Министерство образования и науки Российской Федерации

ФГАОУ ВПО «Северо-Восточный федеральный универститет им. М.К. Аммосова»

Институт математики и информатики

Кафедра информационных технологий

**НАУЧНО-ИССЛЕДОВАТЕЛЬСКОАЯ РАБОТА**

«Разработка веб-приложения для редактирования расписания учебных занятий»

Выполнил: магистрант II курса группы М-ИВТ-16, ИМИ СВФУ

Татаринов Артем Олегович

Научный руководитель:

Леверьев В. С. старший преподаватель кафедры ИТ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка, подпись руководителя)

Руководитель НИР:

Полуянова В. А. старший преподаватель кафедры ИТ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка, подпись руководителя)

Якутск

2018

Оглавление

[Введение 3](#_Toc483949531)

[Глава 1. Технологии публикации электронных расписаний 4](#_Toc483949532)

[1.1. Формат файлов iCalendar 4](#_Toc483949533)

[1.2. Сетевой доступ к календарям 5](#_Toc483949534)

[1.3. Программное обеспечение для работы с сетевыми календарями 8](#_Toc483949535)

[1.4. Онлайн-сервисы для создания расписаний 10](#_Toc483949536)

[Выводы к главе 1 12](#_Toc483949537)

[Глава II. Технологии разработки веб-приложений 13](#_Toc483949538)

[2.1. Преимущества и недостатки языков программирования 13](#_Toc483949539)

[2.2. Сравнение фреймворков Django и Flask 24](#_Toc483949540)

[2.3. Обзор библиотек визуализации 25](#_Toc483949541)

[Выводы к главе 2 27](#_Toc483949542)

[Заключение 28](#_Toc483949543)

[Список использованных источников 29](#_Toc483949544)

# Введение

Использование разного рода мобильных устройств и гаджетов, таких как смартфон, стало обыденным делом для современного человека. Этот факт провоцирует разработчиков создавать разные приложения, начиная развлекающими видеоиграми и заканчивая полезными приложениями, которые автоматизируют рутинные задачи. Одной из таких рутинных задач является составление расписания учебных занятий. Автоматизация данного процесса облегчит работу и сократит трудозатраты. Возможность экспортирования в формат iCalendar позволит использовать мобильные приложения типа iCal или Google Calendar. Благодаря этому, учащиеся и преподаватели будут иметь быстрый доступ к расписанию со смартфона, а также своевременно получать оповещения об изменениях в расписании.

Целью данной работы является разработка веб-приложения для редактирования расписания учебных занятий.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. провести обзор существующих технологий и приложений;
2. выбрать технологии и инструменты разработки;
3. написание и тестирование программного кода.

# Глава 1. Технологии публикации электронных расписаний

## Формат файлов iCalendar

**iCalendar** – формат компьютерных файлов, который позволяет пользователям публиковать данные о расписании событий и отправлять запросы на встречи и задачи. Обычно файлы в этом формате имеют расширение .ics. Используя соответствующее программное обеспечение, получатели файла данных iCalendar могут легко подтвердить или предложить наиболее удобную дату/время события. Формат файла первоначально был подробно описан в RFC 2445. Затем в сентябре 2009 года этот стандарт был заменен на RFC 5545 «Internet Calendaring and Scheduling Core Object Specification (iCalendar)».

iCalendar используется и поддерживается большим количеством продуктов, в том числе Google Calendar, Apple Calendar (ранее iCal), Facebook Events, Yahoo! Calendar, Evolution, Windows Calendar, расширение Lightning для Mozilla Thunderbird, SeaMonkey, Joomla, WordPress и Yandex Calendar.

**xCal** – XML-представление стандарта iCalendar. xCal не является альтернативой и не является следующим поколением iCalendar. xCal представляет компоненты, свойства и параметры, определенные в iCalendar. Этот формат был разработан для облегчения его перевода обратно в формат iCalendar с использованием преобразования XSLT.

**hCalendar** (сокращенно от HTML iCalendar) – стандарт микроформата для отображения семантического (X)HTML-представления информации о событии календаря формата iCalendar на веб-страницах, с использованием классов HTML и атрибутов rel.

Он позволяет использовать инструменты синтаксического анализа для извлечения подробностей события и отображения их с помощью какого-либо другого веб-сайта, индексирования или поиска, для их загрузки в календарь или в планировщик задач. [16]

## Сетевой доступ к календарям

**CalDAV** – сетевой протокол, который является расширением протокола HTTP. Протокол описан в стандарте RFC 4791. Это позволяет нескольким клиентам получить доступ к одной и той же информации, что позволяет совместное планирование и обмен информацией.

Архитектура CalDAV организует данные (события, задачи, информацию о занятости, заметки) в каталогах(коллекциях), где находятся несколько элементов(ресурсов). Ресурсы и коллекции могут быть доступны для одного или нескольких пользователей, используя стандартную семантику HTTP и DAV для обнаружения конфликтующих изменений или для обеспечения блокировки.

Для контроля доступа используется концепция ACL, поэтому каждая операция (просмотр, редактирование, удаление и т. д.) может быть запрещена или доступна пользователю. По этой причине в спецификации требуется, чтобы серверы CalDAV поддерживали «Протокол управления доступом WebDAV» (RFC 3744). [17]

**SyncML** – независимый от платформы язык синхронизации для управления данными. Он был создан для разработки единого протокола передачи данных для всех типов портативных устройств.

Программная структура основана на двух протоколах [14]:

* Протокол представления. Определяет формат представления SyncML-сообщений и описание внутренней конструкции работы SyncML.
* Протокол синхронизации. Определяет действия между SyncML-клиентом и SyncML-сервером.

SyncML поддерживает семь различных типов синхронизации:

* Two-way sync – стандартная двусторонняя синхронизация при которой клиент и сервер обмениваются модифицированными данными;
* Slow sync – форма двусторонней синхронизации, для которой все элементы сравниваются друг с другом. Клиент отправляет все свои данные из хранилища данных на сервер, а сервер выполняет анализ синхронизации полученных данных и данных на сервере;
* One-way sync from client only – односторонняя синхронизация, в котором только клиент отправляет свои изменения на сервер;
* Refresh sync from client only – тип синхронизации­, в котором только клиент отправляет все свои данные из хранилища данных на сервер;
* One-way sync from server only – односторонняя синхронизация, в котором только клиент получает все изменения с сервера;
* Refresh sync from server only – тип синхронизации, в котором только сервер отправляет все свои данные из хранилища данных клиенту. Это обычно называют операцией импорта с точки зрения клиента;
* Server-alerted sync – синхронизация с уведомлением при изменении серверных данных.

**Google Calendar API** – программный интерфейс приложения для интеграции с Google Calendar. С помощью этого интерфейса можно искать и просматривать события общедоступного календаря. Авторизация дает возможность получать и изменять частные календари и их события при определенном уровне доступа. Это позволит более эффективно планировать совместные события. Благодаря этому интерфейсу, мобильные приложения, веб-приложения и другие системы могут создавать, отображать или синхронизировать данные календаря.

