# Оглавление:

Введение 3

1. Постановка задачи 4
2. Теоретические сведения 5
   1. Краткая характеристика 5
   2. История 6

2.3 Особенности и принципы работы ASP.NET MVC 7

1. Практическая часть 8
   1. Подготовка к работе и создание базы данных 8
   2. Начало работы с ASP.NET Framework и создание проекта 10

3.3 Разработка архитектуры проекта 12

3.3.1 Контроллер и его методы 14

* + 1. Создание представлений 19

3.4 Аутентификация и авторизация 25

3.5 Тестирование и результат 28 Заключение и выводы 34

Список литературы 35

Приложение 1 36

Приложение 2 40

Приложение 3 41

# Введение

На современном этапе развития технологий сложно представить себе жизнь без Интернета. Сайты, размещенные в сети Интернет, позволяют быстро доносить информацию большому количеству лиц, привлекать новых участников, организовать обратную связь с посетителями и автоматизировать длительные по времени и регулярно повторяющиеся процессы. Благодаря таким сайтам, различные организации формируют свой образ далеко за пределами своего фактического местонахождения.

В качестве темы выпускной квалификационной работы была выбрана тема «Создание информационного сайта кафедры». Данная тема актуальна тем что, хоть у кафедры МО ЭВМ и есть сайт, в нем отсутствуют функции, которые позволяли бы выполнять регулярно повторяемые операции (такие как распределение нагрузки) в автоматическом режиме. Помимо этого, данный сайт устарел в моральном и техническом плане. В результате анализа данных проблем и было принято решение выбрать эту тему.

# **1. Цели и задачи**

Целью данной выпускной квалификационной работы являлся процесс создания веб-сайта с использованием технологии, который будет соответствовать современным критериям сайтов учебных заведений. Сайт должен соответствовать следующим критериям:

* Иметь удобное представление информации
* Иметь удобное редактирование информации
* Быть быстро действенным
* Быть кроссплатформенным и кроссбраузерным
* Автоматизировать длительные по времени и регулярно проводимые операции

Достижение указанной цели осуществлялось путём решения следующих задач:

* Разработка структуры сайта и определение функциональных возможностей.
* Разработка структуры и создание базы данных.
* Разработка основных функций сайта: система авторизации, размещение новостей, вывод и редактирование информации о сотрудниках, автоматическое распределение учебной между преподавателями на основе учебного плана.
* Оформление и стилизация веб-сайта.

С учётом вышеперечисленных требований к функциональности сайта была составлена диаграмма вариантов использования сайта, изображенная на рисунке 1.

