**Министерство образования Иркутской области**

Государственное бюджетное профессиональное

образовательное учреждение Иркутской области

«Иркутский авиационный техникум»

(ГБПОУИО «ИАТ»)

КР.09.02.07.22.201.11 ПЗ

ИНФОРМАЦИОННАЯ СИСТЕМА

«БОЛЬНИЦА. РЕГИСТРАТУРА»

|  |  |  |
| --- | --- | --- |
| Председатель ВЦК: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Кудрявцева) |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Кудрявцева) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (А.В. Макрицкий) |
|  |  |  |

Иркутск 2022

**Введение**

Регистратура больницы является одним из важнейших структурных подразделений медицинского учреждения. В данный момент запись на прием к врачу является проблематичной из-за сложности работы с бумажными носителями, а также недостатка легких и понятных информационных средств регистрации. Компьютерные технологии автоматизируют информационные процессы лечебного учреждения, тем самым процесс регистрации становится более простым.

В связи с этим, возникла необходимость в разработке такой информационной системы как программа для записи и хранения данных о пациенте, которая должна обеспечить поиск все о пациенте, какого врача он выбрал и др.

Цель курсовой работы – разработать информационную систему «Регистратура. Больница».

Задачи курсовой работы:

Провести предпроектное исследование предметной области (выбрать тему, поставить цель и задачи, описать предметную область).

Разработать техническое задание.

Спроектировать программный продукт.

Разработать и отладить программный продукт.

Составить программную документацию в виде руководства пользователя.

Актуальность разрабатываемой программы заключается в том, что теперь каждый желающий может прийти в больницу, подойти к электронной регистратуре и воспользоваться ей. Компьютеры помогают человеку в работе, развлечении, образовании, в научных исследованиях и др. Программа для записи данных о пациенте «Регистратура больницы» позволит исключить многолюдную очередь в регистратуру.

1. **Описание предметной области ИС**

Предметной областью курсовой работы является регистратура в больнице. Регистратура - структурное подразделение медицинского учреждения, непосредственно осуществляющее медицинскую деятельность, по формам и технологиям здравоохранения.

Регистратура любой поликлиники ведет учет и регистрацию населения, проживающего на участке, обсуживающимся данной поликлиникой. Оформляет первичный прием, заполняет больничные карточки, отслеживает ведение историй болезни и посещение больными узких специалистов, оформляет и выдает талоны на прием к узким специалистам. Формирует и отслеживает расписание работы узких специалистов. Выдает результаты анализов и больничные карточки посетителям поликлиники. Ведет архив больничных карточек.

В настоящее время работа с пациентами в регистратуре поликлиники ведется в ручном режиме, что приводит к потере времени и ненадлежащему контролю.

Выбранная тема исследования является актуальной в связи с внедрением автоматизированной системы регистрации пациентов, что значительно упростит и ускорит процесс обслуживания пациентов.

Информация, хранимая в базе данных информационной системы регистратуры:

* Сведения об участках.
* Информация о врачах.
* Сведения о приемах.
* Информация о пациентах (карточка пациента).
* Расписание работы врачей.
* Сведения об участках.

Один врач может обслуживать несколько участков. Информация об участке должна содержать:

* номер участка;
* расписание работы участковых врачей;
* информацию о врачах, которые обслуживают участки.

Информация о пациенте хранится в его карточке. Карточка имеет номер. В карточке указывается:

* Фамилия.
* Имя.
* Отчество пациента.
* Возраст.
* Пол.
* Домашний адрес.
* Страховой полис.

Расписание работы врачей. В расписании работы врачей указывается:

* на каком участке работает врач.
* дни и часы приема.
* номер кабинета.
* специальность.

## **Анализ инструментов, используемых в разработке программного продукта**

Грамотно выбранные инструменты для создания программного продукта, заранее определяют его результат.

Проектировать структуру информационной системы удобно через MySQL Workbench и Draw.io, а дизайн – через сервис Figma. Информационная система будет серверной, для работников. Для реализации клиентской части отлично подойдут следующие инструменты: HTML5, CSS3 и JS, включая AJAX.

MySQL Workbench — это унифицированный визуальный инструмент для архитекторов баз данных, разработчиков и администраторов баз данных. MySQL Workbench предоставляет моделирование данных, разработку SQL и комплексные инструменты администрирования для настройки сервера, администрирования пользователей, резервного копирования и многого другого.

Draw.io — это бесплатное кроссплатформенное программное обеспечение для рисования графиков с открытым исходным кодом, разработанное на HTML5 и JavaScript. Его интерфейс можно использовать для создания таких диаграмм, как блок-схемы, каркасы, UML-диаграммы, организационные диаграммы и сетевые диаграммы.

Figma – графический редактор для создания прототипов сайтов и приложений. Над проектом одновременно могут работать несколько человек, так как можно выдать доступ на редактирование или комментирование любому. В Figma обычно создают прототипы сайтов и приложений, иллюстрации, векторную графику, рисуют элементы интерфейса. Ещё здесь создают макеты сайтов для тильды: есть возможность импортировать дизайн.

HTML – язык разметки гипертекста. Язык разметки дает браузеру необходимые инструкции о том, как отображать тексты и другие элементы страницы на мониторе. Язык HTML интерпретируется браузерами и отображается в виде документа, в удобной для человека форме.

CSS – каскадные таблицы стилей, которые используются для определения стилей (правил) оформления документов — включая дизайн, вёрстку и вариации макета для различных устройств и размеров экрана.

