема практики: Разработка приложения**

**«Разработка системы дистанционного тестирования пациентов в клинических исследованиях»**

Руководитель

к.ф.-м.н. ИВМиМГ СО РАН Лукинов В. Л.

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2017 г.

Студента 4 курса Рыбкин О. Ю.

гр. 315в1 «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_2017 г.

Новосибирск

2017

**СОДЕРЖАНИЕ**

[Перечень сокращений и условных обозначений 4](#_Toc476113415)

[Введение 5](#_Toc476113416)

[1 Описание предметной области 7](#_Toc476113417)

[2 Постановка задачи 8](#_Toc476113418)

[3 Аналоги 9](#_Toc476113419)

[4 Функциональные требования к программному средству 10](#_Toc476113420)

[4.1 Взаимодействие компонентов системы 10](#_Toc476113421)

[4.1.1 Веб-сайт 10](#_Toc476113422)

[4.1.2 RESTful API 10](#_Toc476113423)

[4.1.3 Сервер 11](#_Toc476113424)

[4.1.4 База данных 11](#_Toc476113425)

[4.1.5 Диаграмма взаимодействия 11](#_Toc476113426)

[4.2 Взаимодействие пользователей системы 12](#_Toc476113427)

[4.2.1 Пациенты 12](#_Toc476113428)

[4.2.2 Специалисты по клиническим исследованиям 12](#_Toc476113429)

[4.2.3 Администраторы 12](#_Toc476113430)

[4.2.4 Диаграмма прецедентов 13](#_Toc476113431)

[5 Нефункциональные требования к программному средству 14](#_Toc476113432)

[5.1 Требования к надёжности 14](#_Toc476113433)

[5.2 Требования к безопасности 14](#_Toc476113434)

[6 Характеристика выбранных программных сред и средств 15](#_Toc476113435)

[6.1 Каркас веб-приложений Symfony 15](#_Toc476113436)

[6.2 Платформа Android 15](#_Toc476113437)

[7 Описание базы данных 17](#_Toc476113438)

[Заключение 27](#_Toc476113439)

[Список использованных источников 28](#_Toc476113440)

# Перечень сокращений и условных обозначений

Каркас веб-приложений (англ. Web application framework) – сборка готовых программных компонентов, позволяющая автоматизировать рутинные действия, связанные с разработкой веб-приложений. Каркас веб-приложений позволяет упростить разработку веб-приложений, мотивируя разработчика на использование определённой архитектуры приложения, разделение кода на логические компоненты, повторное использование кода в разных компонентах программы.

REST (англ. Representational State Transfer – передача состояния представления) – стиль взаимодействия компонентов сетевого приложения. Клиент отправляет серверу запрос на определённый адрес (URI) в определённом формате (XML, HTML, JSON), как пример, запрос на получение списка всех пользователей программы. В зависимости от запроса сервер выполняет необходимое клиенту действие и отправляет ему ответ, содержащий требуемую клиентом информацию.

RESTful API – интерфейс программирования приложений с использованием принципов REST. Интерфейс определяет, на какой адрес необходимо отправлять запросы, в каком формате должны быть реализованы запросы.

# Введение

Темой данной выпускной квалификационной работы является разработка программной системы, позволяющей пациентам проходить дистанционное тестирование в клинических исследованиях. Взаимодействуя с системой через мобильное приложение или веб-браузер, пациент выбирает необходимый тест, выполняет его и сохраняет результаты на сервере для анализа.

Клиническое исследование – научное исследование, в котором специально обученные специалисты исследуют действие новых или существующих лекарственных препаратов с участием групп добровольно согласившихся пациентов. Прежде чем попасть на рынок, новые препараты должны пройти клинические исследования для подтверждения их безопасности и эффективности. При проведении клинических исследований пациенты обязаны докладывать о своём состоянии специалистам, проводящим это исследование.