На официальном сайте имеются пошаговые руководства по интеграции разрабатываемого приложения с Google Calendar на различных языках программирования. Так же, Google предоставляет клиентские библиотеки для удобства разработки. [19]

**Webcal** – неофициальная унифицированная схема идентификатора ресурсов (URI) для доступа к файлам iCalendar. Она также является названием двух приложений веб-календаря, написанных на Perl.

Схема Webcal была разработана для использования с приложением Apple iCal и стала стандартом де-факто для доступа к форматным файлам iCalendar через WebDAV, обычно с использованием метода GET. Это не официальная схема URI, как http и ftp, зарегистрированная в IANA. Префикс протокола Webcal используется для запуска внешнего обработчика протокола, который передает URL-адрес файла .ics, а не передает загруженное содержимое файла. Идея состоит в том, что с этим префиксом протокола целевой файл должен быть подписан, а не импортирован в приложение календаря, как это произошло бы с простой загрузкой. [20]

RFC 4324 – документ, в котором описан протокол доступа к календарю (**CAP**). CAP позволяет календарному пользователю (CU) использовать календарный пользовательский агент (CUA) для доступа к хранилищу календаря (CS) на базе iCAL и управлять календарной информацией. В частности, документ определяет, как запрашивать, создавать, изменять и удалять компоненты iCalendar (например, события, задачи или ежедневные записи журнала). Кроме того, он определяет, как искать информацию о занятом времени. Синхронизация с CUA не рассматривается, но считается, что это возможно с использованием CAP.

CAP также может использоваться для хранения и выборки объектов iCalendar Transport-Independent Interoperability Protocol (iTIP). Используемые iTIP-объекты точно определены в RFC 2446. Когда объекты iCalendar передаются между CUA и CS, могут добавляться некоторые дополнительные свойства и параметры. CUA отвечает за правильное создание объектов iCalendar для не-CAP-процессов. [21]

**HTTP** – протокол для обмена или передачи гипертекста. Он функционирует как протокол запроса-ответа в вычислительной модели клиент-сервер. Например, веб-браузер может быть клиентом, а приложение, запущенное на компьютере, на котором размещен веб-сайт, может быть сервером. Клиент отправляет HTTP-сообщение с запросом на сервер. Сервер, который предоставляет ресурсы, такие как файлы HTML и другой контент, или выполняет другие функции от имени клиента. Затем возвращает ответное сообщение клиенту. Ответ содержит информацию о состоянии завершения запроса и может также содержать запрошенный контент в своем теле сообщения.

Ресурсы HTTP идентифицируются и размещаются в сети с помощью унифицированных указателей ресурсов (URL), используя схемы URI (Uniform Resource Identifiers) http и https. URI и гиперссылки в документах HTML образуют взаимосвязанные гипертекстовые документы. [22]

## Программное обеспечение для работы с сетевыми календарями

**iCal** – популярный планировщик событий среди пользователей продуктов Apple. Он немного отличается от других продуктов Apple, так как разрабатывался независимой командой разработчиков. Для синхронизации с iCal используется формат данных iCalendar.

Не менее популярным планировщиком задач является **Google Calendar**. К ключевым преимуществам можно отнести возможность совместного использования календаря, что позволит планировать общие встречи. Для синхронизации с Google Calendar используется программный интерфейс приложения Calendar API.

**Microsoft Outlook** – менеджер личной информации от Microsoft, доступный в составе пакета Microsoft Office. Он используется в основном в качестве почтового приложения. Также включает календарь, диспетчер задач, диспетчер контактов, заметки, журнал и браузер веб-страниц.

Outlook не полностью поддерживает данные и синхронизацию спецификаций для календаря и контактов, таких как iCalendar, CalDAV, SyncML и vCard 3.0. Outlook 2007 утверждает, что полностью совместим с iCalendar. Он поддерживает vCard 2.1, но не поддерживает большое количество контактов в формате vCard в виде отдельного файла. Outlook также подвергся критике за наличие собственных расширений «Outlook» для этих интернет-стандартов.

Поскольку Microsoft Outlook не поддерживает протокол CalDAV и CardDAV, сторонние поставщики программного обеспечения разработали надстройки Outlook, чтобы позволить пользователям синхронизироваться с серверами CalDAV и CardDAV. CalConnect имеет список программного обеспечения, который позволяет пользователям синхронизировать свои календари с вышеуказанными серверами. [3]

**Mozilla Sunbird** – свободно распространяемое кроссплатформенное календарное приложение с открытым исходным кодом, разработанное Mozilla Foundation, Sun Microsystems и многими добровольцами. Sunbird была автономной версией календарного проекта Mozilla.

Он был разработан как автономная версия календаря Lightning и расширение расписания для почтовых клиентов Mozilla Thunderbird и SeaMonkey. В отличие от Lightning, Sunbird больше не получает обновления своей базы данных часовых поясов.

Mozilla Sunbird построен на стандарте iCalendar и использует сервер WebDAV для обеспечения функции общих календарей. Он поддерживает смену тем оформления и установку модулей расширения. Приложение способно синхронизироваться с Google Calendar с помощью плагина Provider for Google Calendar. [4]

**Evolution** – бесплатное программное обеспечение, лицензированное в соответствии с условиями общей широкой публичной лицензии GNU (LGPL). Он является официальным персональным информационным менеджером GNOME. Он объединяет электронную почту, адресную книгу, календарь, список задач и функцию заметок. Его пользовательский интерфейс и функциональность аналогичны Microsoft Outlook.

Календарь Evolution поддерживает формат файла iCalendar, стандарты WebDAV и CalDAV, а также Google Calendar. Синхронизация с SyncEvolution производиться с помощью SyncML. [23]

## Онлайн-сервисы для создания расписаний

Существует ряд электронных сервисов для планирования задач. Наиболее близкими к теме данной работы является FlipTable.ru и Schoodle.ru.

**FlipTable.ru** – сервис электронного расписания для учебных заведений. Данный сервис предоставляет услуги в нескольких ВУЗах страны: Государственный морской университет адмирала Ф. Ф. Ушакова, Сургутский педагогический университет, Технологический институт Северо-Восточного Федерального университета (филиал г. Мирный) и др. Данный сервис имеет ряд преимуществ, такие как контроль наложений расписаний, импорт справочников преподавателей/аудиторий/дисциплин и экспорт в форматах PDF, Excel и iCalendar. Возможность экспорта в формате iCalendar недоступна в бесплатном тарифе. Платный тариф составляет 100 руб./мес. за 1 группу. [24]

Сам процесс редактирования недостаточно удобен, даже при импорте всех справочников. Для того, чтобы добавить занятие необходимо для каждого дня недели заполнить форму с несколькими полями: дисциплина, тип занятия, подгруппа, номер группы, аудитория, лектор. Данный подход недостаточно эффективен, если расписание составляет один человек. Например, при редактировании поля «Дисциплина» появляется выпадающий список, со всеми загруженными дисциплинами, отфильтрованными по набранным буквам. При большом количестве групп этот процесс займет значительное количество времени. Необходимо упростить данный процесс с помощью функции Drag&Drop, который позволит с легкостью управлять событиями.

