

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ, НГУ)

Кафедра компьютерных систем

Воронов Артём Васильевич

Разработка медицинской информационной системы НГУ

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ
по направлению высшего профессионального образования
230100.68 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА
ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Тема диссертации утверждена распоряжением по НГУ № 4 от «11» января 2012г.
Тема диссертации скорректирована распоряжением по НГУ № 180 от «14» мая 2013г.

Руководитель

Пищик Б. Н.

к.т.н., доцент

Новосибирск, 2013г.

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ, НГУ)

Кафедра компьютерных систем

УТВЕРЖДАЮ

Зав. Кафедрой Пищик Б. Н.

.....
(подпись, дата)

ЗАДАНИЕ
на магистерскую диссертацию

студент Воронов Артём Васильевич

факультета ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Направление подготовки 230100.68 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ
ТЕХНИКА

Магистерская программа: Информационное и программное обеспечение
автоматизированных систем

Тема: Разработка медицинской информационной системы НГУ

Цели работы: исследование задачи создания медицинской информационной системы и
решение этой задачи для медицинского центра НГУ – лечебно-профилактического
учреждения амбулаторного типа.

Руководитель

Пищик Б. Н.

к.т.н., доцент

.....
(подпись, дата)

СОДЕРЖАНИЕ

| | |
|--|-----------|
| ВВЕДЕНИЕ | 4 |
| 1 ОПИСАНИЕ ПРОБЛЕМЫ..... | 5 |
| 2 ПОСТАНОВКА ЗАДАЧИ И ЦЕЛИ РАБОТЫ | 7 |
| 3 ВЫБОР ПОДХОДА К ПОСТРОЕНИЮ МИС | 8 |
| 3.1 Покупка проприетарной МИС | 8 |
| 3.2 Использование свободно распространяемой МИС..... | 8 |
| 3.3 Разработка собственного продукта | 8 |
| 4 ОБЗОР СИСТЕМЫ OPENMRS..... | 9 |
| 5 ТЕХНОЛОГИИ ДЛЯ РАЗРАБОТКИ МИС | 10 |
| 6 ПРОТОТИП МЕДИЦИНСКОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ..... | 12 |
| 6.1 Функциональные требования к электронной регистратуре | 12 |
| 6.2 Архитектура..... | 14 |
| 6.3 Элементы безопасности..... | 19 |
| 6.4 Пользовательский интерфейс | 20 |
| ЗАКЛЮЧЕНИЕ | 24 |
| ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ | 25 |
| ЛИТЕРАТУРА | 26 |
| ПРИЛОЖЕНИЕ А. ЭЛЕМЕНТЫ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА | 27 |

ВВЕДЕНИЕ

Данная работа выполнялась в рамках проекта по разработке информационной системы для медицинского центра НГУ. Проект изначально определялся как некоммерческий, и предполагалось создание программного продукта в рамках архитектуры web-приложений на основе современных технологий и свободно распространяемого программного обеспечения.

Работа связана с проблемой создания и внедрения медицинской информационной системы (МИС), которая возникает при автоматизации процессов в лечебно-профилактическом учреждении (ЛПУ). В данном случае медицинский центр НГУ рассматривался как ЛПУ амбулаторного типа. Заказчиком проекта описывалась проблема автоматизации процесса оказания медицинской помощи, где пациент рассматривается от точки инициации, когда он впервые приходит в медицинское учреждение, и в регистратуре на него заводится амбулаторная карта, и до точки выхода, когда пациента выписывают, и на него оформляется стат. талон, который впоследствии направляется в фонд обязательного медицинского страхования.

Данная проблема не является новой и широко обсуждается в различных статьях, книгах и научных работах [1, 2, 3, 4]. Имеются экспертные сообщества, которые занимаются этим вопросом, более того разработаны стандарты, регулирующие различные аспекты информатизации медицинских учреждений. Существуют также уже готовые решения: только в России по данным Ассоциации развития медицинских информационных технологий (АРМИТ) зарегистрированы сведения о 775 различных программных продуктов для здравоохранения [5].

В таких условиях основной сложностью при решении данной проблемы является разработка программного продукта, который бы одновременно удовлетворял требованиям заказчика проекта, регламентам, используемым в медицинских учреждениях, и тем рекомендациям и нормативной базе, разработанной для информатизации ЛПУ.

В работе рассматривается описание вышеуказанной проблемы, формальная постановка задачи и целей работы, описание выбранного метода решения и полученного в результате программного продукта.

1 Описание проблемы

В данной работе рассматривается проблема автоматизации процессов в лечебно-профилактическом учреждении на примере медицинского центра НГУ. Суть проблемы заключается в разработке программного продукта, который бы одновременно удовлетворял требованиям заказчика проекта, регламентам, используемым в медицинских учреждениях, и тем рекомендациям и нормативной базе, разработанной для информатизации медицинских учреждений.

Со стороны заказчика имеются требования автоматизировать процесс оказания медицинской помощи от момента инициации пациента, когда он впервые приходит в медицинское учреждение, и в регистратуре на него заводится амбулаторная карта, и до точки выхода, когда пациента выписывают, и на него оформляется стат. талон, который впоследствии направляется в фонд обязательного медицинского страхования. Причём необходимо не просто повторить функционал большинства современных медицинских информационных систем в рамках данного процесса, но и обеспечить интеграцию информации о пациентах с информацией о студентах, чтобы можно было в будущем проводить различные исследования и делать медицинские отчёты на основе различных характеристик, например, таких как курс, успеваемость, факультет. Важными задачами в данном случае являются:

- Упростить работу медицинского персонала по работе с документацией, а именно обеспечить автозаполнение тех или иных полей, использование шаблонов при их создании и проверку орфографии.
- Обеспечить возможность печати основной медицинской документации. Например, чтобы каждый новый приём у врача мог быть представлен в виде отдельного листа бумаги, и соответственно мог бы быть прикреплен к амбулаторной карте.
- Обеспечить механизмы по работе с электронной медицинской картой пациента

В регламентах регулирующих работу медицинского персонала с пациентами можно обнаружить множество других требований, которые необходимо учитывать при разработке системы:

- Существуют ограничения по времени на приём пациента у врача. Имеется ряд инструкций, которые описывают какая информация должна регистрироваться в статистическом талоне амбулаторного пациента (форма 025-12/у), в медицинской карте амбулаторного пациента (форма № 025/у-04), в контрольной карте диспансерного наблюдения. (форма № 030/у-04), в направлении на

госпитализацию, восстановительное лечение, обследование, консультацию (форма № 057/у-04) и в другой медицинской документации.

- Пациент должен быть информирован об обработке его персональных данных.

Кроме того существуют требования со стороны стандартов в области разработки МИС [6, 7, 8, 9], которые также должны учитываться при подготовке решения для медицинского центра НГУ. К ним относятся:

- Общие требования, регламентирующие структуру электронной персональной медицинской записи (ЭПМЗ) и электронных медицинских архивов, обеспечение их сохранности, неизменности и достоверности.
- Требования к организации прав доступа, к электронной цифровой подписи, к персонифицируемости и к пользовательским интерфейсам ЭПМЗ
- Требования к созданию бумажной копии электронной амбулаторной карты (ЭАК) и ЭПМЗ.

Если принять во внимание тот факт, что рынке медицинских информационных систем существует множество различных решений (только в России по данным АРМИТ зарегистрированы сведения о 775 различных программных продуктов для здравоохранения [5]), то задача автоматизации ЛПУ сводится к проблеме выбора: поиск готового решения или же создание собственного решения с привлечением специалистов в сфере информационных технологий.

2 Постановка задачи и цели работы

Совместно с научным руководителем и другими участниками проекта по данной проблеме была поставлена задача автоматизации основных этапов процесса оказания медицинской помощи в медицинском центре НГУ: работу регистратуры и приёма пациента у врача.

В рамках поставленной задачи были сформулированы следующие цели для данной работы:

1. Исследовать задачу создания медицинской информационной системы.
2. Решить эту задачу для медицинского центра НГУ – лечебно-профилактического учреждения амбулаторного типа.

Для достижения данных целей было решено использовать следующие методы: обзор открытых и специализированных источников о проблемах разработки МИС и их внедрения в ЛПУ, разработка прототипа МИС, поиск и анализ готовых решений в области медицинских информационных технологий.

В результате перед автором работы были поставлены следующие задачи:

- Исследовать задачу разработки электронной регистратуры, сформулировать требования и разработать модель прецедентов, где отражались бы основные субъекты взаимодействующие с системой и доступный им функционал.
- Спроектировать архитектуру всей системы в целом и выделить архитектурно-значимые элементы, участвующие в реализации функционала электронной регистратуры.
- Сделать обзор используемых технологий для построения современной МИС, исследовать тенденции и используемые практики в этой сфере.
- Реализовать функционал электронной регистратуры.
- Подготовить площадку для дальнейшего развития системы, как свободно распространяемого программного обеспечения.