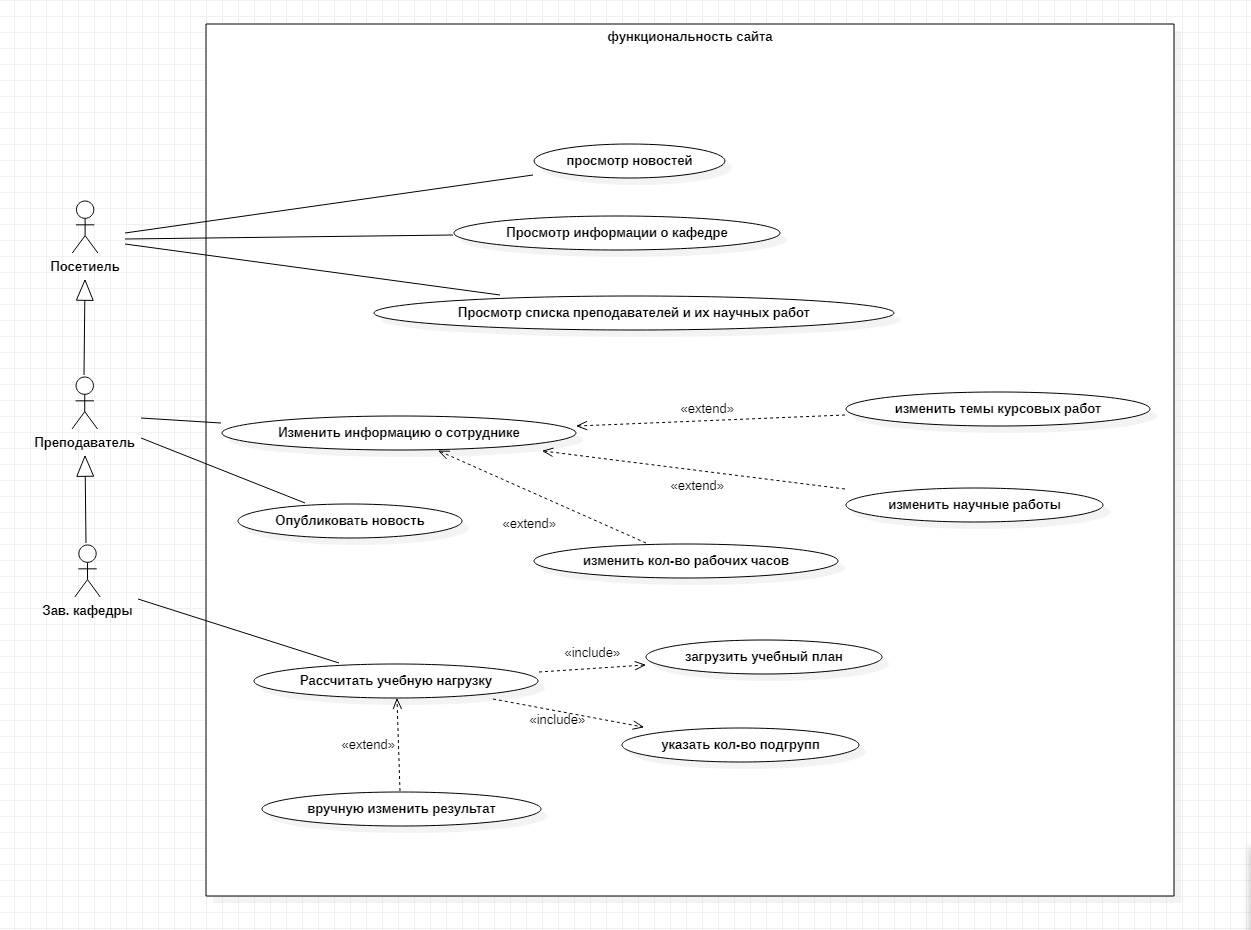


Рисунок 1 - диаграмма вариантов использования сайта

# 2. Описание средств разработки

Принимая во внимание все вышеперечисленные требования, было принято решение выбрать ASP.NET MVC 5 как основную технологию для

разработки данного сайта. Начиная с 5 версии ASP.NET

является кроссплатформенной технологией и может использоваться на

операционных системах Windows и Linux.

Каждая технология, предназначенная для веб-разработки, имеет как достоинства, так и недостатки. Сравнивая ASP.NET с другими популярными технологиями веб-разработки, такими как Ruby-on-Rails, Node.js, PHP или Python, выделяется одно ключевое достоинство. Перечисленные средства являются интерпретируемыми, в то время как C# – компилируемый язык. Это предоставляет платформе .NET большое преимущество в скорости работы. Большая часть ошибок отлавливается разработчиком в момент компиляции, а все компоненты, не требуют интерпретатора, вместо этого работая с фреймворком, который, в свою очередь уже скомпилирован и вызывает напрямую функции операционной

системы.

Как упоминалось ранее - разрабатываемый продукт

является не компилируемым. И хотя язык C# является компилируемым, сам проект ASP.NET MVC 5 не требует сборки. Вместо этого при необходимости компиляция всех файлов проходит в режиме реального времени, а изменения можно вносить в код даже во время работы приложения.

В итоге, проект, созданный на платформе ASP.NET MVC 5 будет обладать одновременно двумя преимуществами: первое - возможность разработки с использованием большого количества интерпретируемых платформ, второе - быстродействие и высокая производительность создаваемых сервисов.

В качестве основного паттерна приложения был выбран стандартный шаблон MVC (Model-View-Controller). Данный шаблон разделяет приложение на три основных части: модель приложения, пользовательский интерфейс и взаимодействие с пользователем.

Схематичное изображение взаимодействия компонентов MVC представлено на Рисунке 2.1.

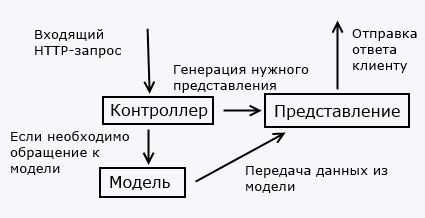


Рисунок 2 – Схема взаимодействия компонентов MVC

**Контроллер** является основным компонентом приложения. Данный класс связывает модель и представление. При получении введённых пользователем данных, он их обрабатывает согласно внутренней логике, а затем, при необходимости, обращается к модели и создает представление, соответствующее данному контроллеру.

**Представление** отвечает за визуальную часть приложения, оно выступает в качестве интерфейса, с которым взаимодействует пользователь, зашедший на сайт.

**Моделью** является набор классов, которые описывают логику используемых данных.

**Выбор системы управления базой данных, создание структуры**

Для хранения и систематизации данных на сайте используются базы данных. Для управления базой данных используется система управления базой данных – СУБД. На данный момент на рынке представлено большое количество различных СУБД, среди них – Oracle, MS SQL, MySQL, SQLite.

1. MySQL. На сегодняшний день является одной из самых распространённых СУБД. Как правило, используется при разработке веб сайтов и других веб приложений. Система является бесплатной и открытой, что является одним из значимых факторов ее популярности.

Преимуществам данной системы являются:

* СУБД легко установить. Как правило, она идет в комплекте с веб сервером, например как LAMP(Linux,Apache,MySQL,PHP). Программный продукт может управлять через консоль или графический интерфейс.
* СУБД практически полностью поддерживает язык SQL.
* MySQL может достаточно просто работать с большими объемами данных и при необходимости легко масштабироваться.
* Некоторые упрощения некоторых стандартов языка SQL позволяет ускорить работу СУБД.