Экспортирование в персональные мобильные календари, такие как iCal и Google Calendar, является отличным функционалом для отображения расписания. Преимуществом таких календарей является возможность синхронизации с ними.

**Schoodle.ru** – бесплатный онлайн-сервис для составления занятий для школ, вузов и техникумов. Ввод первоначальных данных осуществляется посредством диалога с пользователем.

Данный сервис предоставляет возможность выбора количества учебных дней в неделе, ввода расписания перерывов(звонков), выбора количества групп(классов) для редактирования расписания, ручную запись занятий во временную сетку или копирования из других источников. [25]

К недостаткам данного сервиса можно отнести отсутствие проверки занятости аудиторного фонда/преподавателя/группы. Еще одним недостатком является ручное редактирование расписания.

А. Ю. Антропов и С. А. Варламова в своей статье «Проблемы автоматизированного составления расписания занятий средней общеобразовательной школы» [1] указывают, какой функционал должен иметь web-приложение для составления расписания.

Исходя из этой статьи можно предположить, что web-приложение должно решать следующие задачи:

* Добавление, редактирование и удаление начальных данных, например, количество учебных дней в неделе;
* Проверка данных, вводимых пользователем, на ошибки (противоречивость, ограничения);
* Публикация готового расписания на web-сайте;
* Экспорт расписания в форматы Microsoft Excel, iCalendar.

**Timetableweb.com** – платный онлайн-сервис для составления расписаний. В отличии от приведенных выше сервисов он имеет функцию автоматической генерации расписания по указанным ограничениям. Так же он имеет графический интерфейс для ручного редактирования. Экспорт расписания выполняется в форматах файла xls и pdf [31].

Данный сервис не является интуитивно понятным и лишен локализации на других языках. Для того, чтобы воспользоваться всеми функциями этого сервиса необходимо пройти полную активацию, который стоит 40 евро.

## Выводы к главе 1

В данной главе были рассмотрены формат iCalendar, его XML-представление xCal, а также hCalendar, который используется для отображения событий календаря формата iCalendar на веб-страницах. Кроме того, были рассмотрены протокола передачи данных, такие как CalDAV, SyncML, HTTP и протокол доступа к календарю RFC 4324.

Для разработки веб-приложения по всем параметрам подходят формат iCalendar и протокол передачи данных HTTP.

# Глава II. Технологии разработки веб-приложений

Для достижения поставленной цели необходимо выбрать наиболее подходящие для этой задачи технологии и средства разработки. В данной главе будут рассмотрены преимущества и недостатки языков программирования, также сравнение фреймворков и библиотек.

## 2.1. Преимущества и недостатки языков программирования

**2.1.1. Python**

Python – широко используемый язык программирования высокого уровня для программирования общего назначения, созданный Гвидо ван Россумом и впервые выпущенный в 1991 году. В интерпретируемом языке Python имеет философию дизайна, которая подчеркивает удобочитаемость кода (в частности, использование отступов для разделения кодовых блоков вместо фигурных скобок или ключевых слов) и синтаксис, который позволяет программистам писать меньшее число строк кода, чем это возможно в таких языках, как C++ или Java. [2]

Python обладает системой автоматического управления памятью и поддерживает несколько парадигм программирования, включая объектно-ориентированное, императивное, функциональное программирование и процедурные стили. Он имеет большую и всеобъемлющую стандартную библиотеку. [15]

Дизайн Python предлагает некоторую поддержку для функционального программирования в традиции Lisp. Язык имеет функции map(), reduce() и filter(); cписки, словари и наборы; генератор выражений. Стандартная библиотека имеет два модуля (itertools и functools), которые реализуют функциональные инструменты, заимствованные из Haskell и Standard ML. [2]

Преимущества[26]:

* открытая разработка;
* прост в изучении;
* особенности синтаксиса стимулируют разработчика писать хорошо читаемый код;
* предоставляет средства быстрого прототипирования и динамической семантики;
* имеет большое сообщество;
* множество полезных библиотек и расширений языка можно легко использовать в своих проектах благодаря предельно унифицированному механизму импорта и программным интерфейсам;
* механизмы модульности хорошо продуманы и могут быть легко использованы.

В области веб-разработки Python не уступает другим языкам программирования благодаря различным веб-фреймворкам, таким как Django или Flask.

Django CMS – свободная система управления контентом с открытым исходным кодом. Она написана на языке Python с использованием фреймворка Django. По умолчанию Django CMS обрабатывает многоязычный контент. Его административный интерфейс так же поддерживает несколько языков.

Первоначально Django CMS была задумана веб-разработчиками, которые были разочарованы техническими ограничениями и ограничениями безопасности других систем. Его ядро позволяет легко интегрировать его с другим программным обеспечением, а простота использования делает его доступным выбором для менеджеров контента, редакторов контента и администраторов веб-сайтов. [32]

**2.1.2. JavaScript**

JavaScript, часто сокращаемый как «JS», является высокоуровневым, динамическим, нетипизированным и интерпретируемым языком. Он был стандартизован в спецификации языка ECMAScript. [5] Наряду с HTML и CSS, JavaScript является одной из трех основных технологий создания контента в World Wide Web. Все современные веб-браузеры поддерживают его без использования плагинов. JavaScript является многопарадигматическим языком, поддерживающим объектно-ориентированный, императивный и функциональный стиль программирования. Он имеет API для работы с текстом, массивами, датами и регулярными выражениями, но не включает в себя какие-либо операции ввода-вывода, такие как сети, хранилища или графические объекты, полагаясь на них в среде хоста, в которую он внедрен [6].

JavaScript также используется в средах, которые не основаны на Web, таких как документы PDF, браузеры для конкретных сайтов и виджеты на рабочем столе. Новые и более быстрые виртуальные машины JavaScript и платформы, созданные на их основе, также повысили популярность JavaScript для серверных веб-приложений. На клиентской стороне разработчики традиционно используют JavaScript как интерпретируемый язык. Программисты также используют JavaScript в разработке видеоигр, при разработке настольных и мобильных приложений, а также при программировании на стороне сервера с использованием среды, такой как Node.js.

Преимущества[27]:

* все браузеры поддерживают JavaScript;
* имеет большое количество готовых плагинов и скриптов;
* постоянно совершенствуется;
* взаимодействие с приложением может осуществляется даже через текстовые редакторы;
* перспектива использования языка в процессе обучения программированию и информатике.

Недостатки:

* пониженный уровень безопасности ввиду повсеместного и свободного доступа к исходным кодам популярных скриптов;
* множество мелких раздражающих ошибок на каждом этапе работы. Большая часть из них легко исправляется, но их наличие позволяет считать этот язык менее профессиональным, сравнительно с другими.