3 Выбор подхода к построению МИС

Существует несколько подходов к построению информационной системы для медицинского учреждения. Основными подходами являются: покупка и внедрение проприетарной системы, использование свободно распространяемой МИС с открытым исходным кодом и разработка собственного продукта.

3.1 Покупка проприетарной МИС

Среди всех продуктов, предназначенных для автоматизации процессов в лечебно-профилактических учреждениях, а также являющиеся проприетарными можно выделить следующие наиболее известные системы: ДОКА+, Медиалог, Пациент, Карельская медицинская информационная система (КМИС), IntelClinic, MGERM.

Исходя из того, что проект является некоммерческим, и выполняется силами студентов, то данный подход не был взят за основу. Однако, за время работы в проекте удалось ознакомиться с системой ДОКА+ на уровне пользователя, для того чтобы было с чем сравнивать при поиске другой системы и при разработке собственной МИС. Для этого участникам проекта была организована экскурсия в Институт клинической и экспериментальной медицины (ИКЭМ), где система ДОКА+ эксплуатируется уже несколько лет.

3.2 Использование свободно распространяемой МИС

Изначально система задумывалась как свободное программное обеспечение с открытым исходным кодом. Поэтому данный подход изначально казался наиболее перспективным. К наиболее известным МИС, которые являются свободным программным обеспечением, относятся: OpenMRS, FreeMED, GNUmed, GNU Health, ClearHealth, OpenEHR, ZEPRS, Hospital OS, Mirth, OSCAR McMaster, THIRRA, HOSxP, VistA.

За время работы была предпринята попытка адаптации системы OpenMRS к задачам проекта, а именно необходимо было исследовать архитектуру, русифицировать, изменить и доработать пользовательский интерфейс, а также некоторые компоненты в архитектуре и базе данных. Однако в результате было принято решение оставить этот подход, поскольку трудозатраты на указанные работы превышали допустимые сроки. Подробности обзора системы находятся в разделе 4.

3.3 Разработка собственного продукта

Данный подход предполагает разработку и анализ требований к системе на основе ожиданий заказчика и открытых источников посвящённым теме разработке МИС, а также проектирование архитектуры и реализацию соответствующего функционала. В результате было принято решение использовать данный подход и разрабатывать свой продукт, и далее развивать его как открытое программное обеспечение.

4 Обзор системы OpenMRS

Система OpenMRS построена на основе многослойной архитектуры (рисунок 1): пользовательский интерфейс, слой бизнес-логики и слой данных. Кроме того предусмотрен отдельный слой для модулей, позволяющих вести настройку системы и доработку необходимого функционала. За технологическую основу взята Java Enterprise Edition с использованием Spring Framework. Система имеет web-интерфейс: на клиентской стороне используется Javascript и библиотека jQuery. На серверной же стороне язык программирования – java, СУБД – MySQL, а в качестве библиотеки для объектно реляционного отображения используется Hibernate. Кроме того система предоставляет REST API для работы с объектами системы: имеются сервисы на уровне бизнес-логики позволяющие работать с пользователям системы, пациентами, врачами.

Система OpenMRS является свободно распространяемым программным обеспечением и имеет открытый исходный код: на момент написания данной работы его можно получить на следующем электронном ресурсе <https://source.openmrs.org/>

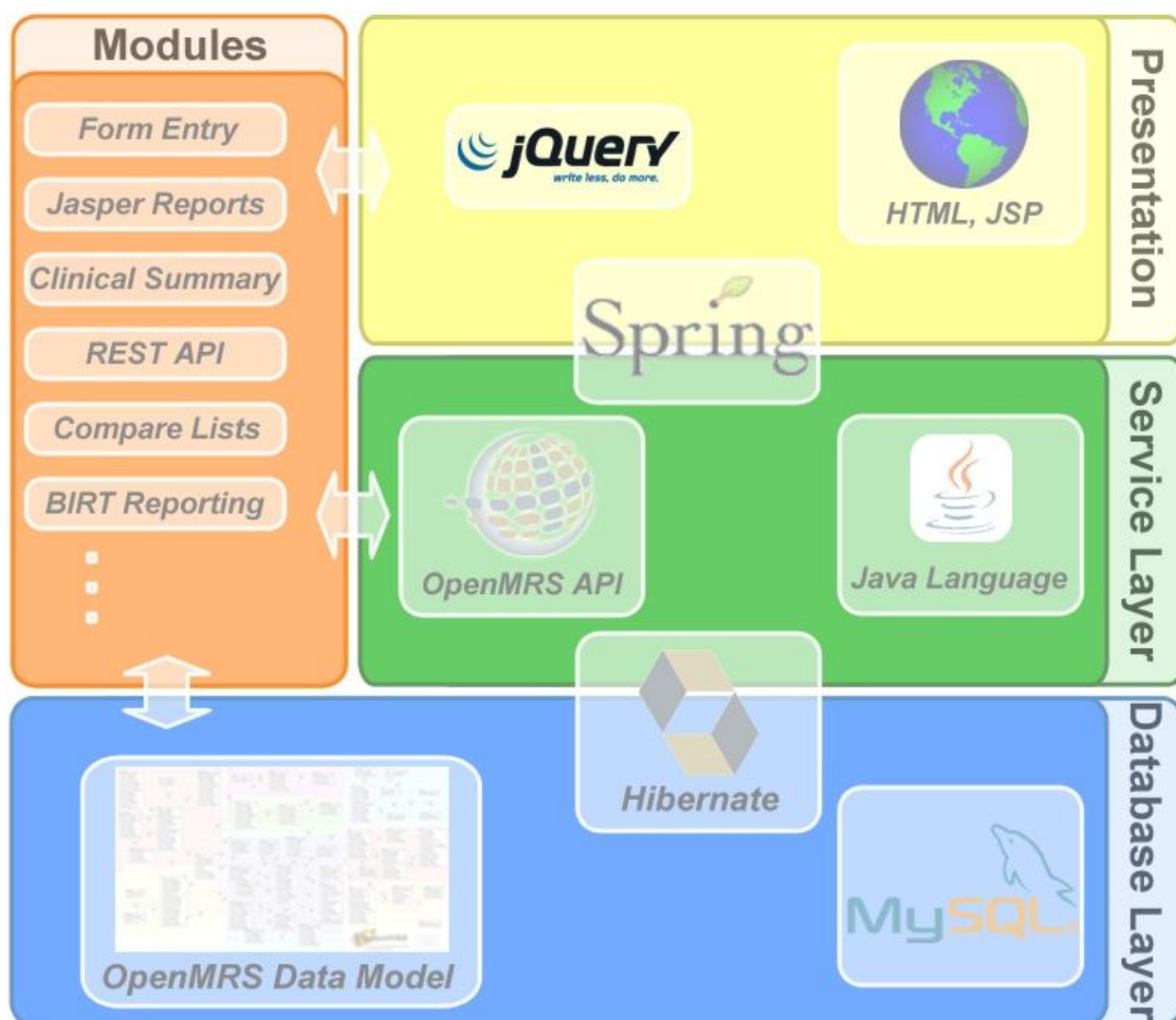


Рисунок 1. Архитектура МИС OpenMRS

5 Технологии для разработки МИС

Медицинские информационные системы де-факто должны поддерживать web-интерфейс. Поэтому современную МИС имеет смысл проектировать как web-приложение. Для такого рода задач широко применяется архитектура MVC (Module-View-Controller). На сегодняшний день имеется множество платформ реализующих данный подход. Наиболее известные из них:

- Ruby on Rails
- Play Framework
- Apache Tapestry
- ASP.NET MVC
- Spring Framework MVC
- Grails

Все они предоставляют более менее схожий функционал для разработки полноценного MVC-приложения. Отличаются они в основном языками программирования: для Ruby on Rail – это Ruby, для ASP.NET MVC – C#, Spring Framework MVC и Apache Tapestry – Java, Play Framework – Scala и Java, Grails – Groovy. Исходя из этого, можно обобщить технологические аспекты вышеуказанных платформ, которые, которые необходимы для разработки современной МИС:

- Шаблоны web-страниц на серверной стороне. Большинство современных платформ поддерживают использование шаблонов для разработки пользовательского интерфейса. Основная идея заключается в их переиспользовании для быстрого создания новых элементов на уровне View (представление в модели MVC) и динамической генерации web-страниц. Для платформ поддерживающих стек технологий Java EE– это java server pages, для ASP.NET active server pages, для Grails - groovy server pages, для Play Framework – это scala шаблоны. Соответственно все вышеуказанные платформы имеют такой функционал в том или ином виде: присутствуют лишь различия в синтаксисе.
- Интеграция с современными СУБД. Возможность работы с реляционными базами данных (PostgreSQL, MS SQL, MySQL, H2SQL и др.), а также с NoSQL решениями (MongoDB, OrientDB и др.).