C развитием уровня Web- и мобильных технологий и увеличением доступности сети Интернет возможность дистанционного тестирования представляет собой достаточно выгодную и осуществимую функциональность для внедрения и реализации в медицинской инфраструктуре. При корректной реализации дистанционное тестирование позволяет снизить затраты медицинского персонала и пациентов клинических исследований, предоставляя пациенту возможность прохождения тестирования не выходя из дома и снижая время присутствия пациента в лечебном учреждении. Также дистанционное тестирование позволяет проводить постоянное наблюдение за здоровьем пациентов вне зависимости от их местоположения, что увеличивает количество получаемой информации и уменьшает вероятность совершения ошибок, а также улучшает общее впечатление пациента о клиническом исследовании.

Целью данной работы является создание программной инфраструктуры, использующей модель клиент-сервер и позволяющей пациентам (клиентам) выполнять дистанционное тестирование с сохранением результатов на сервере для дальнейшего анализа в клинических исследованиях.

Данная разработка предназначена для использования в клинических исследованиях для проведения дистанционного тестирования с помощью персональных компьютеров и мобильных устройств.

Для достижения цели были выполнены следующие задачи:

* + Изучена предметная область выполняемой цели [1].
  + Изучен комплект средств разработки Android для создания мобильного Android-приложения [2].
  + Изучен каркас веб-приложений Symfony [3].
  + Выявлены основные функциональные и нефункциональные требования к программному продукту, создана документированная спецификация этих требований.
  + Создана постановка выполняемой задачи.
  + Спроектирована архитектура создаваемой системы.
  + Спроектирован пользовательский интерфейс для создаваемых веб-сайта и мобильного приложения.
  + Выполнена реализация программной системы.
  + Выполнены анализ, тестирование и отладка программной системы.
  + Подведены итоги к полученным результатам.

Для выполнения задач использовались следующие операционные и программные средства:

* Операционная система Windows 7.
* Виртуальная машина с установленной в ней операционной системой Xubuntu 16.04.2 LTS.
* Интегрированная среда разработки Android Studio 2.2.3.
* Виртуальная машина с установленной в ней операционной системой Android 4.1 Jelly Bean.
* Веб-сервер Nginx 1.10.0.
* Каркас веб-приложений Symfony 3.2.0.
* База данных PostgreSQL 9.5.4.

# Описание предметной области

После успешного проведения доклинических исследований нового препарата с положительными результатами и получения согласия от уполномоченных органов здравоохранения проводится клиническое исследование нового препарата.

Клиническое исследование препарата изначально проводится с участием небольшого числа больных (тестовая группа) и здоровых (контрольная группа) людей. Число людей увеличивается по мере накопления положительной информации о безопасности и эффективности препарата.

По мере проведения клинического исследования собирается информация о различных показателях, таких, как: пульс, давление, метаболизм, ощущаемое самочувствие, безопасное дозирование препарата. Собранная информация используется для определения:

* Эффективности нового лекарственного препарата по сравнению с уже существующими.
* Переносимости нового препарата по сравнению с уже существующими.
* Работоспособности нового препарата в случаях, когда лечение существующими препаратами безуспешно.
* Экономической выгоды нового препарата.
* Простоты и удобства нового препарата.

Сбор информации также производится после регистрации препарата для массового применения для получения информации об эффектах и безопасности препарата на большой популяции.

Одной из форм сбора информации от пациентов является тестирование. Используя различные тесты, такие, как Short Form-36, специалисты по клиническим исследованиям собирают информацию о состоянии здоровья пациента, производят оценку качества жизни пациента. Тестирование до, во время и после применения препарата позволяет определить влияние препарата на данные показатели, что в дальнейшем помогает улучшить эффективность и безопасность проверяемого препарата.

# Постановка задачи

Для сбора информации для клинического исследования необходимо, чтобы пациент находился в ответственном за клиническое исследование лечебном учреждении. Данная необходимость создаёт дополнительные трудности для пациента и специалистов по клиническим исследованиям: пациенту необходимо время и в некоторых случаях денежные средства, чтобы прибыть в лечебное учреждение, встреча должна быть организована в удобное для пациента и специалиста время. В случае невозможности самостоятельного прибытия пациента в учреждение необходимость присутствия пациента создаёт ещё больше трудностей для выполнения клинического исследования.

Для помощи в решении данной проблемы была поставлена задача создания системы дистанционного тестирования. Использование сети Интернет предоставляет возможность передавать необходимую информацию на огромные расстояния, что помогает снизить время, необходимое для присутствия пациента в лечебном учреждении.