Одним из популярных фреймворков для языка JavaScript является Express.js. Он является фреймворком веб-приложений для Node.js, реализованный как свободное и открытое ПО под лицензией MIT. Кроме проектирования веб-приложений, он направлен на создание API. Express может являться бэк-эндом для программного стека MEAN, вместе с базой данных MongoDB и фреймворком AngularJS для фронт-энда [7].

Meteor.js – фреймворк JavaScript с открытым исходным кодом, написанная с использованием Node.js. Meteor позволяет быстро создавать прототипы и является кросс-платформенным. Он использует Distributed Data Protocol и шаблон publish-subscribe для автоматического распространения изменений данных. Это лишает необходимости писать какой-либо код синхронизации. Meteor зависит от jQuery и может использоваться с любой библиотекой виджетов пользовательского интерфейса JavaScript. [33]

**2.1.3. Java**

Java – универсальный язык программирования, который является параллельным, основанным на классах, объектно-ориентированным языком. Скомпилированный код Java может запускаться на всех платформах, поддерживающих Java без необходимости перекомпиляции. Приложения Java обычно компилируются в байт-код, который может выполняться на любой виртуальной машине Java (JVM) независимо от архитектуры компьютера. Java была первоначально разработана Джеймсом Гослингом из Sun Microsystems (которая впоследствии была приобретена корпорацией Oracle) и выпущена в 1995 году в качестве основного компонента Java-платформы Sun Microsystems. Язык получил большую часть своего синтаксиса от C и C++, но у него меньше возможностей на нижнем уровне, чем у любого из них. [8]

Преимущества Java:

* кроссплатформенность;
* высокая безопасность;
* огромное количество бесплатных программных библиотек.

Недостатки:

* медлительность;
* ошибки в стандартных библиотеках.

Spring Framework – универсальный фреймворк с открытым исходным кодом для Java-платформы. Данный фреймворк был написан Родом Джонсоном, который впервые опубликовал ее вместе с изданием своей книги «Expert One-on-One Java EE Design and Development». Spring получил большую известность и признание среди разработчиков благодаря особенностям ядра и наличием множества расширений для разработки на платформе Java Enterprise.

Spring имеет свою MVC-платформу веб-приложений. Наиболее важными интерфейсами, перечисленных в Spring MVC, являются: HandlerMapping, HandlerAdapter, Controller, View, ViewResolver, HandlerInterceptor, LocaleResolver и MultipartResolver. Веб-приложениях можно использовать различные части Spring, а не только Spring MVC. [9]

**2.1.4. Ruby**

Ruby – динамический, рефлективный, интерпретируемый, высокоуровневый язык программирования. Создатель этого языка Юкихиро Мацумото объединил части других языков, таких как Perl, Smaltalk, Eiffel, Ada и Lisp. При создании этого языка он был нацелен на балансирование между парадигмой функционального программирования и принципами императивного программирования. [10]

Язык Ruby используется в веб-разработке в составе открытого веб-фреймворка Rails, чаще называемого Ruby on Rails (RoR). Ниже перечислены преимущества и недостатки языка с источника [26]

Преимущества:

* открытая разработка;
* работает на многих платформах;
* может внедряться в HTML-разметку;
* относится к языкам программирования сверхвысокого уровня (VHLL), то есть обладает высоким уровнем абстракции и предметным подходом в реализации алгоритмов;
* реализует концептуально чистую объектно-ориентированную парадигму;
* предоставляет продвинутые методы манипуляции строками и текстом;
* легко интегрирует в свои программы высокопроизводительные серверы баз данных (DB2, MySQL, Oracle и Sybase);
* благодаря VHLL программы на Ruby хорошо масштабируются и легко сопровождаются;
* простой и чистый синтаксис значительно облегчает программистам первые шаги в обучении этому языку;
* имеется простой программный интерфейс для создания многопоточных приложений;
* имеет продвинутые средства для работы с массивами;
* возможности языка можно расширить при помощи библиотек, написанных на C или Ruby;
* зарезервированные слова могут являться идентификаторами, если это не создаёт неоднозначности для парсера;
* дополнительные возможности для обеспечения безопасности;
* встроенный отладчик.

Недостатки:

* обучение языку выше начального уровня может оказаться непростым;
* информационных ресурсов, посвящённых Ruby, явно недостаточно;
* Ruby менее производителен по сравнению со многими другими языками, применяемыми в веб-разработке;
* Ruby относительно медленно разрабатывается и развивается.

Альтернативой Ruby on Rails является фреймворк Sinatra. Это свободная библиотека программного обеспечения с открытым исходным кодом, написанный на Ruby.

Sinatra не следует за типовым шаблоном model-view-controller, используемым в других фреймворках, таких как Ruby on Rails. Вместо этого Sinatra фокусируется на «быстрой разработке веб-приложений в Ruby с минимальными усилиями». [34]

**2.1.5. Perl**

Perl — высокоуровневый, интерпретируемый, динамический язык программирования общего назначения, созданный Ларри Уоллом. Особенностью данного языка являются широкие возможности для работы с текстом. Он приобрел свою популярность благодаря регулярным выражениям, встроенных в синтаксис. Perl унаследовал некоторые функции других языков, таких как C, AWK и скриптовых языков командных оболочек UNIX. Согласно источнику [27], язык имеет следующие преимущества и недостатки.

Преимущества:

* встроенные средства для работы со сложными структурами;
* свободный синтаксис (одна и та же задача может решаться разными способами);
* много готовых библиотек – модулей;
* поддержка работы с регулярными выражениями;
* простая обработка больших объемов данных;
* возможность программирования объектно-ориентированным или «функциональным» стилем;
* кросс-платформенность.

Недостатки:

Для решения некоторых задач часть преимуществ языка Perl превращаются в недостатки. Большое количество модулей затрудняет их поиск, что не нравится программистам, которым из всего многообразия нужны лишь пара конкретных.

**2.1.6. PHP**

PHP – язык программирования, исполняемый на стороне веб-сервера, спроектированный Расмусом Лердорфом в качестве инструмента создания динамических и интерактивных веб-сайтов.

Этот язык оказался достаточно гибким и мощным, поэтому пользуется большим спросом среди веб-разработчиков. Благодаря этому он используется в проектах различной сложности: от простого блога до сложных веб-приложений.

На сайте [26] приводят следующие преимущества и недостатки PHP.

Преимущества:

* является свободным программным обеспечением, распространяемым под особой лицензией (PHP license);
* легок в освоении на всех этапах;
* поддерживается большим сообществом пользователей и разработчиков;
* имеет развитую поддержку баз данных;
* имеется огромное количество библиотек и расширений языка;
* может использоваться в изолированной среде;
* предлагает нативные средства организации веб-сессий, программный интерфейс расширений;
* является довольно полной заменой проприетарной среды ASP (Active Server Pages) от Microsoft;
* может быть развёрнут почти на любом сервере;
* портирован под большое количество аппаратных платформ и операционных систем.