- Интеграция с облачными сервисами. Возможность разместить разрабатываемое приложение в облаке, воспользовавшись сервисами PaaS. Примерами могут быть Jelastic и Heroku – для java-приложений, или Microsoft Azure – для ASP.NET.
- Кэширование. Для ускорение работы web-приложения можно сохранять результаты работы как на клиентской стороне, так и на серверной. Современные MVC-платформы поддерживают такой механизм. Скажем, для ASP.NET – это реализуется при помощи атрибутов, устанавливающих параметры хранения кэша, а для Play Framework – это реализуется через java-аннотации, причём доступ к закэшированным данным возможен непосредственно через API платформы.
- Базовые элементы безопасности для web-приложения на уровне платформы: экранирование специальных символов при работе с web-страницами, ассоциирование ключа безопасности с сессиями пользователей для предотвращения несанкционированного доступа.
- Возможность конфигурации маршрутизации запросов на web-сервер с разделением на POST и GET. Например, в Play Framework для этого отдельно настраивается конфигурационный файл, где явно указывается путь, вид запроса и используемый контроллер.

6 Прототип медицинской информационной системы

В данной главе будут рассмотрены функциональные требования, предъявляемые к электронной регистратуре, архитектура всей системы, элементы безопасности и пользовательский интерфейс системы.

6.1 Функциональные требования к электронной регистратуре

Основным пользователем системы является медицинский персонал, отвечающий за работу регистратуры. На рисунке 2 изображена диаграмма модели прецедентов, где отображаются основные функциональные требования к электронной регистратуре.

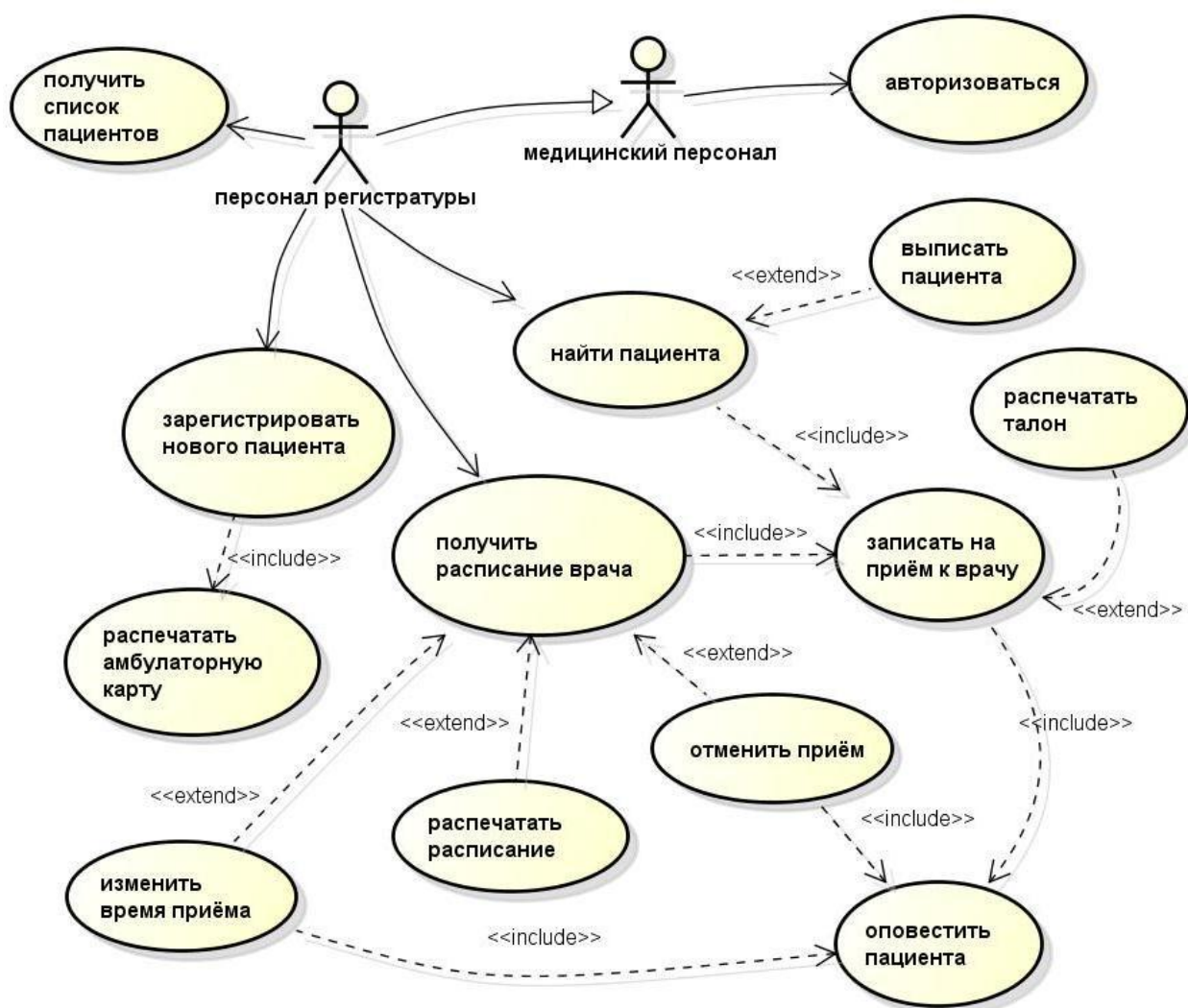


Рисунок 2. Основные прецеденты в работе электронной регистратуры

В целом, все функциональные требования, предъявляемые к электронной регистратуре, можно разделить на несколько категорий:

- **Регистрация пациентов.** Регистрация и хранение персональных данных обсуживаемых пациентов. Поиск гражданина по идентификатору (номеру студенческого, номер полиса обязательного медицинского страхования, и другие). Учет льготников и инвалидов. Автоматизированное ведение реестра льготников и инвалидов, закрепленных за ЛПУ. Возможность создания амбулаторной карты, истории болезни, талона амбулаторного пациента. Возможность внесения документов, удостоверяющих личность.
- **Функции печати.** Печать расписания врача/участка на день с записями пациентов. Печать медицинских документов: статистический талон амбулаторного пациента (форма 025-12/у), медицинская карта амбулаторного пациента (форма № 025/у-04), талон для пациента (напоминание о времени и месте приёма), контрольная карта диспансерного наблюдения. (форма № 030/у-04), направление на госпитализацию, восстановительное лечение, обследование, консультацию (форма № 057/у-04). Печать информированного согласия на обработку персональных данных. Печать списка пациентов на участок или к врачу за некоторый период времени (день, неделя, месяц).
- **Ведение расписания врачей.** Автоматизированное ведение расписания работы врачей. Запись пациентов на приемы врача. Гибкие функции индивидуальной настройки календаря для каждой специальности врача. Поддержка различных справочников видов приема (первичный прием, повторный прием, консультация). Встроенная функция статистического учета информации о направившем враче (отделении). Возможность автоматизированного переноса расписания (например, в случае болезни врача) с соответствующим механизмом оповещения пациентов.
- **Поддержка статистических отчетов.** Статистика обратившихся в регистратуру пациентов. Статистика сделанных записей на приём, их соотношение к выписанным пациентам. Автоматическое составление отчетности по выполнению функции врачебной должности.
- **Ведение сайта ЛПУ.** Удаленное управление сайтом через web-интерфейс. Публикация различных статей, материалов, новостей. Поддержка RSS.
- **Оповещение пациентов.** Оповещение о записи пациента по электронной почте и при помощи отправления SMS.

6.2 Архитектура

Как уже было сказано в главе 5, медицинские информационные системы де-факто должны поддерживать web-интерфейс. В связи с этим, решено было проектировать систему как web-приложение. Соответственно в основу легла архитектура MVC, которая широко применяется в таких задачах. Технологической платформой является Play Framework – фреймворк, который предназначен для разработки web-приложений, обладающих высокой степенью масштабируемости и интероперабельности. В качестве СУБД используется PostgreSQL. Языки программирования для реализации функционала на стороне сервера – Java, Scala и инструментарий для объектно-реляционного отображения Ebean, а для разработки клиентской стороны – Javascript с соответствующим набором современных библиотек: jQuery, Twitter Bootstrap. На рисунке 3 изображена архитектура приложения.