JavaScript – это мультипарадигменный язык программирования, который обычно применяется в качестве встраиваемого инструмента для программного доступа к различным объектам приложений. С точки зрения веб-разработки, без знаний этой технологии невозможно заниматься созданием современных интерактивных сайтов. Язык JS – это то, что «оживляет» разметку страниц (HTML) и пользовательский функционал (CMS) сайтов. С помощью этого языка реализуется возможность реакции страницы или отдельных ее элементов на действия посетителя. Сегодня JavaScript является базовым языком программирования для браузеров. Он полностью совместим с операционными системами Windows, Linux, Mac OS, а также всеми популярными мобильными платформами.

AJAX, или, более длинно, Asynchronous Javascript And Xml - технология для взаимодействия с сервером без перезагрузки страниц. За счет этого уменьшается время отклика и веб-приложение по интерактивности больше напоминает десктоп.

Веб-приложение будет содержать в себе информацию – её необходимо хранить, изменять, структурировать и использовать. Это реализуется благодаря базе данных. Были рассмотрены следующие варианты реализации СУБД:

1. MySQL;

2. MySQL Workbench;

3. PostgreSQL.

MySQL — одна из наиболее используемых систем управления базами данных. MySQL управляет реляционными базами данных, то есть такими, в которых таблицы связаны между собой. MySQL работает по принципу клиент-сервер. Компьютер пользователя (клиент) отправляет запрос. Сервер баз данных его обрабатывает и предоставляет ответ. Именно поэтому часто можно услышать понятие MySQL-сервер. Это сервер, на котором хранится база данных. Система MySQL написана на языках программирования C и C++. Для работы MySQL используется язык структурированных запросов SQL.

[MySQL Workbench](https://www.mysql.com/products/workbench/) — универсальный инструмент для работы и управления базами данных. С его помощью можно создавать и редактировать таблицы и другие объекты, управлять доступом пользователей и полноценно администрировать БД. Workbench может пригодиться разработчикам, администраторам БД и архитекторам систем.

Несмотря на название, он может работать не только с MySQL, но и с другими СУБД: Oracle Database, Microsoft SQL Server, PostgreSQL и другими. Одна из главных особенностей Workbench — визуальное моделирование. Оно позволяет увидеть все таблицы и связи между ними в наглядном виде. Это намного упрощает проектирование базы данных.

PostgreSQL — это объектно-реляционная система управления базами данных (ORDBMS), наиболее развитая из открытых [СУБД](https://blog.skillfactory.ru/glossary/subd/) в мире. Имеет открытый исходный код и является альтернативой коммерческим базам данных. СУБД позволяет гибко управлять базами данных (БД). С ее помощью можно создавать, модифицировать или удалять записи, отправлять транзакцию — набор из нескольких последовательных запросов на особом языке запросов SQL.

Для наглядности сравнения вариантов реализации базы данных была составлена таблица 1.

Таблица 1 – Сравнение средств реализации базы данных

|  |  |  |  |
| --- | --- | --- | --- |
| Название БД | MySQL | MySQL Workbench | PostgreSQL |
| Большое кол-во типов данных | + | + | + |
| Популярность | + | + | - |
| Отказоустойчивость | - | + | + |
| Не требует удаленного сервера | - | + | - |
| Простота использования | - | + | - |
| Портативность | - | + | - |

Таким образом, в качестве базы данных для будущего продукта была выбрана MySQL Workbench, так как она предоставляет весь необходимый функционал для разработки продукта, и при этом она проста в изучении и овладении.

Для взаимосвязи баз данных и северной части продукта необходимо использовать серверный язык. Для реализации этого были рассмотрены два языка программирования – Python и Php.

Python — это активно развивающийся скриптовый язык, который используют для решения большого объема самых разноплановых проблем и задач. Python пригодится в создании компьютерных и мобильных приложений, его применяют в работе с большим объемом информации, при разработке web-сайтов и других разнообразных проектов, используют в машинном обучении. Данный язык программирования используют крупные известные корпорации, такие как Spotify и Амазон (например, для анализа данных и создания алгоритма рекомендаций), YouTube и даже Walt Disney. Таким образом, Python нашел свое место в различных областях — с его помощью можно решить множество задач разной сложности. Синтаксис ядра языка минималистичен, за счёт чего на практике редко возникает необходимость обращаться к документации. Сам же язык известен как интерпретируемый и используется в том числе для написания скриптов. Недостатками языка являются зачастую более низкая скорость работы и более высокое потребление памяти написанных на нём программ по сравнению с аналогичным кодом, написанным на компилируемых языках, таких как C или C++. Недостатками языка являются использование языка в серверной разработке, только благодаря фреймворку, а также зачастую более низкая скорость работы и более высокое потребление памяти написанных на нём программ по сравнению с аналогичным кодом, написанным на компилируемых языках, таких как С или C++. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений, высокоуровневые структуры данных. Поддерживается разбиение программ на модули, которые, в свою очередь, могут объединяться в пакеты.

PHP — это распространённый язык программирования общего назначения с открытым исходным кодом. PHP специально сконструирован для веб-разработок и его код может внедряться непосредственно в HTML.

Язык PHP обладает рядом неоспоримых преимуществ:

* Высокая скорость работы и, соответственно, общая производительность ресурсов.
* Бюджетность, экономичность. Найти специалиста не представляется проблемой, стоимость написания программ на php не высока.
* Простота освоения, простой синтаксис.
* Отличная совместимость и переносимость — php-коды работают одинаково хорошо с разными платформами.
* Набор текста кода и его редактирование можно осуществлять в любом текстовом или html-редакторе.