Система дистанционного тестирования позволяет пациенту, используя веб-страницу в браузере либо приложение для смартфона, проходить необходимое для клинического исследования тестирование. Приложение для смартфона также предоставляет дополнительные возможности по прохождению тестирования пациентом в связи с имеющимися у смартфона сенсорами.

Для создания данной системы необходимо выполнить следующее:

* Проектирование и реализация мобильного приложения.
* Проектирование и реализация веб-сайта.
* Проектирование и реализация базы данных.
* Проектирование и реализация серверного приложения для принятия и обработки запросов с мобильного приложения.
* Тестирование взаимодействия компонентов системы и работы системы.

# Аналоги

Одним из аналогов данного программного средства является программное обеспечение TAO (Testing Assisté par Ordinateur – тестирование с помощью компьютеров) [4]. TAO упрощает разработку программных платформ, целью которых является предоставление возможности тестирования с помощью цифровых устройств – компьютеров, смартфонов, планшетов.

TAO предоставляет такие возможности, как:

* Создание различных типов тестов, предпросмотр результатов на различных типах устройств и размерах экранов.
* Изменение доступности тестов для разных пользователей, длительности тестов, максимального количества попыток.
* Взаимодействие с специальными браузерами, которые позволяют наблюдать за тестируемым во время тестирования в целях пресечения обмана.
* Создание отчётов по результатам тестирований.
* Контроль уровня доступа для различных групп пользователей.

# Функциональные требования к программному средству

## Взаимодействие компонентов системы

Система состоит из следующих компонентов: веб-сайт, RESTful API, сервер и база данных.

### Веб-сайт

Взаимодействие пользователя с с сервером через веб-сайт следующим образом:

* Пользователь заходит на веб-сайт, браузер получает веб-страницу от сервера и отображает её пользователю.
* Пользователь взаимодействует со страницей (производит авторизацию, проходит тестирование), в результате чего браузер отправляет данные на сервер.
* Сервер принимает и обрабатывает данные и по необходимости сохраняет их в базе данных.
* Сервер отправляет браузеру результат обработки данных и по необходимости новую веб-страницу.

### RESTful API

Взаимодействие пользователя с сервером через RESTful API происходит через мобильное приложение следующим образом:

* Пользователь использует мобильное приложение - проходит тестирование, настраивает календарь.
* Приложение производит синхронизацию с сервером – происходит передача данных о выполненных тестах и настройках календаря на сервер с помощью запросов через RESTful API с использованием заранее определённых адресов и форматов.
* Сервер получает данные и отправляет мобильному приложению информацию о новых тестах и информацию об изменениях, внесённых в календарь специалистами.

### Сервер

На сервере располагаются веб-сайт для браузеров и RESTful API для мобильных приложений. База данных может располагаться либо на сервере, либо на другом компьютере (другом сервере).

### База данных

Сервер считывает, записывает и удаляет данные в базе данных.

### Диаграмма взаимодействия

Взаимодействие компонентов системы изображено на рисунке 4.1.