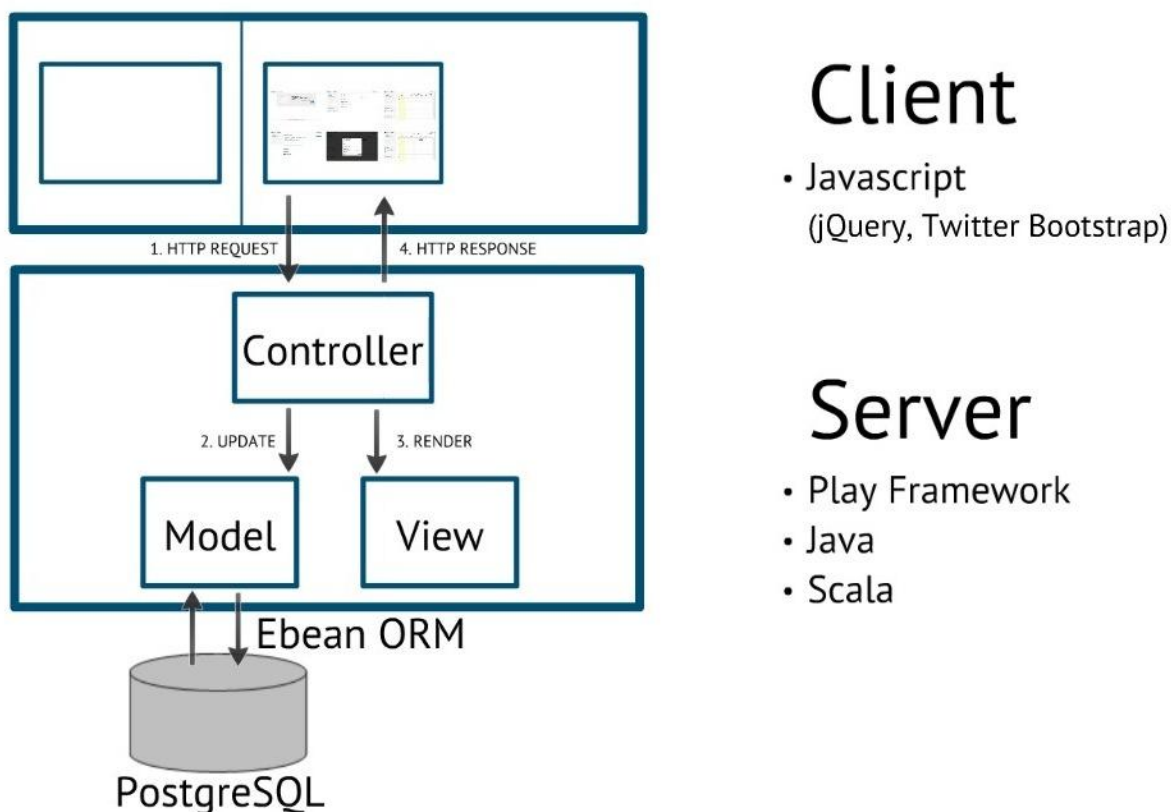


Рисунок 3. Архитектура системы

Структура данных электронной регистратуры базируется на ряде объектов с уровня бизнес-логики приложения, которые при помощи инструментария объектно-реляционного отображения Ebean преобразовываются в схему базы данных. К таким объектам можно отнести персону, пациента, врача, и календарь. На рисунке 4 представлена часть объектной модели.

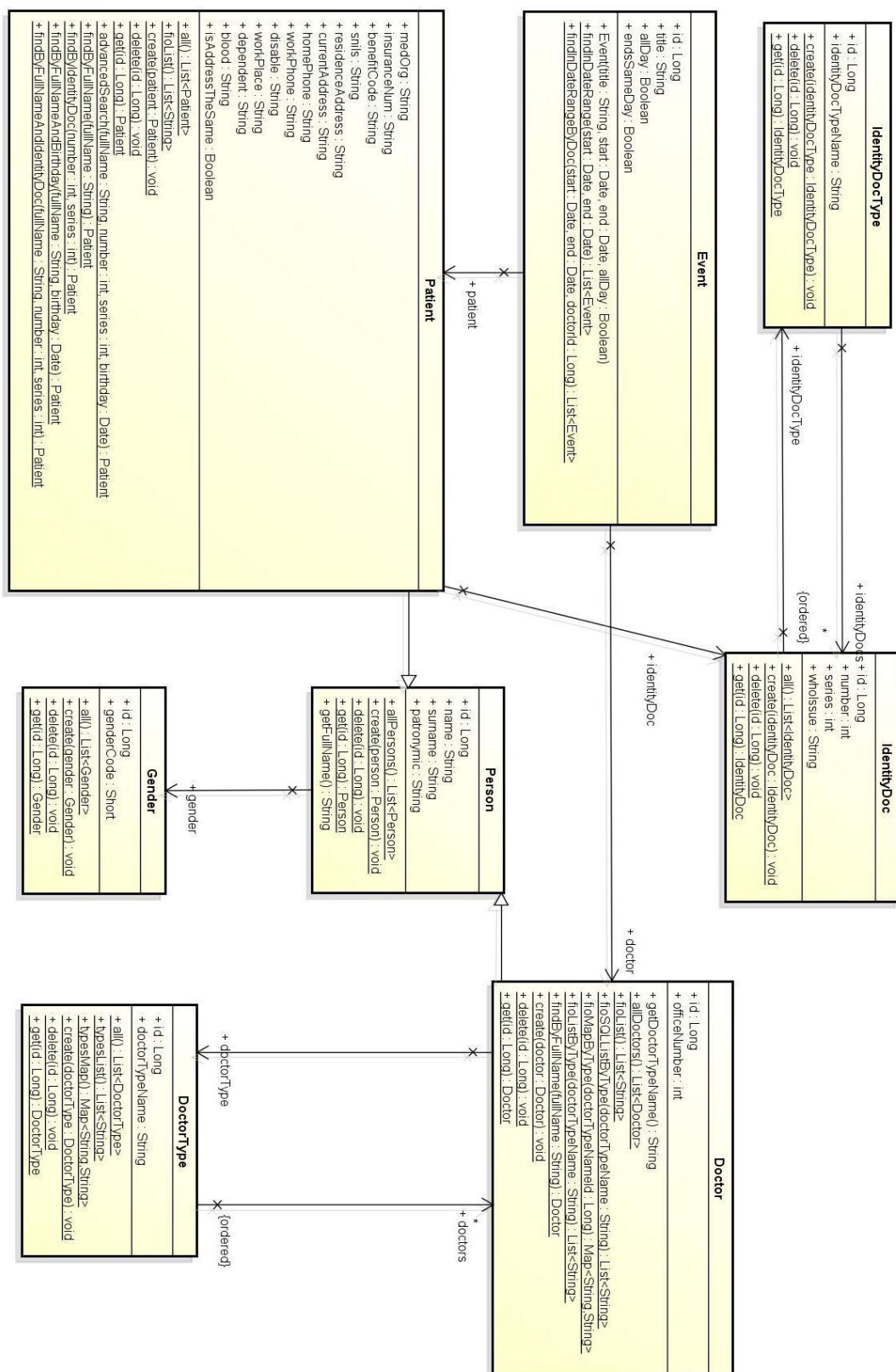


Рисунок 4. Часть объектной модели электронной регистратуры

На рисунке 5 изображено два основных контроллера, отвечающих за работу с пациентами и расписанием врачей на уровне бизнес-логики модуля электронной регистратуры.