К недостаткам использования MySQL можно отнести:

* Не полное соответствие стандартам языка SQL. Если до момента использования данной СУБД специалист работал с системой, использующей стандартный SQL, то при переходе на MySQL могут возникнуть некоторые трудности связанные с изучением специфики используемого языка.
* Проблемы с надежностью. MySQL из-за некоторых методов обработки информации в некоторых ситуациях уступает другим СУБД. Например, при кластерной обработке MySQL может организовать DDos атаку на собственную базу.

2.        Postgre SQL

1. Из числа всех бесплатных СУБД является самым профессиональным решением. Postgre SQL бесплатная и открытая система. Но в отличии от MySQL старается полностью соответствовать стандартам языка SQL.

К достоинствам данной системы можно отнести:

* Postgre бесплатная система с открытым исходным кодом, которая соответствует стандартам SQL.
* Достаточно большое количество дополнений которые позволяют расширять функционал системы.[17]
* В отличии от других СУБД обеспечивает более высокий уровень целостности данных и транзакций.

К недостаткам можно отнести:

* При выполнении достаточно простых операций СУБД использует значительно больше ресурсов, чем ее конкуренты.
* Не смотря на достаточно большую базу пользователей, достаточно трудно найти хостинг, который будет поддерживать Postgre.

3.        Microsoft SQL server

* Microsoft SQL server СУБД разработанная компанией Microsoft. Так же как MySQL и Postgre предназначена для управления реляционными базами данных. Microsoft SQL server использует язык Transact-SQL. Данный язык соответствует стандартам языка SQL.

К преимуществам СУБД можно отнести:

* Шифрование всей хранимой информации в БД. Шифрование и расшифровка информации происходит непосредственно перед чтением или записью информации.
* Интуитивно понятный интерфейс.
* Высокая производительность СУБД.
* Легкая масштабируемость системы.

К недостаткам можно отнести:

* Ограниченный выбор платформ. СУБД будет работать только под управлением операционных систем семейства windows.
* Ограниченное число средств для разработки клиентских приложений. По умолчанию количество платформ для разработки клиентской части ограниченно. Другие СУБД поддерживают большее количество платформ.
* Microsoft SQL server является не бесплатным ПО. SQL server имеет несколько редакций которые отличаются предоставляемым функционалом и стоимостью.

[http://evansys.com/articles/perspektivy-i-tekhnologii-razvitiya-v-oblasti-tekhnicheskikh-nauk-sbornik-nauchnykh-trudov-po-itogam/sektsiya-20-informatsionnye-tekhnologii/sravnenie-sistem-upravleniya-bazami-dannykh/]

На основании проведенного анализа была выбрана Microsoft SQL в качестве основной СУБД, поскольку Microsoft SQL server является самым мощным решением из всех рассмотренных. По умолчанию данная СУБД обладает огромным функционалом. Продукт компании Microsoft позволяет решать самые сложные задачи по хранению и обработке больших массивов данных.

Выбрав СУБД для использования, была разработана структура базы данных, изображенная на рисунке 3.

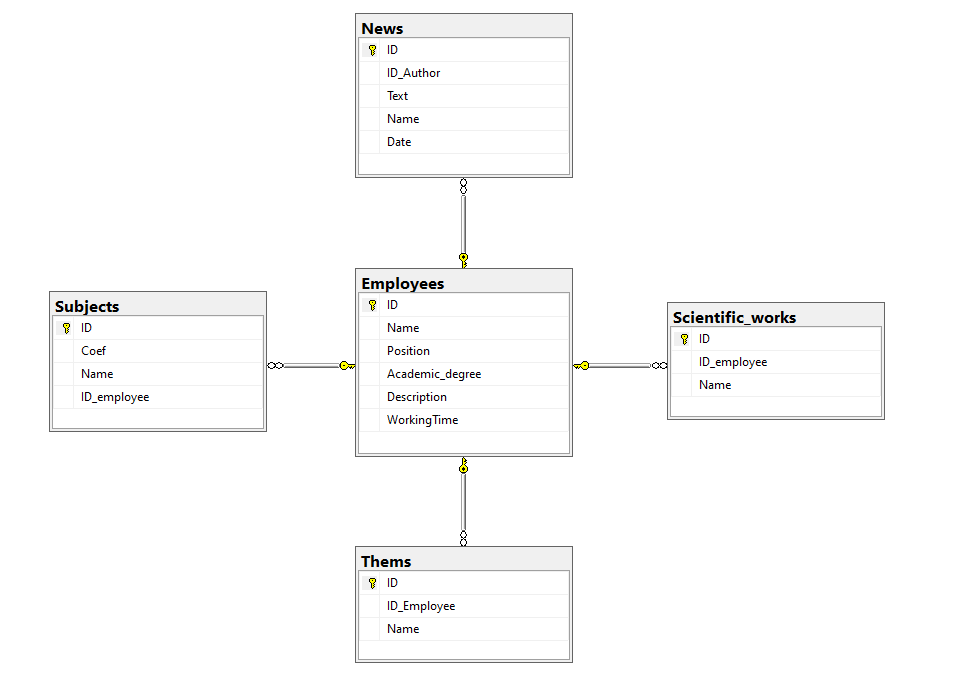


Рисунок 3 - структура базы данных