Недостатки:

* не подходит для создания десктопных приложений или системных компонентов;
* имеет слабые средства для работы с исключениями;
* глобальные параметры конфигурации влияют на базовый синтаксис языка, что затрудняет настройку сервера и разворачивание приложений;
* объекты передаются по значению, что смущает многих программистов, привыкших к передаче объектов по ссылке, как это делается в большинстве других языков;
* веб-приложения, написанные на PHP, зачастую имеют проблемы с безопасностью.

Для веб-разработки на PHP существует огромное количество фреймворков: Yii, Laravel, Symphony, Zend, Drupal, Codeigniter, Cakephp и другие. Согласно источнику [35] самым популярным фреймворком является Laravel. За ним идут Symfony и Codeigniter.

Laravel – бесплатный веб-фреймворк с открытым исходным кодом PHP, предназначенный для разработки веб-приложений в соответствии с архитектурным шаблоном model-view-controller (MVC).

У Laravel множество функций, обеспечивающих быструю разработку приложений. Он имеет собственный движок для шаблонов Blade. Его простой синтаксис упрощает реализацию частых функций: сессии, аутентификация, кеширование, анализ очередей и маршрутизацию RESTful. Также у Laravel есть локальная среда разработки Homestead [35].

Symfony – фреймворк веб-приложений PHP. Она была опубликована как бесплатное программное обеспечение 18 октября 2005 года и выпущена под лицензией MIT.

Symfony имеет ряд PHP-библиотек, которые можно использовать повторно. Это позволяет выполнять множество рутинных задач: создание форм, конфигурацию объектов, маршрутизацию, аутентификацию, создание шаблонов и другие. Любой компонент можно установить с помощью Composer-менеджера пакетов для PHP [35].

Codeigniter – веб-фреймворк с открытым исходным кодом для быстрой разработки веб-сайтов, предназначенный для создания динамических веб-сайтов с PHP.

Codeigniter, так же, как и Laravel, основан на шаблоне разработки model-view-controller (MVC). Хотя классы контроллеров являются необходимой частью разработки в Codeigniter, модели и представления являются необязательными. Codeigniter также можно модифицировать, чтобы использовать Hierarchical Model View Controller (HMVC), который позволяет разработчикам поддерживать модульную группировку контроллеров, моделей и представлений, расположенных в формате подкаталогов [35].

**2.1.7. C#**

C# – объектно-ориентированный язык программирования. Он был написан группой разработчиков под началом Андерса Хейлсберга в компании Microsoft. Это язык разработан специально для платформы Microsoft .NET Framework и стандартизован как ECMA-334 и ISO/IEC 23270. Основной синтаксис языка C # аналогичен синтаксису других языков C-стиля, таких как C, C ++ и Java. [11]

Перцев С.А. и Островский В.В. в своей работе «Язык программирования C#. Достоинства и недостатки, основные возможности. Сопоставление с другими языками программирования» приводят следующие сильные и слабые стороны языка. [12]

Преимущества:

* язык программирования C# претендует на подлинную объектную ориентированность (всякая языковая сущность претендует на то, чтобы быть объектом);
* компонентно-ориентированный подход к программированию, способствующий меньшей машинно-архитектурной зависимости результирующего программного кода, гибкости, переносимости и легкости повторного использования (фрагментов) программ;
* ориентация на безопасность кода (в сравнении с С и С++);
* унифицированная система типизации;
* расширенная поддержка событийно-ориентированного программирования.

Недостатки:

* довольно сложный синтаксис (75% из Java, 10% из C++, 5% из Visual Basic);
* мало свежих концептуальных идей (приблизительно менее чем 10% конструкций языка);
* относительно невысокая производительность (намного медленнее, чем язык C, но сравним с Java);
* не кроссплатформенный язык. Так как С# разработан компанией Microsoft, то и работает он только под операционной системой Windows, хотя в данный момент уже разрабатывается кроссплатформенная версия данного языка.

Для разработки веб-приложений на языке C# используется ASP.NET MVC Framework. Это платформа веб-приложений, разработанная Microsoft. Она реализует шаблон model-view-controller (MVC).

«ASP.NET MVC Framework – легкая, высоко тестируемая платформа для презентации, интегрированная с существующими функциями ASP.NET. Одними из этих интегрированных функций являются мастер-страницы и аутентификация на основе членства. Структура MVC определена в System.Web.Mvc» [36].

## 2.2. Сравнение фреймворков Django и Flask

Самым популярным фреймворком является Django и список сайтов, которые его используют, впечатляет: Bitbucket, Pinterest, Instagram и The Onion. Для сайтов, которые имеют общие требования, Django выбирает очень разумные значения по умолчанию, и из-за этого он стал популярным выбором для веб-приложений среднего и крупного размера.

Как сказано выше Django нацелена на более крупные приложения, но использует разные подходы к расширяемости и гибкости. Он позволяет включить любые пакеты («батареи»), которые потребуются веб-приложению, поэтому разработчикам нужно только начать работать, подключая различные модули Django. Подход Django «с подключенными батареями» позволяет разработчикам быстро погружаться в разработку веб-приложения, не задумываясь об инфраструктуре своего приложения раньше времени. Этот фреймворк предоставляет шаблоны, формы, маршрутизацию, аутентификацию, базовое администрирование баз данных и многое другое.

Django включает в себя ORM, а Flask предоставляют разработчику возможность выбрать способы хранения данных. Наиболее популярным ORM для веб-приложений, отличных от Django, является SQLAlchemy, но есть множество других вариантов от DynamoDB и MongoDB до простой локальной устойчивости, такой как LevelDB или обычный SQLite.

Django является достаточно «зрелым» фреймворком и накопил множество плагинов и расширений для удовлетворения невероятно широкого круга потребностей.

Flask – «микрофреймворк», предназначенный, прежде всего, для небольших приложений с более простыми требованиями.

Наличие выбора компонентов приложений Flask дает большую гибкость для разработчиков, которым необходимо взаимодействовать с различными рабочими процессами или системами шаблонов.

Flask отлично подходит для разработчиков, работающих с небольшими проектами, которым требуется быстрый способ создания простого веб-сайта на Python. Он питается множеством небольших одноразовых инструментов или простых веб-интерфейсов, созданных поверх существующих API. [29]

## 2.3. Обзор библиотек визуализации

**MxGraph** – интерактивная JavaScript-библиотека диаграмм HTML 5. MxGraph использует родной язык рисования векторной графики, доступный для обеспечения богатой функциональности диаграмм в тонкой архитектуре клиента. Он также включает back-end функциональность для .NET, PHP и Java, которые обеспечивают доступ к графической модели.

Для использования библиотеки всего лишь необходимо включить его как ссылку JavaScript в свой HTML-файл. Это позволит быстро получить доступ к функциональному, встроенному компоненту диаграмм. Эта библиотека используется в Draw.io. Библиотека разработана с 2005 года и поддерживает даже IE 6.