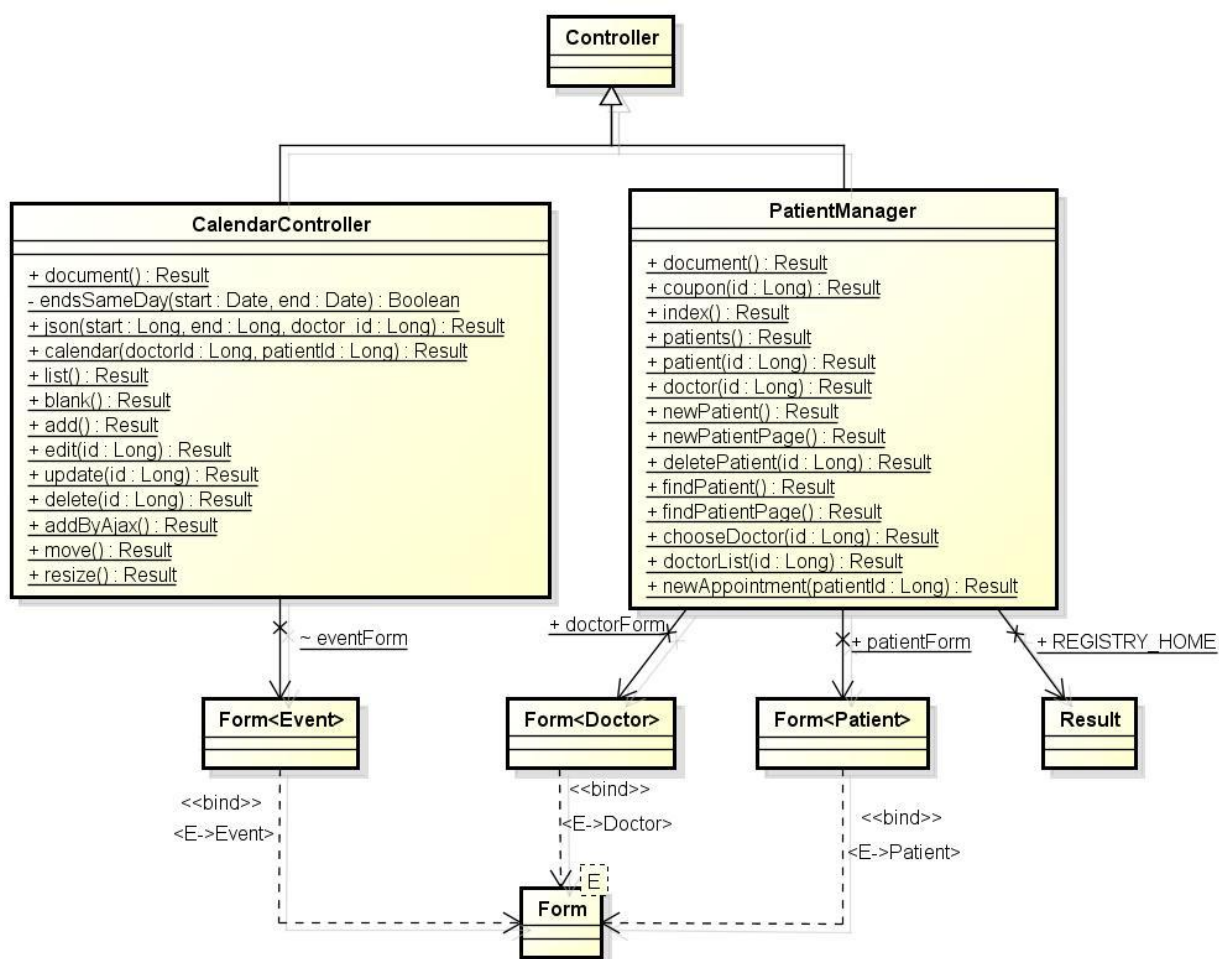


Рисунок 5. Основные контроллеры электронной регистратуры

Рассмотрим взаимодействие системы и пользователя на примере прецедента записи пациента на приём к врачу. Для того чтобы записать пациента на приём, вводится его фамилия, имя и отчество: на основе этого он идентифицируется в системе. Далее выбирается специальность врача, предполагаемая дата приёма и конкретный медицинский специалист (рисунок 6).

Главная страница

Регистратура

Артём Воронов

ИНСТРУМЕНТЫ

Запись на приём

Создание нового пациента

в разработке

Записанные пациенты

Все пациенты

Амбулаторная карта

Расписание врача

Создать шаблон

Статистика

Пациент

Иванов Иван Иванович

Дата рождения: 19 янв 1978

Адрес: 630124, Россия, Новосибирская область, г. Новосибирск, ул. Пирогова 2, 121

Запись на приём

Иванов Иван Иванович

Специальность врача
Терапевт

Предполагаемая дата приёма
11.04.2013

Свободные врачи
Кошкаркина Ольга Петровна

Выбрать

Отмена

Рисунок 6. Запись на приём к врачу

Далее предполагается получить расписание врача, для того чтобы сделать в нём новую запись. Для этого нажимается кнопка “Выбрать”, после чего формируется AJAX-запрос и запускается следующий сценарий. Контроллер, отвечающий за записи в календаре врача, получает запрос, и инициализирует объектную модель при помощи объектно-реляционного отображения (рисунок 7). В результате мы получаем все события из календаря указанного врача за некоторый период времени. Далее используется шаблон, который заранее был создан на серверной стороне для графического отображения календаря, и заполняется данными событиями. И уже после этого на сторону клиента возвращается результат (рисунок 8). Применение AJAX позволяет на клиентской стороне инициировать другие сценарии и взаимодействовать с системой в то время пока выполняется данная операция.

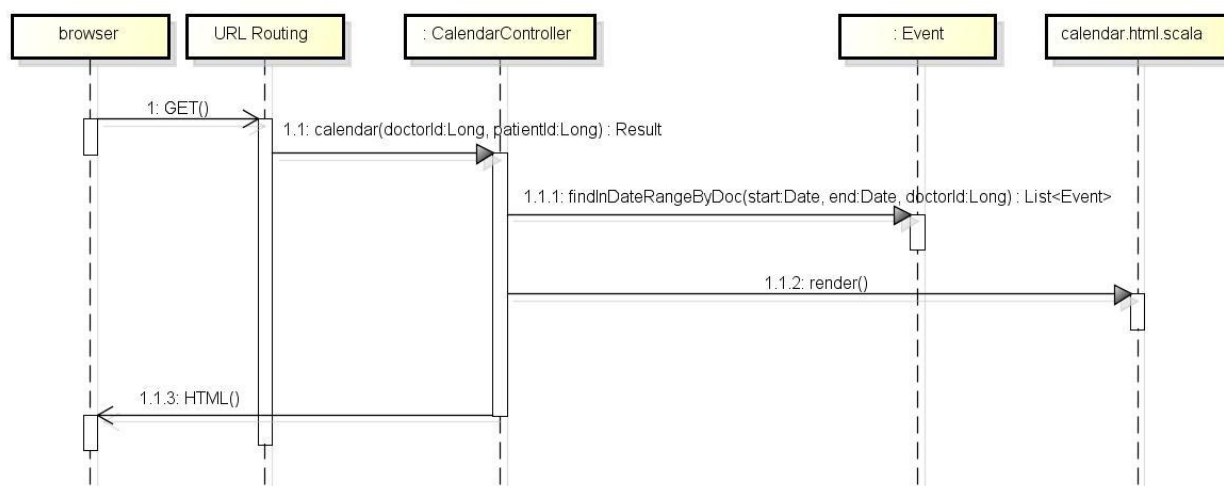


Рисунок 7. Реализация прецедента записи на приём к врачу

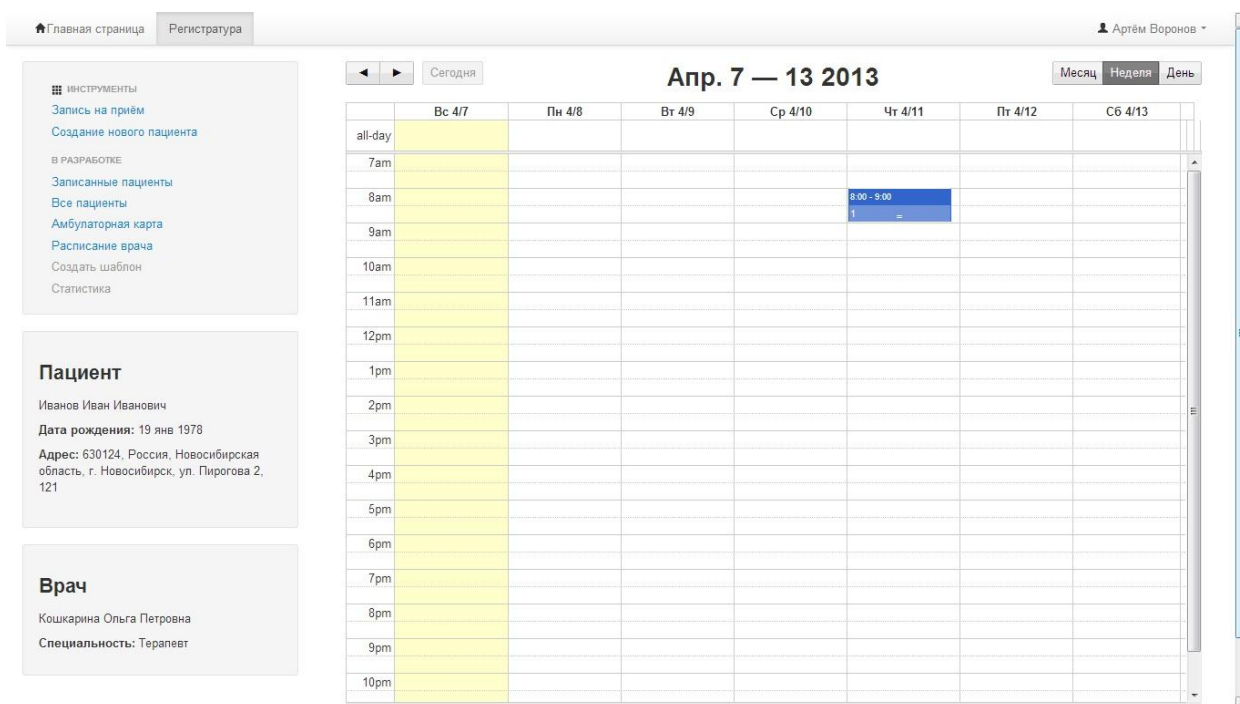


Рисунок 8. Расписание врача

Всё остальное взаимодействие пользователя и системы выглядит аналогично: асинхронные запросы, взаимодействие контроллеров, объектной модели и шаблонов на серверной стороне.