Для наглядности сравнения языков программирования была составлена таблица 2.

Таблица 2 – Сравнение языков программирования для разработки программного продукта

|  |  |  |
| --- | --- | --- |
| Название языка программирования | Php | Python |
| Наличие библиотек | + | + |
| Инструменты для работы с БД | + | + |
| Объектно-ориентированные возможности | + | + |
| Лёгкий понятный синтаксис | + | - |
| Более активное сообщество | + | - |
| Более лёгкая простая модульность | + | - |

Таким образом, php будет более лучшим вариантом, ведь он более компактный и простой в освоении язык. Он вобрал все лучшие особенности таких популярных языков, как C, Java и Perl, а также его сообщество более активное.

Для разработки программного продукта рассмотрены следующие инструментальные средства разработки программных продуктов:

1. PHPStorm.

2. Netbeans.

3. Atom.

PhpStorm — коммерческая кросс-платформенная [интегрированная среда разработки](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D0%B3%D1%80%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D1%80%D0%B5%D0%B4%D0%B0_%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8) обеспечивает первоклассную поддержку PHP 5.3–8.1, мгновенно предотвращает ошибки, предоставляет точное автодополнение и безопасные рефакторинги, а также возможность редактирования кода на HTML, CSS и JavaScript.

IDE обеспечивает умное автодополнение, подсветку синтаксиса, расширенные настройки форматирования, проверку ошибок на лету, сворачивание фрагментов кода, поддержку языковых комбинаций и много других возможностей для продуктивной разработки. Автоматические рефакторинги позаботятся о порядке в вашем коде и помогут легко и безопасно выполнять глобальные настройки проекта.

Сотни встроенных инспекций проверяют код прямо в режиме редактирования, а также позволяют проанализировать весь проект на наличие ошибок и проблем в структуре кода. Для большинства инспекций доступны быстрые исправления, которые позволяют откорректировать код мгновенно. Нажмите Alt+Enter, чтобы просмотреть список возможных исправлений для любой инспекции. PhpStorm помогает быстро перемещаться по коду и экономить время при работе над большими проектами. В один клик можно перейти к определению любого метода, функции или переменной, или найти их использования в проекте.

NetBeans IDE — свободная интегрированная среда разработки приложений (IDE) на языках программирования Java, Python, PHP, JavaScript, C, C++, Ада и ряда других. Проект NetBeans IDE поддерживается и спонсируется компанией Oracle, однако разработка NetBeans ведётся независимым сообществом разработчиков-энтузиастов (NetBeans Community) и компанией NetBeans Org. Последние версии NetBeans IDE поддерживают рефакторинг, профилирование, выделение синтаксических конструкций цветом, автодополнение набираемых конструкций на лету и множество предопределённых шаблонов кода. Для разработки программ в среде NetBeans и для успешной инсталляции и работы самой среды NetBeans должен быть предварительно установлен Sun JDK или J2EE SDK подходящей версии. Среда разработки NetBeans по-умолчанию поддерживала разработку для платформ J2SE и J2EE. Начиная с версии 6.0 NetBeans поддерживает разработку для мобильных платформ J2ME, C++ (только g++) и PHP без установки дополнительных компонентов.

Atom (в прошлом Atomicity) — бесплатный текстовый редактор с открытым исходным кодом для Linux, macOS, Windows с поддержкой плагинов, написанных на JavaScript, и встраиваемых под управлением Git. Большинство плагинов имеют статус свободного программного обеспечения, разрабатываются и поддерживаются сообществом.

Atom основан на Electron (ранее известный как Atom Shell) — фреймворке кроссплатформенной разработки с использованием Chromium и io.js. Редактор написан на CoffeeScript и LESS. Версия 1.0 была выпущена 25 июня 2015 г.

Сравнение IDE для разработки программного продукта наглядно представлено в таблице 3.

Таблица 3 – Сравнение IDE для разработки программного продукта

|  |  |  |  |
| --- | --- | --- | --- |
| Название IDE | PHPStorm | NetBeans | Atom |
| Распространяется бесплатно | + - | + | + |
| Автоматическое сохранение | + | - | + |
| Подсказки по коду | + | + | + |
| Интеграция с системой контроля версия (GIT) | + | + | - |
| Возможность расширения функционала библиотеками | + | + | + |
| Заточен под PHP-разработку | + | + | + |
| Поддержка CSS/HTML/JS | + | + | + |
| Комфортное использование на слабых ПК | + | - | + |

Таким образом, после рассмотрения вариантов средств разработок, было принято решение использовать PHPStorm. PHPStorm намного функциональнее, чем приведённые выше Atom и NetBeans. В PHPStorm имеется автоматическое сохранение, которое не позволит случайно потерять все наработки, а также он более удобен в работе.

Обоснование выбора

Для создания программного продукта было решено использовать

средства:

1) Для создания структурных схем, контекстной и диаграмм декомпозиции использовались CASE-средства – Draw.io.

2) Для наглядного составления структуры базы данных использовался инструмент для визуального проектирования баз данных, их редактирования и полного администрирования – MySQL Workbench.

3) Для разработки дизайна web-приложения использовался онлайн-сервис для разработки дизайна и прототипа сайта – Figma.

4) На этапе разработки программного продукта использовались инструменты и среды для разработки: PHP, Laravel, HTML5, CSS3, JavaScript и AJAX, а также редактор кода PHPStorm.