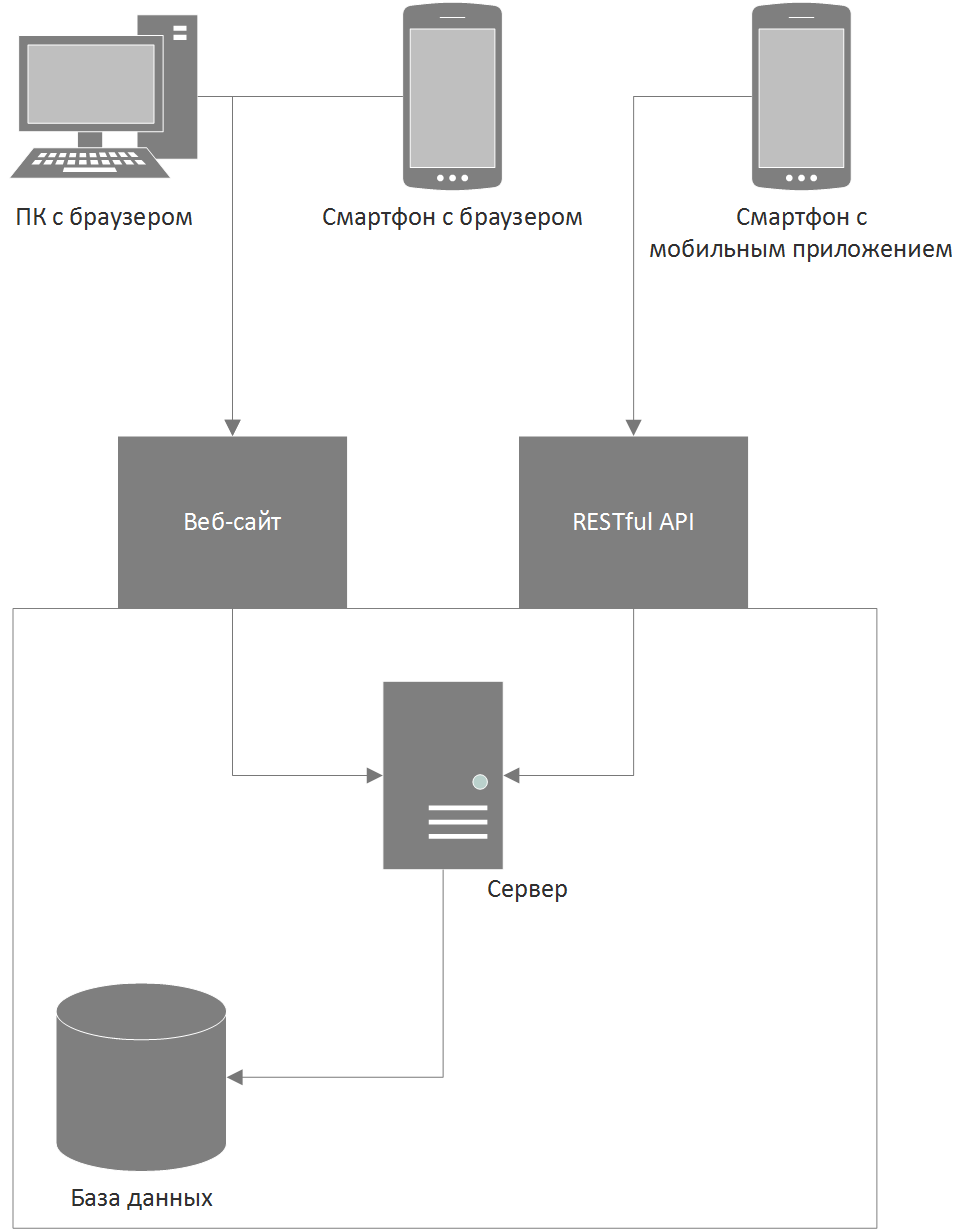


Рисунок 4.1 – Диаграмма взаимодействия компонентов системы

## Взаимодействие пользователей системы

### Пациенты

Программное средство предоставляет следующий функционал пациентам клинических исследований:

* Создание учётной записи или аутентификация с использованием существующей учётной записи.
* Изменение параметров собственной учётной записи.
* Удаление собственной учётной записи.
* Выполнение тестов через Android-приложение в смартфоне или веб-страницу в браузере.
* Изменение расписания тестов в календаре с возможностью напоминания через приложение, сайт или электронную почту.
* Отправление электронной почты специалистам, ответственным за пациента.

### Специалисты по клиническим исследованиям

Специалисты по клиническим исследованиям получают следующий функционал:

* Аутентификация с использованием существующей учётной записи.
* Просмотр результатов тестирования участвующих в проводимых специалистом клинических исследованиях пациентов.
* Добавление и удаление пациентов-участников в клинических исследованиях, которые проводит специалист.
* Изменение расписания тестов в календаре для пациентов.
* Отправление электронной почты участникам клинических исследований.

### Администраторы

Администраторам даются следующие возможности:

* Аутентификация с использованием существующей учётной записи.
* Создание и удаление учётных записей.
* Изменение данных и ролей учётных записей.
* Добавление, изменение и удаление тестовых данных пациентов.
* Создание и удаление тестов, доступных для выполнения пациентам.

### Диаграмма прецедентов

Взаимодействие пользователей системы изображено в виде диаграммы прецедентов (Use-case diagram) на рисунке 4.2.