6.3 Элементы безопасности

Главным принципом при разработке прототипа МИС считалось обеспечение конфиденциальности. Модель угроз включает в себя все широко известные уязвимости web-приложений. Основными из них являются: межсайтовый скриптинг (Cross Site Scripting, XSS), подделка межсайтовых запросов (Cross Site Request Forgery, CSRF) и SQL-инъекции.

В системе применены следующие меры предосторожности для обеспечения защиты от вышеперечисленных угроз:

1. Для борьбы с SQL-инъекциями все запросы в базе данных выполняются через параметризованные запросы, а не через простую конкатенацию. В качестве примера может служить выборка пациента по номеру и серии документа удостоверяющего личность: на рисунке 9 используются параметры при составлении запроса, а не конкатенация строки sql и аргументов метода.

```

1 public static Patient findByIdentityDoc(int number, int series) {
2     String sql = "select * from patient
3                 left join identity_doc on patient.identity_doc_id = identity_doc.id
4                 where number=:number and series=:series";
5     SqlQuery sqlQuery = Ebean.createQuery(sql)
6         .setParameter("number", number)
7         .setParameter("series", series);
8     ...
9 }

```

Рисунок 9. Параметризованный SQL-запрос

2. Для борьбы с межсайтовым скриптингом используется экранирование спецсимволов HTML, которое автоматически происходит когда работает модуль Play Framework, отвечающий за работу шаблонами Scala на серверной стороне.
3. Для борьбы с подделкой межсайтовых запросов используется механизм, при котором с каждой сессией пользователя ассоциируется дополнительный секретный ключ, предназначенный для выполнения запросов. В рамках разработанного прототипа этот ключ представляет строку длиной в 64 символа зашифрованную при помощи криптосистемы AES. Кроме того, платформа Play Framework имеет API для генерации таких ключей.

6.4 Пользовательский интерфейс

Поскольку система представляет собой web-приложение, то всё взаимодействие пользователей с ней осуществляется через web-браузер. В зависимости от роли пользователя система предоставляет соответствующий графический интерфейс. Точка входа в систему показана на рисунке 10 – здесь пользователю необходимо ввести логин и пароль. Кроме того, данная web-страница позволяет получить доступ к исходному коду проекта, его описанию и разделу с контактами разработчиков (рисунки 11 и 12).

Рисунок 10. Web-страница для авторизации в системе

Рисунок 11. Описание проекта

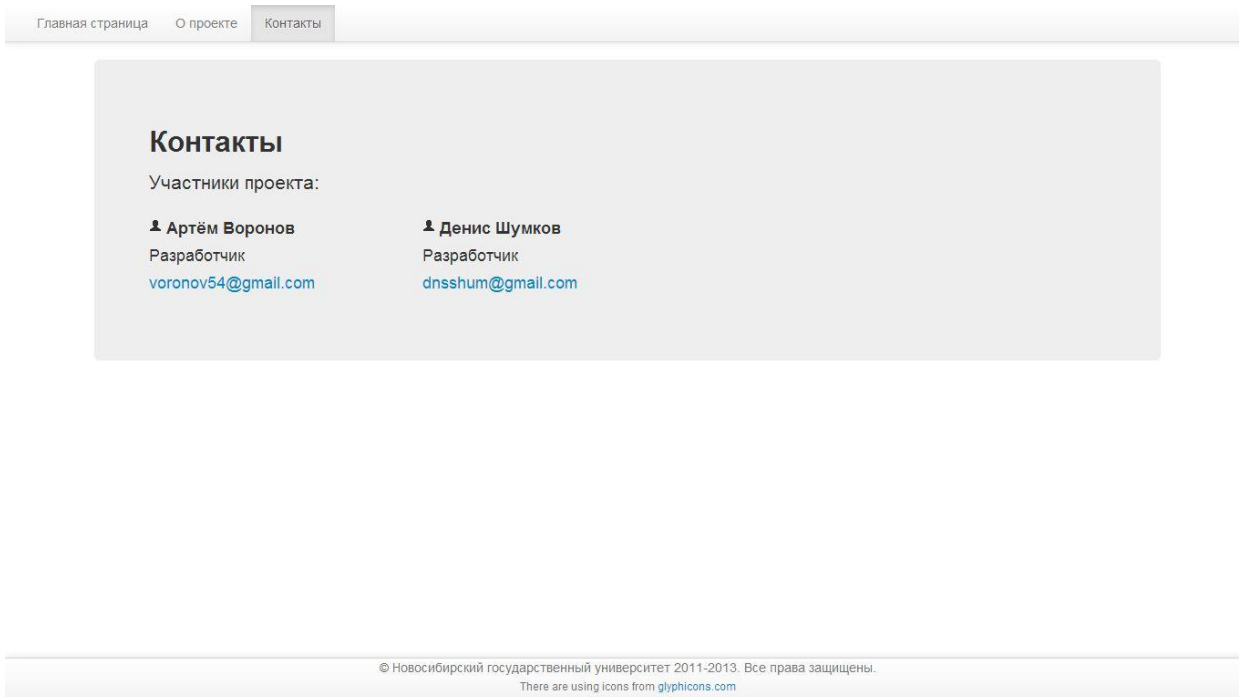


Рисунок 12. Контакты разработчиков

Персоналу регистратуры после авторизации в системе становятся доступен соответствующий функционал: на рисунке 13 представлена web-страница, где можно вести поиск пациента по фамилии, имени и отчеству. Для удобства пользователя была реализована автоподстановка в строку поиска на основе зарегистрированных в системе пациентов (рисунок 14), для появления подсказки нужно начать вводить данные.

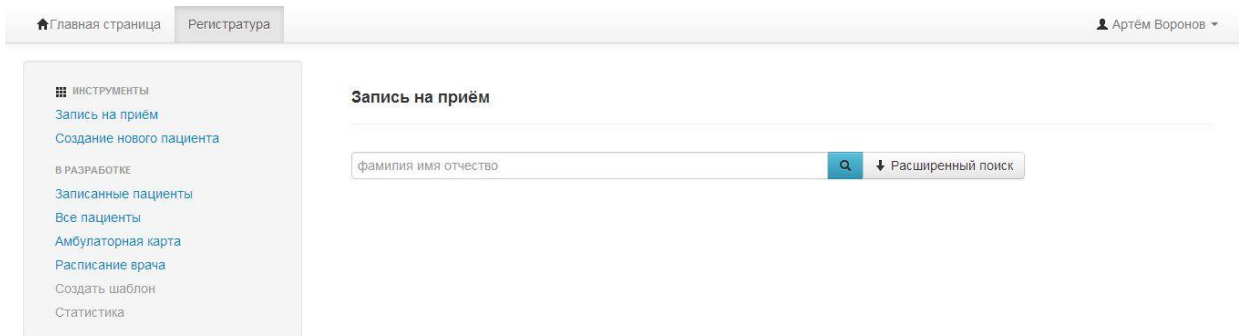


Рисунок 13. Поиск пациента

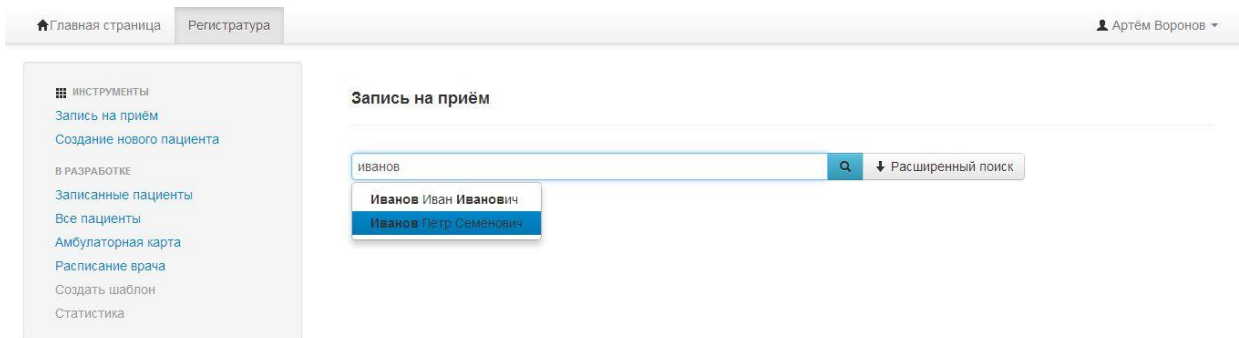


Рисунок 14. Автозаполнение строки поиска из базы данных

Кроме того, в системе реализован расширенный поиск пациента, предполагающий использование других характеристик. В текущей версии системы – это дата рождения, номер и серия документа удостоверяющего личность (рисунки 15 и 16).