Основные преимущества веб-технологий:

1. Распределенность (пользователь может работать с системой из любого  
   места, связанного с WEB-сервером по сети, находясь в любой точке земного шара);
2. Переносимость (Web-клиенты (браузеры) существуют для любых  
   платформ, от настольных компьютеров до сотовых телефонов. Web-сервера используются для большинства платформ, а Web-приложения обычно пишутся на переносимых языках);  
   привычность интерфейса (почти каждый пользователь компьютера хотя бы раз запускал браузер и работал в нем);  
   простота установки и обслуживания (новую версию web-приложения не надо устанавливать на все компьютеры – достаточно установить на сервер).
3. Простота интерфейса - Пользователи не любят гигантских окошек с  
   сотнями полей ввода, а программисты и разработчики интерфейсов  
   почему-то любят. Web как раз не поощряет сложный интерфейс,  
   скорее поощряет простой.
4. Простота программирования интерфейса - Создавать HTML из  
   шаблонов куда проще и удобнее, чем создавать графические  
   приложения визуальными редакторами интерфейсов.
5. Простота программирования в целом - 3х-уровневая архитектура:  
   база данных – логика приложения (сервер) – логика представления  
   (клиент, браузер)

**3.Техническое задание**

Техническое задание, или ТЗ — это документ, в котором фиксируются требования к проекту. Условно ТЗ можно назвать любое поручение исполнителю, главное, чтобы в нем были ясно прописаны характеристики итогового продукта.

ТЗ представлено в Приложении А

1. **Проектирование ИС**

**4.1 Структурная схема ИС**

Проектирование информационной системы началось с построения диаграммы вариантов использования. На рисунке 1 представлена диаграмма прецедентов Uses CASE. Она содержит 3 актёра, которые могут выполнять суммарно 6 функций.

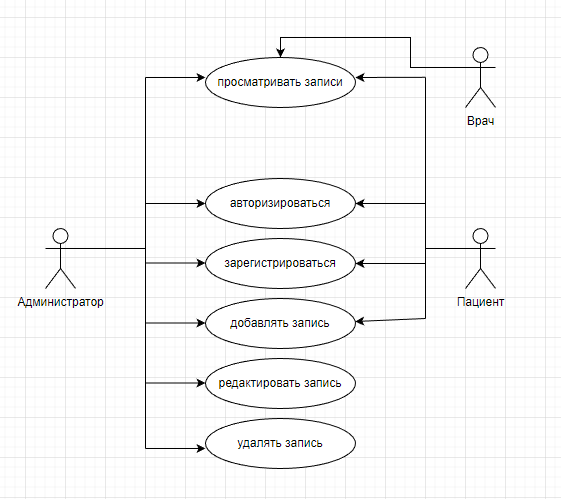


Рисунок 1 - Диаграмма прецедентов

На рисунке 2 представлена диаграмма деятельности. Она содержит 3 роли. Процесс начинается с пациента, который добавляет запись. Далее регистратура проверяет правильность записи, если запись возможна то данные уходят к врачу, если нет то пациент заново записывается.

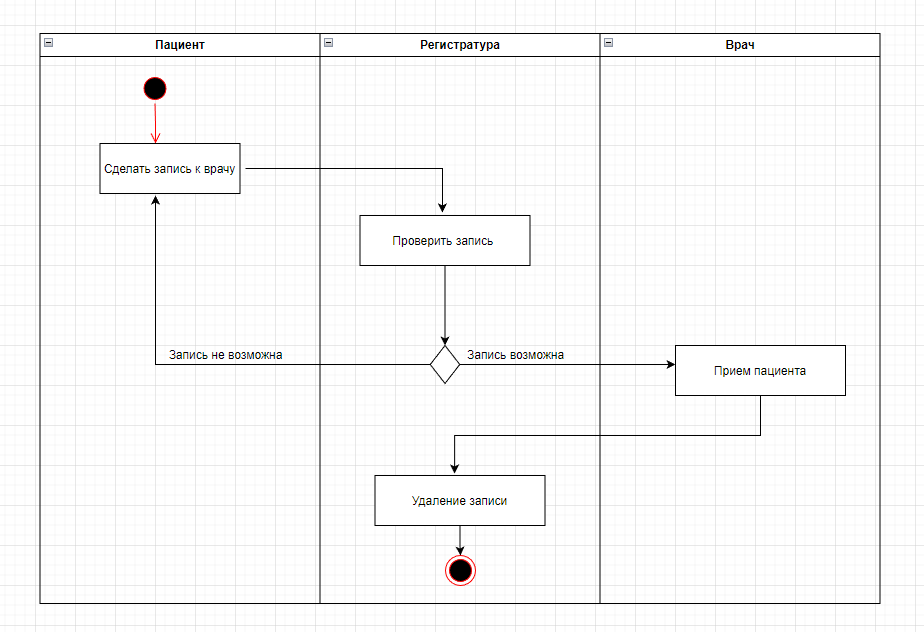


Рисунок 2 – Диаграмма деятельности

На рисунке 3 представлена диаграмма компонентов. В общей сложности 7 элементов.