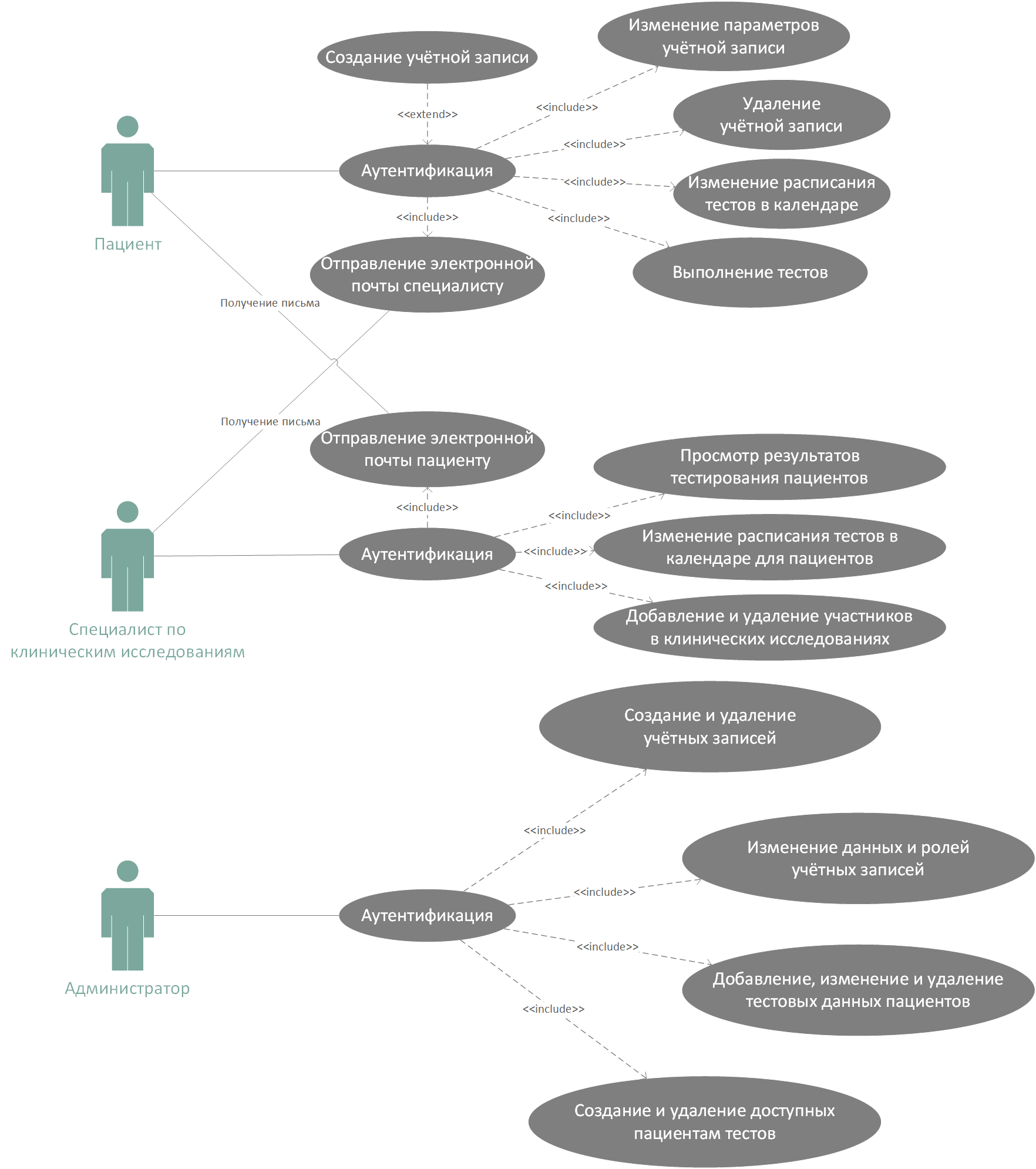


Рисунок 4.2 – Взаимодействие пользователей в виде диаграммы прецедентов

# Нефункциональные требования к программному средству

## Требования к надёжности

К программному средству предъявляются следующие требования в плане надёжности:

* Ввод некорректных данных пользователем не должен приводить к сбою в программе.
* Порча и подделка данных со стороны клиента не должна приводить к сбою в сервере.

