

ОТЗЫВ

НАУЧНОГО РУКОВОДИТЕЛЯ

на _____ дипломный проект _____
(наименование вида работы)
_____ Каратаев Арсен Асхатулы _____
(Ф.И.О. обучающегося)
_____ 5B070400 - Вычислительная техника и программное обеспечение _____
(шифр и наименование специальности)

Тема: «Автоматизация бизнес процесса формирования рабочего учебного плана».

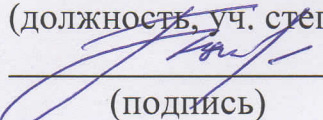
Перед студентом Каратаевым Арсеном стояла задача разработать разработать веб-сервис по автоматизации бизнес процесса формирования рабочего учебного плана. Данная тема одна из наиболее актуальных для высших учебных заведений т.к. в большинство современных ВУЗов внедрена кредитная система образования .

Был использован официальный стек технологий для разработки веб-сервиса: Язык программирования C# и TypeScript, интегрированная среда разработки Visual Studio, фреймворки Angular, ASP.NET core.

Каратаев Арсен успешно освоил MVP архитектуру приложений и разработал удобный интерфейс.

Поставленные перед студентом задачи выполнены и решались самостоятельно.

Подводя итоги, дипломный проект соответствует специальности 5B070400 – «Вычислительная техника и программное обеспечение», студент заслуживает присвоения академической степени «бакалавра» и может быть допущен до защиты.

Научный руководитель
Заведующий кафедрой ПИ
канд. техн. наук, доцент,
ассистент-профессор
(должность, уч. степень, звание)
 Юнусов Р.
(подпись)

« 16 » _____ 05 2019г.

Протокол анализа Отчета подобия Научным руководителем

Заявляю, что я ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Каратаев Арсен

Название: Дипломная работа Каратаев Арсен

Координатор: Расул Юнусов

Коэффициент подобия 1:4,2

Коэффициент подобия 2:0

Тревога:0

После анализа Отчета подобия констатирую следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, признаю работу самостоятельной и допускаю ее к защите;
- обнаруженные в работе заимствования не обладают признаками плагиата, но их чрезмерное количество вызывает сомнения в отношении ценности работы по существу и отсутствием самостоятельности ее автора. В связи с чем, работа должна быть вновь отредактирована с целью ограничения заимствований;
- обнаруженные в работе заимствования являются недобросовестными и обладают признаками плагиата, или в ней содержатся преднамеренные искажения текста, указывающие на попытки сокрытия недобросовестных заимствований. В связи с чем, не допускаю работу к защите.

Обоснование:

.....
.....
.....
.....
.....
.....

16.05.197

Дата

.....

Подпись Научного руководителя

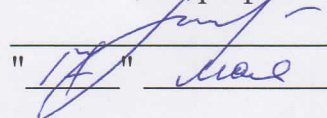
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

СӘТБАЕВ УНИВЕРСИТЕТІ

Институт информационных и телекоммуникационных технологий

Кафедра "Программная инженерия"

ДОПУЩЕН К ЗАЩИТЕ
Заведующий кафедрой ПИ
канд. техн. наук, доцент,
ассистент-профессор

 Р. Юнусов
" 17 " май 2019 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к дипломному проекту

На тему: "Автоматизация бизнес процесса формирования рабочего учебного плана"

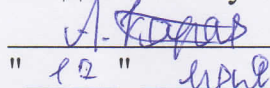
по специальности 5В070400 – Вычислительная техника и программное обеспечение

Выполнил

Каратаев А. А.

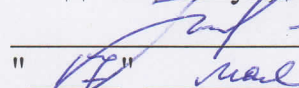
Рецензент

канд. техн. наук

 А.А. Каратаев
" 12 " май 2019 г.

Научный руководитель

канд. техн. наук, доцент

 Р. Юнусов
" 17 " май 2019 г.

Алматы 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

СӘТБАЕВ УНИВЕРСИТЕТІ

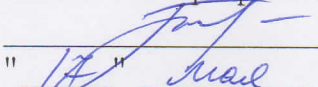
Институт информационных и телекоммуникационных технологий

Кафедра "Программная инженерия"

5B070400 – Вычислительная техника и программное обеспечение

УТВЕРЖДАЮ

Заведующий кафедрой ПИ
канд. техн. наук, доцент,
ассистент-профессор

 Р. Юнусов
" 17 " мая 2019 г.

ЗАДАНИЕ

на выполнение дипломного проекта

Обучающемуся Каратаеву Арсену Асхатұлы

Тема: Автоматизация бизнес процесса формирования рабочего учебного плана

Утверждена приказом проректора по академической работе № 1162 - б
от "16" октября 2018 г.

Срок сдачи законченного проекта

"10" мая 2019 г.

Исходные данные к дипломному проекту: Паспорт проекта, техническая документация по применению технологии, техническое задание, описание БД для хранения информации в виде ER-диаграммы.