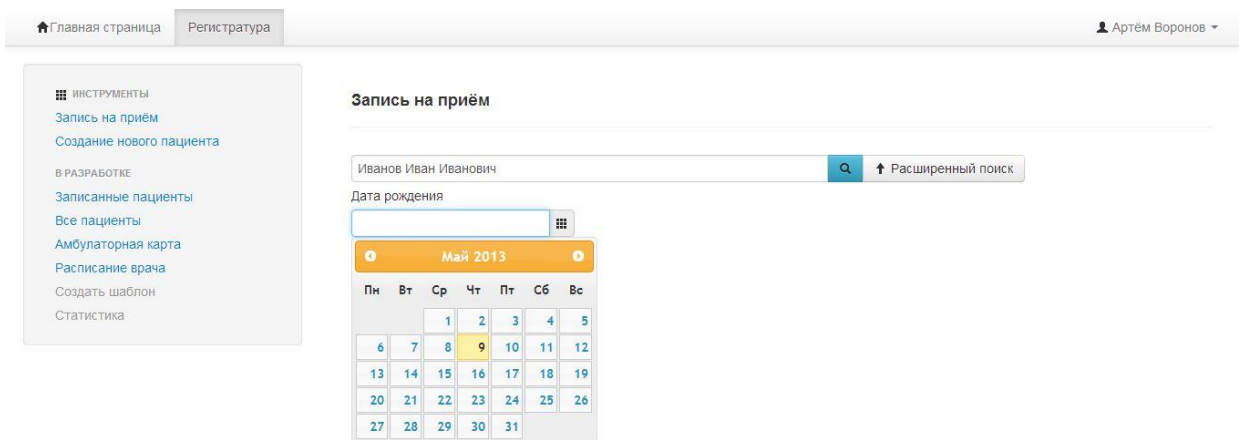


Рисунок 15. Расширенный поиск по дате рождения

Главная страница Регистратура Артём Воронов ▾

ИНСТРУМЕНТЫ
Запись на приём
Создание нового пациента

в РАЗРАБОТКЕ
Записанные пациенты
Все пациенты
Амбулаторная карта
Расписание врача
Создать шаблон
Статистика

Запись на приём

Иванов Иван Иванович 🔍 ↑ Расширенный поиск

Дата рождения
08.05.2013 📅

Номер
6653

Серия
141241

Рисунок 16. Расширенный поиск по номеру и серии

В приложении А приводятся рисунки с остальными элементами пользовательского интерфейса электронной регистратуры: работа с расписанием врача, регистрация пациентов и их запись на приём, просмотр назначенных приёмов, печать талона.

ЗАКЛЮЧЕНИЕ

В ходе работы были достигнуты следующие результаты:

1. Был произведен обзор предметной области, который включает в себя результаты исследования задачи разработки МИС для ЛПУ и существующих подходов к её решению. Подробности обзора находятся в главах 1 и 3. Также в рамках обзора предметной области была исследована система OpenMRS. Результаты находятся в главе 4.
2. На основе результатов исследования предметной области и требований заказчика были сформулированы требования к системе: в разделе 6.1 представлен основной функционал электронной регистратуры.
3. Были проведены исследования технологий, необходимых для создания современной информационной системы для медицинского учреждения. В результате была выбрана платформа Play Framework для реализации прототипа МИС. Описание результатов исследования приводится в главе 5.
4. Была спроектирована архитектура на основе MVC и реализован функционал электронной регистратуры ведения расписания врачей, работы с пациентами (регистрация, запись на приём к врачу, поиск по идентификатору, хранение персональных данных), функции печати. Подробное описание находится в разделе 6.2.
5. Была разработана модель угроз и были приняты меры предосторожности для обеспечения защиты от основных уязвимостей web-приложений: межсайтовый скриптинг (Cross Site Scripting, XSS), подделка межсайтовых запросов (Cross Site Request Forgery, CSRF) и SQL-инъекции. Подробности находятся в разделе 6.3.
6. Был спроектирован и разработан пользовательский интерфейс на основе библиотек Twitter Bootstrap и jQuery. Описание приводится в главе 6.4, приложении А, а также в разделе 6.3.
7. Была решена задача подготовки площадки для дальнейшего развития системы: поскольку проект выполнялся в рамках подхода по разработке свободного программного обеспечения, то исходный код полученного приложения был размещен в репозитории GitHub и находится в публичном доступе. Дальнейшее развитие продукта также будет происходить на базе этого сервиса: https://github.com/ArtemVoronov/mis_nsu.

Результаты данной работы были представлены на научно-студенческой конференции МНСК-2013 в секции Информационные технологии, в результате которой автор работы получил диплом 2 степени [10].

ПЕРЕЧЕНЬ УСЛОВНЫХ ОБОЗНАЧЕНИЙ

Таблица 1. Обозначения и определения

| Термин | Описание |
|--------|--|
| МИС | Медицинская информационная система |
| ЛПУ | Лечебно-профилактическое учреждение |
| АРМИТ | Ассоциация развития медицинских информационных технологий |
| ЭР | Электронная регистратура |
| СМО | Страховая медицинская организация |
| SMS | Short Message Service. Технология, позволяющая осуществлять приём и передачу коротких текстовых сообщений с помощью сотового телефона. К настоящему времени входит в стандарты сотовой связи. |
| ЭИБ | Электронная история болезни. Информационная система, предназначенная для ведения, хранения на электронных носителях, поиска и выдачи по информационным запросам (в том числе и по электронным каналам связи) персональных медицинских записей |
| ЭАК | Электронная амбулаторная карта |
| ПМЗ | Персональная медицинская запись. Любая запись, сделанная конкретным медицинским работником в отношении конкретного пациента |
| ЭПМЗ | Электронная персональная медицинская запись. Любая персональная медицинская запись, сохраненная на электронном носителе. Понятие электронная персональная медицинская запись соответствует международному термину EHR - Electronic Health Record [7]. |
| ЭМА | Электронный медицинский архив. электронное хранилище данных, содержащее ЭПМЗ и другие наборы данных и программ (классификаторы и справочники, списки пациентов и сотрудников, средства навигации, поиска, визуализации, интерпретации, проверки целостности и ЭЦП, распечатки ЭПМЗ и др.). |
| MVC | Module-View-Controller. Архитектура модель-представление-контроллер, широко применяемая при разработке web-приложений. |

ЛИТЕРАТУРА

1. Гусев А. В., Рынок медицинских информационных систем: обзор, изменения, тренды, журнал "Врач и информационные технологии", №3 2012, с. 6-15
2. Мингалеев Г. Ф., Проблемы применения IT-технологий в медицине, Сборник научных трудов всероссийской медицинской научно-практической конференции "Развитие российского здравоохранения на современном этапе" / Министерство здравоохранения Мурманской области — М.: Издательство "АдамантЪ", 2013, с. 66-71
3. Запольская М. М., Комплексная медицинская информационная система для врачей узких специальностей, Сборник научных трудов научно-технической конференции "Перспективные информационные технологии в научных исследованиях, проектировании и обучении" / Издательство СНЦ РАМН - Самара, 2012, с. 277-279
4. Кобринский А. А., Медицинская информатика: учебник для студ. ВУЗов. – М.: Издательский центр «Академия», 2009, с. 132-139
5. Гусев А. В., Рекомендации по выбору медицинской информационной системы, журнал "Менеджер здравоохранения", №5, 2010, с. 38-45
6. ГОСТ Р 52636-2006. Электронная история болезни. Общие положения. Национальный стандарт Российской Федерации. // Стандартинформ, М., 2006.
7. Международный стандарт ISO/TR 20514:2005, Информационные технологии в медицине – Электронная медицинская запись – Определение, назначение, содержание (Health informatics - Electronic health record - Definition, scope and context)
8. ISO/IEC 5218. Информационные технологии. Коды для представления пола человека.
9. Гусев А. В., Эльянов М. М. Методические рекомендации на разработку и внедрение информационной системы «Электронная регистратура (приемное отделение)» [Электронный ресурс, 04.04.2011], К-МИС, 2011, с. 8-23, режим доступа: http://www.armit.ru/news/indexmerop.php?ELEMENT_ID=471
10. Воронов А. В., Разработка медицинской информационной системы НГУ// Материалы 51-й международной научной студенческой конференции «Студент и научно-технический прогресс», Информационные технологии, Новосибирск, 12-18 апреля 2013 г. – с. 158.