## Требования к безопасности

К программному средству предъявляются следующие требования в плане безопасности:

* Шифрование всей информации, передаваемой через сеть Интернет. Используемый стандарт шифрования должен быть не слабее стандарта AES, применяемого с размером ключа 128 бит.
* Использование и своевременное обновление доверенного SSL-сертификата для обеспечения шифрования.
* Обеспечение контроля уровня доступа пользователей.
* Хранение паролей для учётных записей пользователей в базе данных в виде хеш-сумм, созданных с использованием модификатора (соли). Используемая криптографическая функция формирования ключа должна быть не слабее функции bcrypt.
* Защита от перебора паролей (метод «грубой силы»).
* Обеспечение защиты от внедрения SQL-кода через запросы пользователя.

# Характеристика выбранных программных сред и средств

## Каркас веб-приложений Symfony

Для разработки веб-сайта было решено использовать каркас веб-приложений Symfony. Причиной выбора Symfony послужили следующие факторы:

* Проверенность временем данного продукта: дата выпуска первой версии Symfony – 22-ое октября 2005 года.
* Спонсирование и поддержка продукта компанией SensioLabs.
* Большое сообщество разработчиков, использующих Symfony.
* Высокая конфигурируемость и наличие подробной документации.
* Наличие множества подключаемых компонентов, предоставляющих дополнительный функционал.
* Использование Symfony в таких больших проектах, как Drupal, phpBB, Laravel.
* Лицензия MIT, обеспечивающая отсутствие необходимости денежной компенсации и отсутствие ограничений на разработку и применение веб-приложений, использующих Symfony.

Недостатками Symfony являются:

* Сложность разработки веб-приложений по причине большого количества встроенного функционала и высокой конфигурируемости в Symfony.
* Повышенная вероятность наличия ошибок в программе из-за высокой конфигурируемости Symfony.

В связи с сложностью архитектуры разрабатываемого программного продукта высокая конфигурируемость необходима. Перечисленные недостатки компенсируются путём тщательного тестирования корректности работы программы.

## Платформа Android

Для разработки мобильного приложения было решено использовать операционную систему Android как платформу. Причиной выбора Android послужили следующие факторы:

* Большое количество пользователей.
* Наличие легкодоступного и бесплатного комплекта средств разработки.
* Поддержка продукта компанией Google.

У платформы Android существует множество недостатков, таких, как отсутствие обновлений у некоторых производителей для своих устройств, использующих систему Android, но в связи с тем, что платформа Android из всех мобильных платформ имеет наибольшее количество пользователей, данные недостатки избежать невозможно, если продукт предназначен для массовой аудитории.

# Описание базы данных

Все поля всех таблиц обязательны к заполнению и не могут содержать в себе NULL-значения.

Таблица 7.1 - таблица users

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | serial | Идентификатор (номер) пользователя  Первичный ключ |
| username | varchar(100) | Имя пользователя для входа в учетную запись |
| pass\_hash | varchar(255) | Хеш пароля пользователя с модификатором (солью) |
| first\_name | varchar(50) | Имя пользователя |
| last\_name | varchar(50) | Фамилия пользователя |
| registration\_datetime | timestamptz | Дата и время регистрации пользователя |
| last\_login\_datetime | timestamptz | Дата и время последнего входа в учётную запись |
| is\_active | boolean | Активность пользователя – вход в учётную запись возможен только в состоянии «истина» |

Таблица 7.2 - таблица user\_phones

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | integer | Идентификатор пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |
| phone | varchar(50) | Номер телефона  Первичный ключ |

Таблица 7.3 - таблица user\_emails

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | integer | Идентификатор пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (1 или больше) |
| email | varchar(255) | Адрес ящика электронной почты  Первичный ключ |

Таблица 7.4 - таблица user\_addresses

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | integer | Идентификатор пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_id в таблице users, отношение один-к-одному (0 или 1) |
| address | varchar(100) | Адрес  Первичный ключ |

Таблица 7.5 – таблица user\_patronymics

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | integer | Идентификатор пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_id в таблице users, отношение один-к-одному (0 или 1) |
| patronymic | varchar(50) | Отчество  Первичный ключ |