Программное обеспечение доступно только на условиях лицензии mxGraph, стандартной коммерческой лицензии. [13]

**Jgraphx** – библиотека с открытым исходным кодом для визуализации графов. Он целиком написан на языке программирования Java и полностью совместима со Swing’ом. Библиотека разработана с поддержкой различных представлений, сущностей и их отношений. Jgraph позволяет работать с неориентированными и ориентированными графами, с мультиграфами, с гиперграфами. Так же он дает возможность использовать различные алгоритмы позиционирования вершин. [30]

**JointJS** – современная библиотека JavaScript HTML5 для визуализации и взаимодействия с диаграммами и графиками. Его можно использовать для создания статических диаграмм или полностью интерактивных инструментов построения диаграмм, таких как редакторы рабочих процессов, инструменты управления процессами, IVR-системы, API-интеграторы, презентационные приложения и многое другое [37].

JointJS включает:

* различные формы (Rect, Circle, Ellipse, Text, Image, Path, ERD, Org Chart, FSA, UML, BPMN, PN, DEVS, ...);
* пользовательские формы с помощью SVG;
* интерактивные элементы и ссылки;
* подключение элементов диаграммы со ссылками;
* настраиваемые ссылки, их стрелки и ярлыки;
* иерархические диаграммы;
* высокое управление событиями - вы можете реагировать на любое событие, которое происходит внутри документа;
* тонны математических функций, работающих в 2D пространстве;
* анимации;
* фильтры и градиенты;
* поддержка NodeJS;
* быстрота - возможность визуализации сотен (или даже тысяч) элементов и ссылок с мгновенным взаимодействием;
* гибкая система плагинов;
* архитектура MVC;
* и многое другое.

**Cytoscape.js** – библиотека с открытым исходным кодом, написанная на JavaScript. Вы можете использовать Cytoscape.js для анализа и визуализации графов.

Cytoscape.js предоставляет возможность легко отображать и манипулировать богатыми интерактивными графиками. Поскольку Cytoscape.js позволяет пользователю взаимодействовать с графиком, а библиотека клиенту – подключаться к пользовательским событиям, Cytoscape.js легко интегрироваться в разрабатываемые приложения [37].

## Выводы к главе 2

В данной главе были рассмотрены преимущества и недостатки различных языков программирования для веб-разработки. Все приведенные языки программирования хороши для выполнения тех или иных задач. Для достижения поставленной цели необходим простой в изучении язык, обладающий интуитивно понятным синтаксисом и хорошо продуманным механизмом модульности. Таким языком является Python.

Среди представленных фреймворков выбор пал на Django, который является простым и гибким фреймворком, обладающий богатой библиотекой готовых решений. Для визуализации объектов будет использована библиотека MxGraph.

# Глава 3. Разработка приложения

## Структура приложения

Для разработки приложения был использован фреймворк Django. Страницы веб-приложения состоят из шаблонов приведенных в таблице 1.

Таблица 1. Шаблоны

|  |  |
| --- | --- |
| **Шаблон** | **Описание** |
| base | В этом шаблоне подключаются статические файлы |
| event\_form |  |
| group\_form | Шаблон для редактирования информации о группе |
| group\_list | Справочник учебных групп |
| login | Шаблон авторизации |
| menu | Меню |
| room\_form | Шаблон для редактирования информации об аудитории |
| room\_list | Аудиторный фонд |
| semester\_form | Шаблон для редактирования информации о семестре |
| semester\_list | Список семестров |
| subject\_class |  |
| subject\_form | Шаблон для редактирования информации о дисциплине |
| subject\_group | Список учебных групп |
| subject\_list | Список дисциплин выбранной группы |
| teacher\_form | Шаблон для редактирования информации о преподавателе |
| teacher\_list | Справочник преподавателей |
| week | Страница редактора расписания учебных |

Для загрузки данных в справочники «Учебная группа», «Преподаватели», «Аудитории» и «График учебного процесса» используется код приведенный в файле «scr.py»:

|  |
| --- |
| from schedule\_editor import models import xml.etree.ElementTree as ET  tree = ET.parse('D:\\Education\\Диссертация\\РУП\\020302\_18-1ФИИТ.plm.xml') root = tree.getroot()  '''Группа''' group = tree.find('.//Титул') form = tree.find('.//План') qualification = tree.find('.//Квалификация') new\_group = models.StudentGroup() new\_group.name = group.get('ИмяПлана') new\_group.year = group.get('ГодНачалаПодготовки') if (qualification.get('Название') == 'бакалавр'):  new\_group.qualification = 'BAC' elif (qualification.get('Название') == 'магистрант'):  new\_group.qualification = 'MAG' elif (qualification.get('Название') == 'специалист'):  new\_group.qualification = 'SPC'  if (form.get('ФормаОбучения') == 'очная'):  new\_group.form = 'OC' elif (form.get('ФормаОбучения') == 'заочная'):  new\_group.form = 'ZA' new\_group.save()  '''Семестр''' graphic = tree.find('.//ГрафикУчПроцесса') curse\_count = 1 all\_disciplines = tree.findall('.//СтрокиПлана//Строка')  for curse in graphic.findall('.//Курс'):  sem\_count = 1  if 'КаникулНед' not in curse.attrib:  print("curse")  continue  for sem in curse.findall('.//Семестр'):  # new\_semester = models.Semester()  if (int(sem.get('Ном')) == sem\_count):  new\_semester = models.Semester()  new\_semester.student\_group = new\_group  new\_semester.year = int(group.get('ГодНачалаПодготовки')) + (curse\_count - 1) + (sem\_count - 1)  new\_semester.semester = models.Semester.SEMESTER[sem\_count - 1][0]  new\_semester.save()  '''Дисциплины'''  for discipline in all\_disciplines:  for sem\_d in discipline.findall('.//Сем'):  if (int(sem.get('Ном'))-1 == (int(sem\_d.get('Ном'))+1)%2 and int(sem\_d.get('Ном')) == curse\_count \* 2 - sem\_count % 2):  new\_subject = models.Subject()  new\_subject.name = discipline.get('Дис')  new\_subject.semester = new\_semester  new\_subject.save()  sem\_count += 1  curse\_count += 1 |

Инструкции этого файла выгружают данные из рабочего учебного плана, разработанного в формате xml. Считывание необходимых данных производится с помощью инструмента структурированной обработки разметки «xml.etree» стандартной библиотеки Python. После этого, полученная информация сохраняется в базе данных.

## Структура базы данных

При разработке баз данных, процесс описания атрибутов полей требует особого внимания. Во-первых, нужно подобрать имя поля, которое, с одной стороны, должен быть информативным, а с другой - кратким, что обеспечит несомненные удобства при дальнейшей работе с ним. Во-вторых, необходимо определить тип данных полей, исходя из содержания тех сведений, которые будут в нем храниться.

База данных редактора расписания состоит из 7 моделей: дисциплина(Subject), занятие дисциплины(SubjectClass), учебная группа(StudentGroup), преподаватель(Teacher), аудитория(Room), семестр(Semester), событие(Event).

Таблица 1 содержит информацию о дисциплинах: название дисциплины, количество часов работ. Информация, хранящаяся в этой таблице, поможет приложению ~~сформировать периодичность занятий~~ определить период теоретического обучения.