Приложение А. Элементы пользовательского интерфейса

The screenshot shows the 'Создание нового пациента' (Create new patient) form. The interface includes a top navigation bar with 'Главная страница' and 'Регистратура', and a user profile 'Артём Воронов' with a dropdown menu containing 'Настройки', 'Помощь', and 'Выйти'. A left sidebar lists various tools under 'ИНСТРУМЕНТЫ', including 'Запись на приём', 'Создание нового пациента', and others. The main form has four tabs: 'Персональные данные' (selected), 'Страховка и льготы', 'Контактная информация', and 'Место работы'. The 'Персональные данные' tab contains fields for 'Адрес постоянного места жительства' and 'Адрес регистрации по месту пребывания', both with a checkbox 'Адреса совпадают'. Below these are fields for 'Телефон домашний' and 'Телефон рабочий'. At the bottom are buttons for 'Далее', 'Назад', and 'Отмена'.

Главная страница Регистратура Артём Воронов

ИНСТРУМЕНТЫ
Запись на приём
Создание нового пациента
В РАЗРАБОТКЕ
Записанные пациенты
Все пациенты
Амбулаторная карта
Расписание врача
Создать шаблон
Статистика

Создание нового пациента

Персональные данные Страховка и льготы Контактная информация Место работы

Адрес постоянного места жительства
Область, район, населенный пункт, улица, дом, корпус, квартира

☐ Адреса совпадают

Адрес регистрации по месту пребывания
Область, район, населенный пункт, улица, дом, корпус, квартира

Телефон домашний
+7 (383) 363-40-25

Телефон рабочий
+7 (383) 363-40-26

Далее Назад Отмена

Рисунок А.1. Регистрация нового пациента

The screenshot shows the 'Запись на приём' (Appointment) form. The interface is similar to the previous one, with the same top navigation bar and left sidebar. The main form has a search bar for the patient's name, currently showing 'Иванов Иван Иванович'. Below the search bar are dropdown menus for 'Специальность врача' (Therapist) and 'Предполагаемая дата приёма' (11.04.2013). There is also a dropdown for 'Свободные врачи' (Koшкарina Oльга Петровна). At the bottom are buttons for 'Выбрать' and 'Отмена'. A 'Пациент' (Patient) sidebar on the left displays the patient's details: 'Иванов Иван Иванович', 'Дата рождения: 19 янв 1978', and 'Адрес: 630124, Россия, Новосибирская область, г. Новосибирск, ул. Пирогова 2, 121'.

Главная страница Регистратура Артём Воронов

ИНСТРУМЕНТЫ
Запись на приём
Создание нового пациента
В РАЗРАБОТКЕ
Записанные пациенты
Все пациенты
Амбулаторная карта
Расписание врача
Создать шаблон
Статистика

Пациент
Иванов Иван Иванович
Дата рождения: 19 янв 1978
Адрес: 630124, Россия, Новосибирская область, г. Новосибирск, ул. Пирогова 2, 121

Запись на приём

Иванов Иван Иванович

Специальность врача
Терапевт

Предполагаемая дата приёма
11.04.2013

Свободные врачи
Кошкарina Oльга Петровна

Выбрать Отмена

Рисунок А.2. Запись на приём к врачу

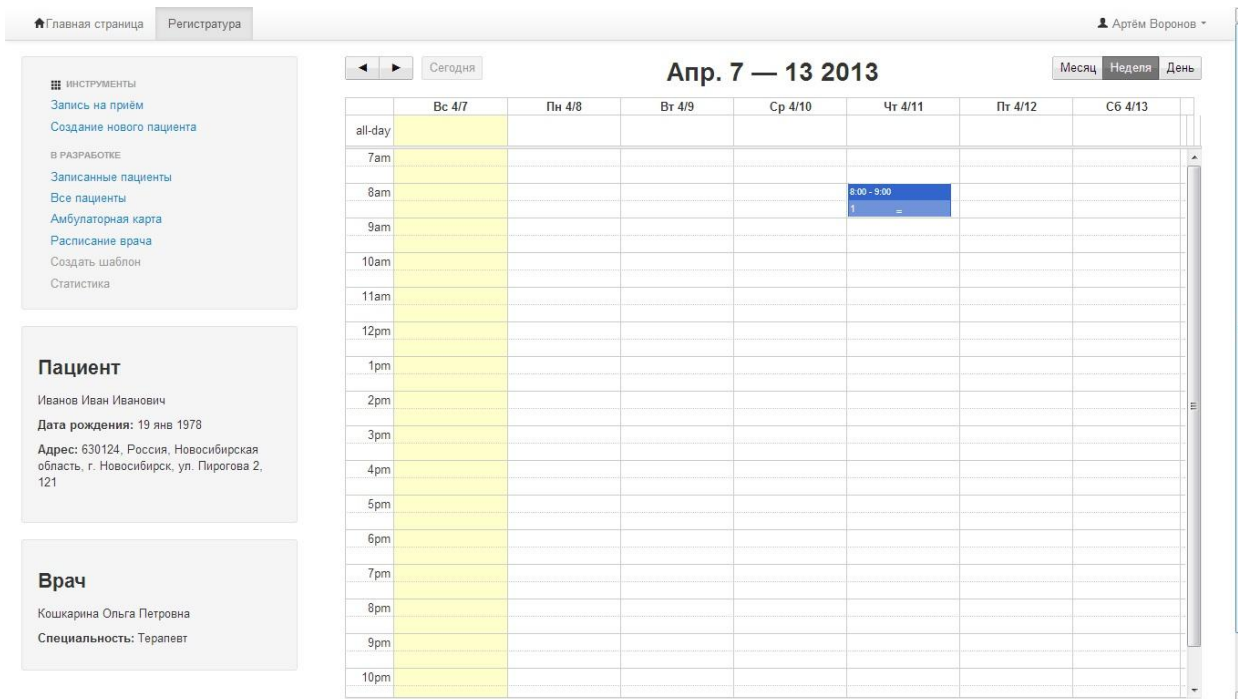


Рисунок А.3. Работа с расписанием врача

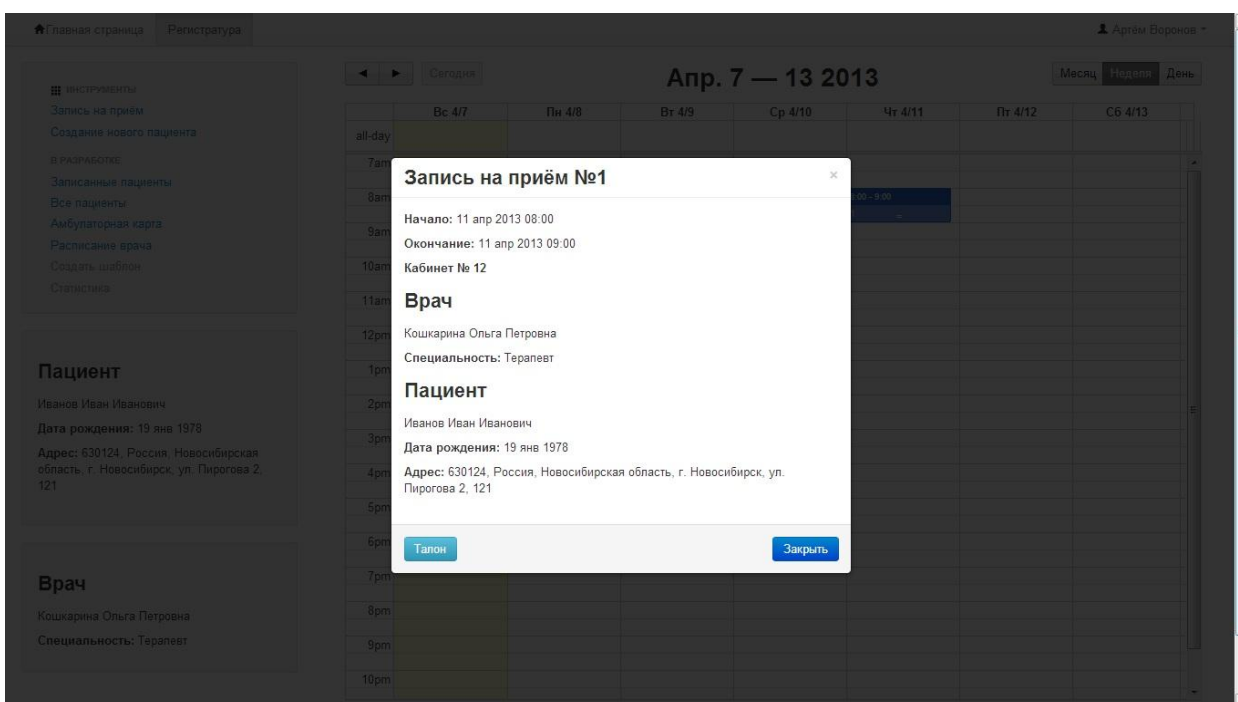


Рисунок А.4. Зарегистрированный приём к врачу