Таблица 7.6 – таблица user\_contact\_info

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | integer | Идентификатор пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_id в таблице users, отношение один-к-одному (0 или 1) |
| contact\_info | text | Отчество  Первичный ключ |

Таблица 7.7 – таблица user\_pictures

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | integer | Идентификатор пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_id в таблице users, отношение один-к-одному (0 или 1) |
| picture\_path | text | Путь к изображению  Первичный ключ |

Таблица 7.8 – таблица user\_messages

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_message\_id | serial | Идентификатор сообщения  Первичный ключ |
| user\_id\_recipient | integer | Идентификатор пользователя-получателя  Внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |
| user\_id\_sender | integer | Идентификатор пользователя-отправителя  Внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |
| message | text | Текст сообщения |
| datetime | timestamptz | Дата и время отправления сообщения |
| read\_by\_recipient | boolean | Статус прочтения получателем |
| deleted | boolean | Статус удаления |

Таблица 7.9 – таблица user\_blocks

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_block\_id | serial | Идентификатор блокировки  Первичный ключ |
| user\_id\_blocked | integer | Идентификатор заблокированного пользователя  Внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |
| user\_id\_blocker | integer | Идентификатор заблокировавшего пользователя  Внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |

Таблица 7.10 – таблица user\_bans

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| ban\_id | serial | Идентификатор бана  Первичный ключ |
| user\_id | integer | Идентификатор пользователя  Внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |
| reason | varchar(100) | Причина бана |
| ban\_datetime | timestamptz | Дата и время бана |
| until\_datetime | timestamptz | Дата и время окончания бана |
| permanent | boolean | Статус бесконечности бана |

Таблица 7.11 – таблица user\_roles

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_role\_id | serial | Идентификатор роли пользователя  Первичный ключ |
| name | varchar(50) | Название роли пользователя |

Таблица 7.12 – таблица users\_xref\_user\_roles

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_id | integer | Идентификатор пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |
| user\_role\_id | integer | Идентификатор роли пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_role\_id в таблице user\_roles, отношение многие-к-одному (0 или больше) |

Таблица 7.13 – таблица permissions

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| permission\_id | serial | Идентификатор права доступа  Первичный ключ |
| name | varchar(50) | Название права доступа |

Таблица 7.14 – таблица user\_roles\_xref\_permissions

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| user\_role\_id | integer | Идентификатор роли пользователя  Первичный ключ, внешний ключ, ссылается на поле user\_role\_id в таблице user\_roles, отношение многие-к-одному (0 или больше) |
| permission \_id | integer | Идентификатор права доступа  Первичный ключ, внешний ключ, ссылается на поле permission\_id в таблице permissions, отношение многие-к-одному (0 или больше) |

Таблица 7.15 – таблица tests

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_id | serial | Идентификатор теста  Первичный ключ |
| name | varchar(100) | Название теста |

Таблица 7.16 – таблица test\_questions

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_question\_id | serial | Идентификатор вопроса к тесту  Первичный ключ |
| test\_id | integer | Идентификатор теста  Внешний ключ, ссылается на поле test\_id в таблице tests, отношение многие-к-одному (1 или больше) |
| contents | text | Текст вопроса |
| can\_select\_multiple\_answers | boolean | Возможность выбора множества ответов |

Таблица 7.17 – таблица test\_question\_pictures

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_question\_id | integer | Идентификатор вопроса к тесту  Первичный ключ, внешний ключ, ссылается на поле test\_question\_id в таблице test\_questions, отношение многие-к-одному (0 или больше) |
| picture\_path | text | Путь к изображению  Первичный ключ |

Таблица 7.18 – таблица test\_answers

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_answer\_id | serial | Идентификатор ответа к вопросу теста  Первичный ключ |
| test\_question\_id | integer | Идентификатор вопроса к тесту  Внешний ключ, ссылается на поле test\_question\_id в таблице test\_questions, отношение многие-к-одному (1 или больше) |
| contents | text | Текст ответа |