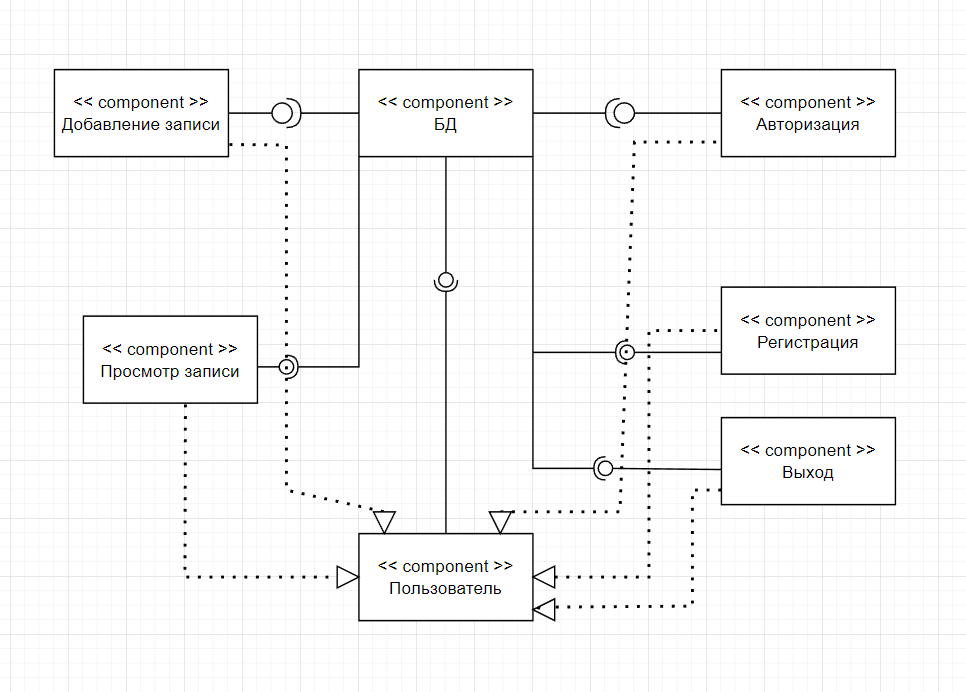


Рисунок 3 – Диаграмма компонентов

На рисунке 4 представлена диаграмма развёртывания. Она показывает, что, чтобы пользоваться программным продуктом, необходим web-сервер, на котором размещаются ИС и БД.

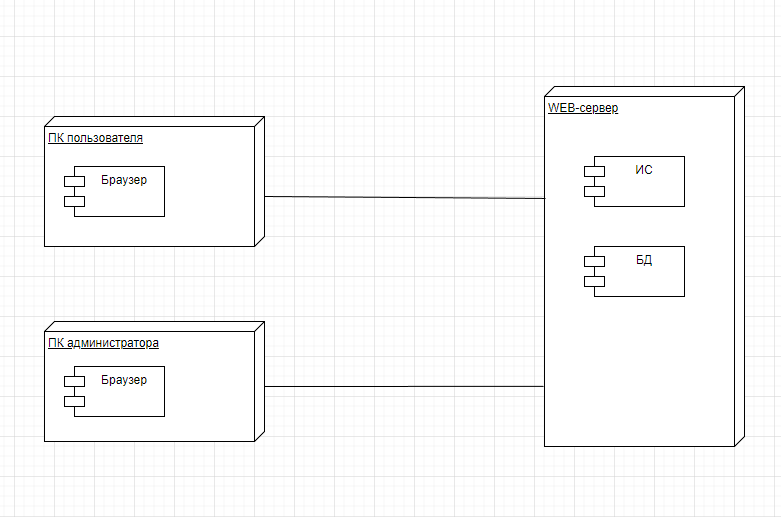


Рисунок 4 – Диаграмма развёртывания

* 1. **Функциональная схема ИС**

На рисунке 5 представлена контекстная диаграмма, отображающая деятельность ИС.