Таблица 1. Дисциплина(Subject)

|  |  |  |
| --- | --- | --- |
| **Поле** | **Тип** | **Описание** |
| id(PK) | integer | ID дисциплины |
| name | varchar(100) | Наименование дисциплины |
| lecture\_hours | integer | Кол-во часов лекций |
| lab\_work\_hours | varchar(2) | Кол-во часов лабораторных |
| practice\_hours | varchar(3) | Кол-во часов практических |
| student\_work\_hours | integer | Кол-во часов СРС |
| control\_hours | integer | Кол-во часов КСР |
| total\_hours | integer | Общее кол-во часов |
| Semester(FK) | integer | ID семестра |

У каждой дисциплины может быть несколько занятий разного типа (лекция/практика/лабораторная работа). В таблице 2 приведена информация о занятиях дисциплины: тип занятия, периодичность, номер занятия, ID дисциплины, ID преподавателя, день недели.

Таблица 2. Занятие дисциплины(SubjectClass)

|  |  |  |
| --- | --- | --- |
| **Поле** | **Тип** | **Описание** |
| id(PK) | integer | ID занятия дисциплины |
| class\_type | varchar(3) | Тип занятия |
| period | integer | Период |
| number | integer | Номер занятия |
| subject(FK) | integer | ID дисциплины |
| Teacher(FK) | integer | ID преподавателя |
| weekday | integer | День недели |

Сведения об учебной группе хранятся в таблице 3. Для определения курса группы будет использоваться значение поля «year».

Таблица 3. Учебные группы(StudentGroup)

|  |  |  |
| --- | --- | --- |
| **Поле** | **Тип** | **Описание** |
| id(PK) | integer | ID группы |
| name | varchar(30) | Наименование группы |
| year | integer | Год поступления |
| form | varchar(2) | Форма обучения |
| qualification | varchar(3) | Квалификация |

Таблица 4 предоставляет информацию о преподавателе.

Таблица 4. Преподаватель(Teacher)

|  |  |  |
| --- | --- | --- |
| **Поле** | **Тип** | **Описание** |
| id(PK) | integer | ID преподавателя |
| first\_name | varchar(50) | Имя |
| last\_name | varchar(50) | Фамилия |
| second\_name | varchar(50) | Отчество |
| user(FK) | integer | ID пользователя |

В таблице 5 указывается номер аудитории и здание, в котором находится аудитория.

Таблица 5. Аудитория(Room)

|  |  |  |
| --- | --- | --- |
| **Поле** | **Тип** | **Описание** |
| id(PK) | integer | ID события |
| number | varchar(50) | Номер аудитории |
| building | varchar(50) | Здание |

В таблице 6 хранится информация о семестрах: календарный год, семестр, дата начала/конца теоретического обучения и экзаменационной сессии.

Таблица 6. Семестр(Semester)

|  |  |  |
| --- | --- | --- |
| **Поле** | **Тип** | **Описание** |
| id(PK) | integer | ID события |
| year | integer | Календарный год |
| semester | varchar(3) | Семестр |
| begin\_study | date | Начало теор. обучения |
| end\_study | date | Конец теор. обучения |
| begin\_exams | date | Начало экз. сессии |
| end\_exams | date | Конец экз. сессии |
| student\_group(FK) | integer | ID группы |

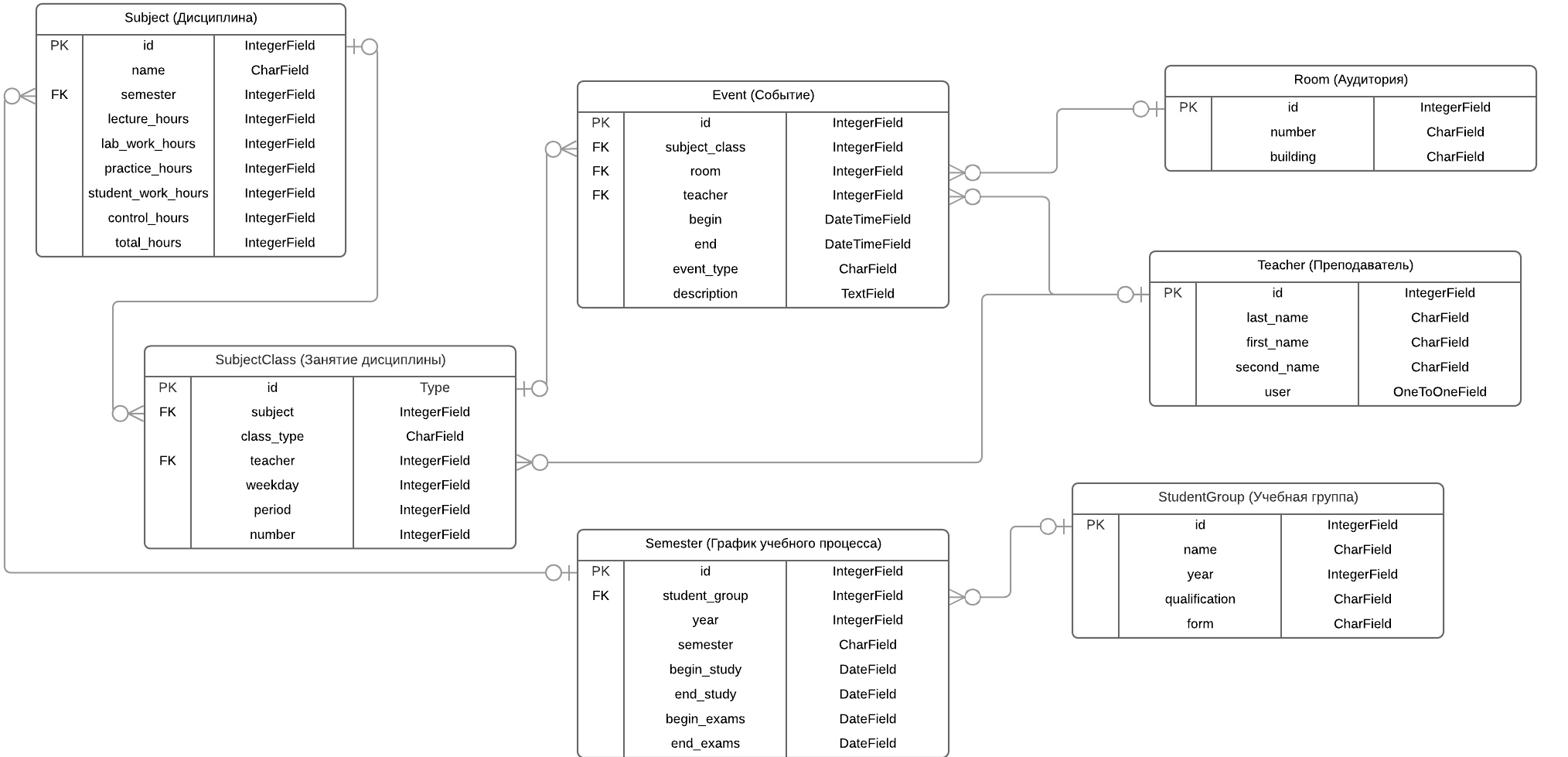
Таблица 7 содержит информацию о событии, которая будет отображаться в расписании: наименование дисциплины, тип занятия, дата и время начала/конца пары, аудитория, преподаватель.