Таблица 7.19 – таблица test\_answer\_pictures

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_answer\_id | integer | Идентификатор ответа к тесту  Первичный ключ, внешний ключ, ссылается на поле test\_answer\_id в таблице test\_answers, отношение многие-к-одному (0 или больше) |
| picture\_path | text | Путь к изображению  Первичный ключ |

Таблица 7.20 – таблица test\_results

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_result\_id | serial | Идентификатор результата теста  Первичный ключ |
| test\_id | integer | Идентификатор теста  Внешний ключ, ссылается на поле test\_id в таблице tests, отношение многие-к-одному (0 или больше) |
| user\_id | integer | Идентификатор пользователя  Внешний ключ, ссылается на поле user\_id в таблице users, отношение многие-к-одному (0 или больше) |
| datetime | timestamptz | Время завершения теста |
| result | text | Текст результата |

Таблица 7.21 – таблица test\_result\_answers

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_result\_answer\_id | integer | Идентификатор ответа результата теста  Первичный ключ |
| test\_result\_id | integer | Идентификатор результата теста  Внешний ключ, ссылается на поле test\_result\_id в таблице test\_results, отношение многие-к-одному (1 или больше) |
| test\_question\_id | integer | Идентификатор вопроса теста  Внешний ключ, ссылается на поле test\_question\_id в таблице test\_questions, отношение многие-к-одному (0 или больше) |

Таблица 7.22 – таблица test\_result\_answers\_xref\_test\_answers

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_result\_answer\_id | integer | Идентификатор ответа результата теста  Первичный ключ, внешний ключ, ссылается на поле test\_result\_answer\_id в таблице test\_result\_answers, отношение один-к-одному (0 или 1) |
| test\_answer\_id | integer | Идентификатор вопроса теста  Первичный ключ, внешний ключ, ссылается на поле test\_answer\_id в таблице test\_answers, отношение многие-к-одному (0 или больше) |

Таблица 7.23 – таблица test\_result\_answer\_custom\_answers

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип** | **Описание** |
| test\_result\_answer\_id | integer | Идентификатор ответа результата теста  Первичный ключ, внешний ключ, ссылается на поле test\_result\_answer\_id в таблице test\_result\_answers, отношение один-к-одному (0 или 1) |
| custom\_answer | text | Текст произвольного ответа  Первичный ключ |

# Заключение

Для выполнения поставленных задач была реализована программная система, позволяющая пациентам проходить дистанционное тестирование в клинических исследованиях.

Были проделаны следующие виды работ:

* + Изучена предметная область выполняемой цели.
  + Изучен комплект средств разработки Android для создания мобильного Android-приложения.
  + Изучен каркас веб-приложений Symfony.
  + Выявлены основные функциональные и нефункциональные требования к программному продукту, создана документированная спецификация этих требований.
  + Создана постановка выполняемой задачи.
  + Спроектирована архитектура создаваемой системы.
  + Спроектирован пользовательский интерфейс для создаваемых веб-сайта и мобильного приложения.
  + Выполнена реализация программной системы.
  + Выполнены анализ, тестирование и отладка программной системы.
  + Подведены итоги к полученным результатам.

# Список использованных источников

1. Ассоциация организаций по клиническим исследованиям [Электронный ресурс] / АОКИ. — Web-сайт. — Ассоциация организаций по клиническим исследованиям (AOKИ), 2008-2017. — URL : http://acto-russia.org/, свободный. — Яз. рус. — (Дата обращения: 09.01.2017).
2. Google Inc. Introduction to Android [Электронный ресурс] / Google ; Android Developers. — Web-сайт. — Google Inc., 2017. — URL : https://developer.android.com/guide/index.html, свободный. — Яз. англ. — (Дата обращения: 12.01.2017).
3. SensioLabs. Symfony Documentation [Электронный ресурс] / SensioLabs. — Web-сайт. — SensioLabs, 2017. — URL : https://symfony.com/doc/current/index.html, свободный. — Яз. англ. — (Дата обращения: 06.02.2017).
4. Open Assessment Technologies S.A. TAO Testing Homepage [Электронный ресурс] / OAT. — Web-сайт. — Open Assessment Technologies S.A., 2017. — URL : https://www.taotesting.com/, свободный. — Яз. англ. — (Дата обращения: 22.01.2017).