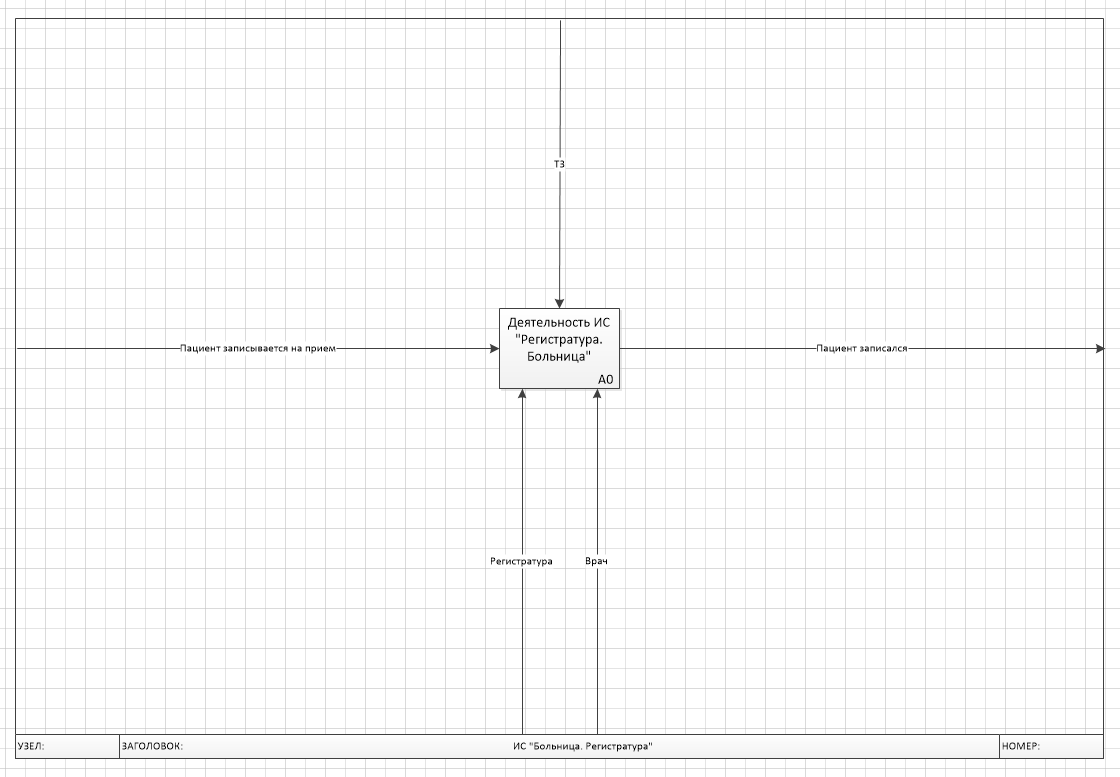


Рисунок 5 – Контекстная диаграмма IDEF0

На рисунке 6 представлена диаграмма декомпозиций (А1), отображающая деятельность ИС более подробно предыдущей.

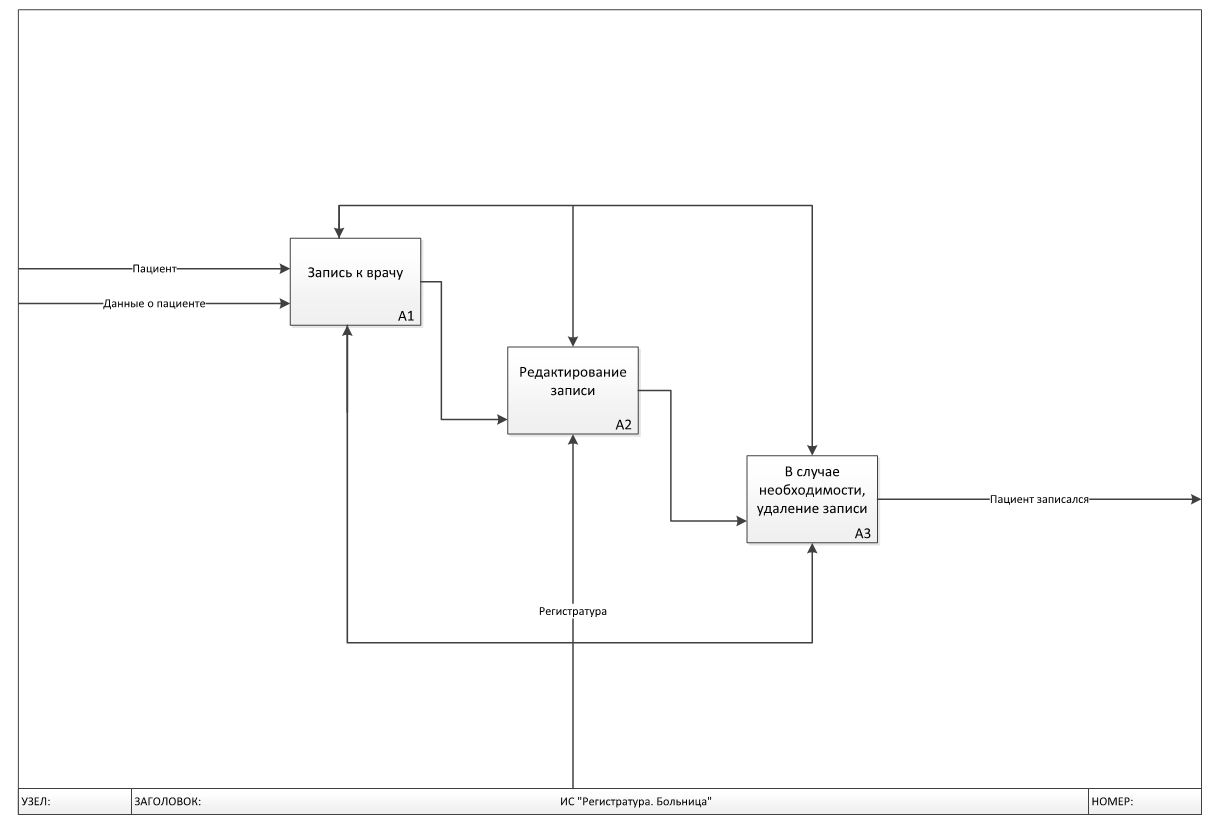
****

Рисунок 6 – Диаграмма декомпозиций А1

На рисунке 7 представлена диаграмма классов. Она содержит классы, среди них база данных, html-страница, информационная сеть «Регистратура. Больница», пользователь, врач, администратор, услуги.