Таблица 7. Событие(Event)

|  |  |  |
| --- | --- | --- |
| **Поле** | **Тип** | **Описание** |
| id(PK) | integer | ID события |
| begin | datetime | Начало |
| end | datetime | Конец |
| event\_type | varchar(30) | Тип занятия |
| description | text | Описание |
| room(FK) | integer | ID аудитории |
| subject\_class(FK) | integer | ID занятия дисциплины |
| teacher(FK) | integer | ID преподавателя |

В результате разработки базы данных и связей между таблицами схема базы данных имеет вид, представленный на Рис. 1.

Рис. 1. Схема базы данных



## Интерфейс приложения

Главная

Расписание занятий

## Вывод к главе 3

# Заключение

В ходе работы были изучены аналоги разрабатываемого приложения. Был проведен анализ технологий публикации электронных расписаний и технологий разработки. В результате анализа, для создания веб-приложения для редактирования расписания учебных занятий, были выбраны язык программирования Python, фреймворк Django, библиотека визуализации MxGraph и формат файлов iCalendar с использованием протокола обмена данными HTTP.

Таким образом, в ходе работы были выполнены следующие задачи:

1. провести обзор существующих технологий и приложений;
2. выбрать технологии и инструменты разработки.

# Список использованных источников

Книги, учебные пособия, статьи:

1. «Juventus Scientia 2016» №2 С. 10-13
2. Стив МакКоннелл «Code Complete» - Microsoft Press, 2009 ­– 100 с.
3. Глеб Архангельский «Формула времени. Тайм-менеджмент на Outlook 2013. 8-е издание.» — М.: Манн, Иванов и Фербер, 2014. — 224 с.
4. Philipp Kewisch «Provider for Google Calendar» ­— Mozilla, 2007
5. Ecma International «ECMAScript Language specification» — Geneva, 2016 — 586 с.
6. Дэвид Флэнаган, «JavaScript. Подробное руководство», 2008
7. Итан Браун «Веб-разработка с применением Node и Express» — Санкт-Петербург: Питер, 2017. — 336 с.
8. Herbert Schildt, «Java. Beginner’s Guide» — McGraw-Hill Osborne Media, 2011.
9. Кларенс Хо, Роб Харроп, «Spring 3 для профессионалов» — М.: «Вильямс», 2012. — 880 с.
10. Е. А. Роганов, Н. А. Роганова, «Программирование на языке Ruby. Учебное пособие» — М.: МГИУ, 2008. — 56 с.
11. Ecma International, «C# Language Specification 4th edition» — 2006.
12. «Экономика и социум» №3(22), 2016.
13. JGraph Lnd., «JGraph User Manual» — 2009 — 136 с.
14. Phillipe Kahn, «SyncML: Synchronizing and managing your mobile data» — Prentice Hall, 2002 — 286 c.

Электронные ресурсы:

1. "About Python". Python Software Foundation. [2012] URL: https:// [www.python.org/about/](http://www.python.org/about/) (Дата обращения 22.05.2017)
2. Calendaring and Scheduling Standards Simplification [2014] URL: https:// datatracker.ietf.org/wg/calsify/charter/ (Дата обращения 22.05.2017)
3. iCal Import/Export CalDAV [2010] URL: <http://www.drlue.tk/> ical/?page\_id=124 (Дата обращения 22.05.2017)
4. Material from Affiliates – SyncML [2014] URL: http://[www. openmobilealliance.org/wp/Affiliates/SyncML.html](http://www.openmobilealliance.org/wp/Affiliates/SyncML.html) (Дата обращения 22.05.2017)
5. Get Started with the Calendar API URL: <https://developers.google.com> /google-apps/calendar/overview (Дата обращения 15.05.2017)
6. Uniform Resource Identifier (URI) Schemes [2017] URL: <https://www>. iana.org/assignments/uri-schemes/uri-schemes.xhtml (Дата обращения 15.05.2017)
7. Calendar Access Protocol (CAP) [2005] URL: <https://tools.ietf.org/html/rfc4324> (Дата обращения 15.05.2017)
8. Hypertext Transfer Protocol -- HTTP/1.1 [1999] URL: <https://tools>. ietf.org/html/rfc2616#section-1.4 (Дата обращения 15.05.2017)
9. Evolution Mail and Calendar [2014] URL: <https://help.gnome.org/> users/evolution/stable/index.html (Дата обращения 15.05.2017)
10. Электронное расписание для учебных заведений URL: https:// fliptable.ru/ (Дата обращения 15.05.2017)
11. Бесплатный сервис для составления и публикации расписания уроков и расписания занятий для школы, техникума и вуза онлайн. URL: <https://schoodle.ru/> (Дата обращения 15.05.2017)
12. PHP, Ruby, Python краткая характеристика трёх языков программирования [2014] URL: http://www.internet-technologies.ru /articles/article\_1991.html (Дата обращения 22.05.2017)
13. Что такое JavaScript? URL: <https://www.ipipe.ru/info/javascript.html> (Дата обращения 22.05.2017)
14. Perl. История появления, области испльзования URL: <https://www>. ipipe.ru/info/perl.html (Дата обращения 22.05.2017)
15. Django vs Flask vs Pyramid [2015] URL: <https://www.airpair.com> /python/posts/django-flask-pyramid (Дата обращения 22.05.2017)
16. Визуализация графов на JAVA [2011] URL: <https://habrahabr.ru/> sandbox/24540/ (Дата обращения 22.05.2017)
17. Онлайн-сервис автоматической генерации расписания URL: <http://www.timetableweb.com> (Дата обращения 30.05.2017)
18. Официальный сайт Django CMS URL: <https://www.django-cms.org/en/> (Дата обращения 30.05.2017)
19. Meteor wants to be the warp drive for building real-time apps [2014] URL: <https://gigaom.com/2014/12/27/meteor-wants-to-be-the-warp-drive-for-building-real-time-apps/> (Дата обращения 30.05.2017)
20. Getting started URL: <http://www.sinatrarb.com/intro.html> (Дата обращения 30.05.2017)
21. 10 лучших PHP-фреймворков для веб-проектов [2016] URL: https:// kultprosvet.net/ru/blog/ 10-luchshih-php-freymvorkov-dlya-veb-proektov (Дата обращения 30.05.2017)
22. ASP.NET MVC [2011] URL: <http://aspnet4.com/asp-net-4-thoery/asp-net-mvc/> (Дата обращения 30.05.2017)
23. Top 5: Best free diagrams javascript libraries [2016] URL: http:// ourcodeworld.com/articles/read/159/top-5-best-free-diagrams-javascript-libraries (Дата обращения 30.05.2017)