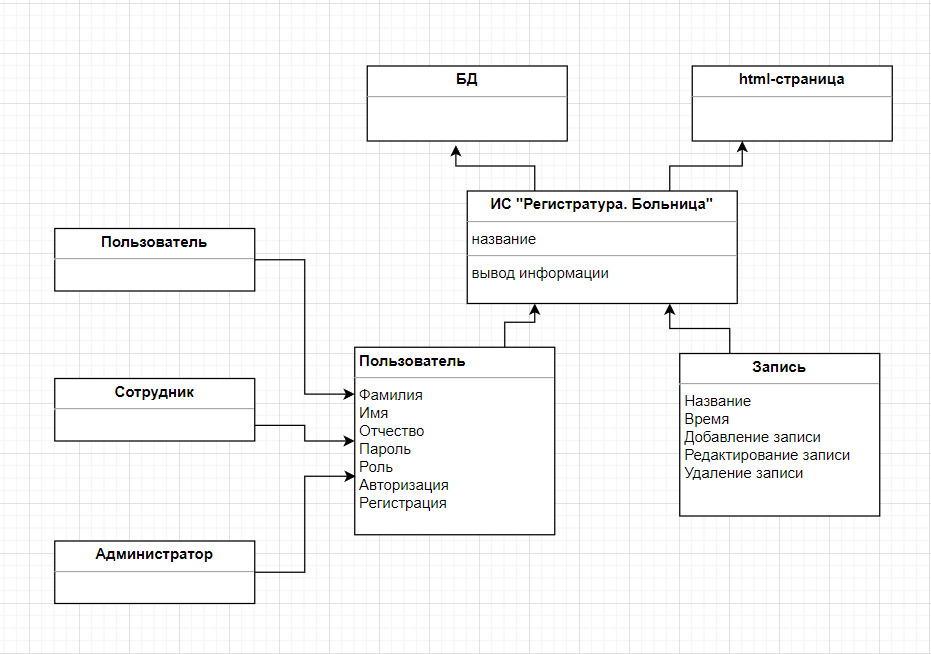


Рисунок 7 – Диаграмма классов

На рисунке 8 представлена диаграмма потоков данных. В центре всего пользователь, который может регистрироваться, просматривать услуги, добавлять, редактировать и удалять заказы. Для соответствующих действий в ИС предусмотрены соответствующие БД.

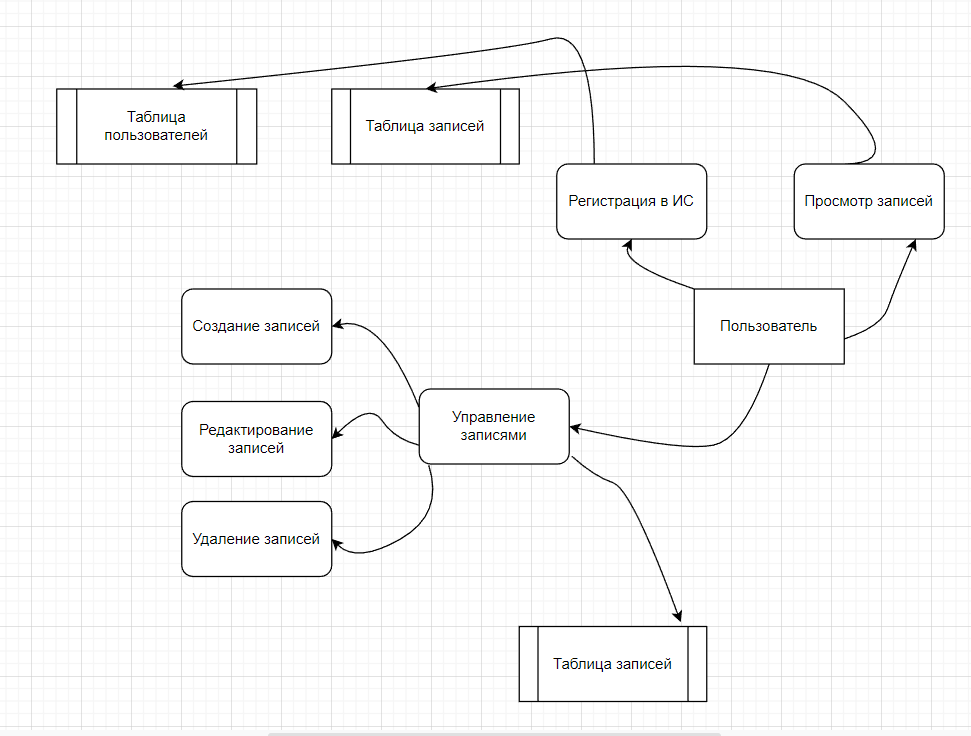


Рисунок 8 – Диаграмма потоков данных

4.3 Проектирование базы данных

Проектирование базы данных начинается с концептуального проектирование базы данных.

Концептуальное проектирование – построение семантической модели предметной области, то есть информационной модели наиболее высокого уровня абстракции. Такая модель создаётся без ориентации на какую-либо конкретную СУБД и модель данных.

На рисунке 9 представлена инфологическая модель базы данных.

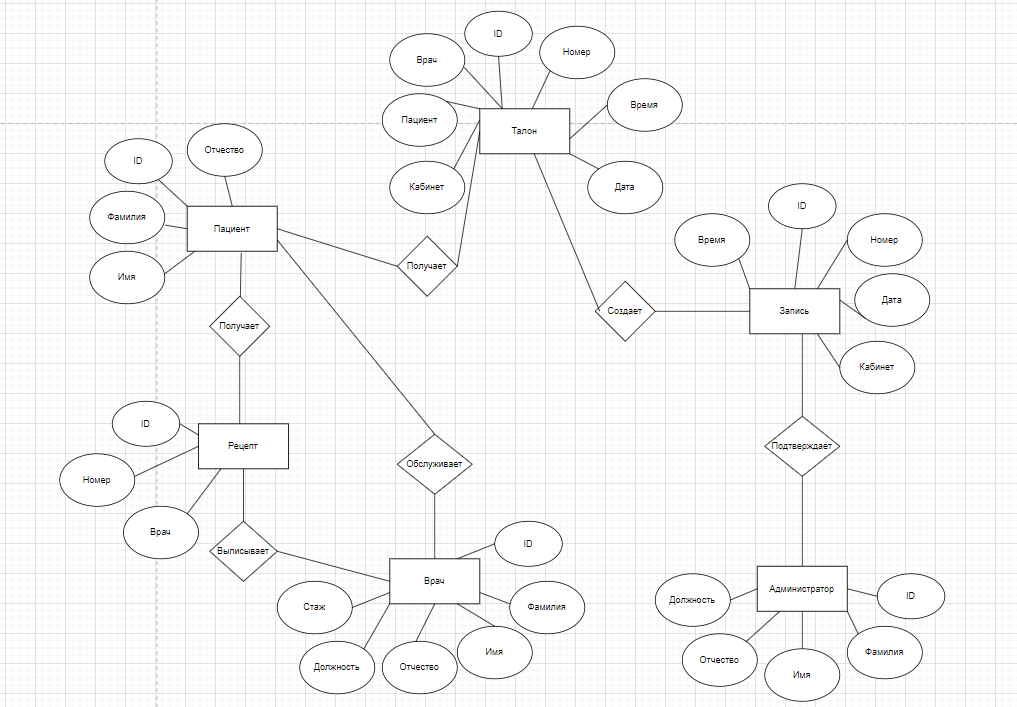
****

Рисунок 9 – Инфологическая модель базы данных

На инфологической модели базы данных выше схематично отображены сущности системы, их атрибуты и связи между ними. Так, в прямоугольник отображены сущности, такие как: Пациент, Администратор, Врач, Рецепт, Талон, Запись. В овалах отображены атрибуты сущностей, например: ФИО, ID и другие. Робами изображены связи между сущностями.

Далее происходит преобразование концептуальной модели в логическую модель, по формальным правилам. Таким образом, логическое (даталогическое) проектирование — создание схемы базы данных на основе конкретной модели данных, например, реляционной модели данных.

**Приложение А**

**1 Общие сведения**

Наименование работы: информационная система «Регистратура. Больница».

Исполнитель: студент иркутского авиационного техникума, группы ИС-20-1, Макрицкий А.В.

Разработка информационной системы проходит в рамках курсовой работы.

Сроки разработки информационной системы с 15.09.2022 по 06.12.2022 года.

**2 Цели и назначение создания автоматизированной системы**

Целью курсовой работы является создание информационной системы «Регистратура. Больница», которое включает в себя возможность записи на прием.

В данной информационной реализуется следующий функционал:

− авторизация;

− регистрация;

− запись на прием;

− просмотр записей;

− удаление записей.

**3 Характеристика объекта автоматизации**

Данная информационная система разрабатывается для облегчения записи в больницу. При помощи сайта будет проще записаться к врачу.

**4 Требования к системе в целом**

**4.1 Требования к структуре и функционированию сайта**

Функции информационная система:

1. Окно «Авторизации»:
   1. авторизация пользователей.
2. Окно «Регистрация»:
   1. регистрация пользователей.
3. Окно «Главная страница»:
   1. Информация о записях;
4. Окно «Услуги»:
   1. Таблица «Услуги»:
      1. добавление записи;
      2. удаление записи;
      3. редактирование данных.
   2. Окно «Пациенты»:
      1. добавление пациентов;
      2. удаление пациентов;
      3. редактирование данных пациентов.
   3. Окно «Врачи»:
      1. добавление сотрудников;
      2. удаление сотрудников;
      3. редактирование данных.

**4.2 Требования к надежности**

Для обеспечения надежности необходимо проверять корректность получаемых данных. Входные данные поступают в виде значений c клавиатуры. Эти значения отображаются в отдельных полях таблицы.

**4.3 Требования к безопасности**

Для обеспечения безопасности в информационной системе, необходимо реализовать разграничение прав доступа.

**4.4 Требования к эксплуатации, техническому обслуживанию, ремонту и**

**хранению компонентов системы**

Минимальные системные требования для сервера:

1. Процессор: Intel Pentium 4 2.0Ghz / AMD XP 2200+;
2. Оперативная память: 512 Мб;
3. Жёсткий диск: 150мб;
4. Операционная система: Windows 7/8/10.
5. Версия MySQL 5.0 и выше;

Минимальные системные требования для рабочей станции:

1. Процессор: Intel Pentium 4 2.0Ghz / AMD XP 2200+;
2. Оперативная память: 512 Мб;
3. Жёсткий диск: 150мб;
4. Операционная система: Windows 7/8/10.

**5 Состав и содержание работ по созданию системы**

В таблице 1 представлены плановые сроки начала и окончания работы по созданию веб-приложения.

Таблица 1 – Этапы выполнения разработки.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Этапы | Сроки | | | | | | | | |
| Провести предпроектное исследование. | 21.09 |  |  |  |  |  |  |  |  |
| Составить техническое задание на разработку в соответствии с ГОСТ. |  | 26.09 |  |  |  |  |  |  |  |
| Провести проектирование программного продукта. |  |  | 04.05 |  |  |  |  |  |  |
| Реализовать программный продукт. |  |  |  | 18.05 |  |  |  |  |  |
| Выполнить тестирование и отладку программного обеспечения. |  |  |  |  | 20.05 |  |  |  |  |
| Представить внедрение и сопровождение программного продукта. |  |  |  |  |  | 25.05 |  |  |  |
| Разработать документы для программного продукта. |  |  |  |  |  |  | 28.05 |  |  |
| Рассчитать стоимость разработки и внедрения программного продукта. |  |  |  |  |  |  |  | 30.05 |  |
| Составить пояснительную записку. |  |  |  |  |  |  |  |  | 01.06 |