Перечень подлежащих разработке в дипломном проекте вопросов:

а) разработка классовой модели реляционной базы данных SQLite согласно представленной диаграмме;

б) реализация многозвенной архитектуры веб-приложения в соответствии с концепцией MVC;

в) проектирование системы разделения ролей пользователей;

г) проектирование и разработка пользовательского интерфейса;

д) разработка, отладка, тестирование программного комплекса.

Перечень графического материала (с точным указанием обязательных чертежей): представлены в презентации.

Рекомендуемая основная литература: из 7 наименований.

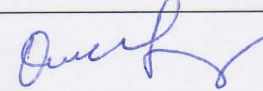

ГРАФИК

подготовки дипломного проекта

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю и консультантам	Примечание
1. Анализ предметной области, разработка технического задания на проектирование программного комплекса	3	Зам. нем.
2. Реализация классовой структуры модели базы	7	Зам. нем.
3. Разработка пользовательского интерфейса, реализация контроллеров для исполнения функций пользователя "Деканат", "Заведующий кафедрой", "Офис регистратора"	10	Зам. нем.
4. Разработка структуры приложения с реализацией логической цепочки иерархических вызовов	7	Зам. нем.
5. Разработка дизайна интерфейса	14	Зам. нем.
6. Написание пояснительной записки к дипломному проекту	7	Зам. нем.

Подписи

консультантов и нормоконтролера на законченный дипломный проект с указанием относящихся к ним разделов проекта

Наименования разделов	Консультанты, И.О.Ф. (уч. степень, звание)	Дата подписания	Подпись
Программное обеспечение	Өмірбекова З.М.	6.05.19.	
Нормоконтролер	Омарова Г.А.	6.05.19	

Научный руководитель _____

Р. Юнусов

Задание принял к исполнению обучающийся _____

А.А. Каратаев

Дата

"17" мая 2019 г.

АННОТАЦИЯ

Документы приходится печатать, переделывать, где-то хранить и согласовывать содержимое, при этом возникают сложности с форматом документа. Данный проект автоматизирует часть работы офис регистратора, заведующих кафедры и деканата, связанной с рабочим учебным планом для факультетов. Использование этой системы поможет улучшить функционирование высшего учебного заведения, сэкономит время и спасет, как минимум, несколько деревьев. С внедрением информационной системы процессы работы с рабочими учебными планами упрощаются и некоторые действия, требовавшие внимание и отнимающие определенное количество времени, будут автоматизированы, путем создания системы управления рабочей учебной программой с определением действующих лиц и разграничением ролей.

АҢДАТПА

Құжаттардың пішімімен қиындықтар туындағанда құжаттарды басып шығару, қайта өңдеу, сақтау және мазмұнды үйлестіру керек. Бұл жоба факультеттердің жұмыс оқу жоспарына қатысты тіркеушінің кеңсесінің, бөлім басшыларының және деканаттың жұмысын автоматтандырады. Бұл жүйені пайдалану жоғары білім берудің жұмысын жақсартуға, уақытты үнемдеуге және кем дегенде бірнеше ағашты үнемдеуге көмектеседі. Ақпараттық жүйені енгізумен жұмыс жоспарларымен жұмыс істеу үрдістері жеңілдетіліп, белгілі бір уақытты талап ететін және назар аударуды қажет ететін кейбір іс-шаралар актерлерді анықтау және рөлдерді саралау арқылы жұмыс оқу жоспарларын басқару жүйесін құру арқылы автоматтандырылады.

ANNOTATION

Documents have to be printed, redone, stored somewhere and harmonize the contents, while having difficulty with the format of the document. This project automates the work of the office of the registrar, department heads and dean's office related to the working curriculum for faculties. Using this system will help improve the functioning of higher education, save time and save at least a few trees. With the introduction of the information system, the processes of working with work plans are simplified and some actions that require attention and take a certain amount of time will be automated by creating a work curriculum management system with defining the actors and differentiating roles.

СОДЕРЖАНИЕ

Введение	8
1 Теоретическая часть	9
1.1 Постановка задачи	9
1.2 Функциональные возможности	10
1.3 Предметная область	11
1.4 Технологии для автоматизации бизнес процесса	13
2 Проектная часть	16
2.1 Техническое задание	18
2.2 Логическая и физическая модели	19
2.3 Диаграммы последовательности действий	20
2.4 Диаграмма использования	22
2.5 Интерфейс приложения	23
Заключение	33
Словарь сокращений	34
Список использованной литературы	35
Приложение А	36
Приложение Б	37

ВВЕДЕНИЕ

В данной работе рассматриваются вопросы оптимизации бизнес процессов высших учебных заведений. В рамках таких заведений существует множество бизнес процессов, среди которых был определен процесс формирования рабочего учебного плана для автоматизации.

Рабочий учебный план - важная деталь в работе высшего учебного заведения. РУП формируется из набора дисциплин по восьми семестрам, с общим количеством в 129 кредитов для предоставления студентам возможности распределения дисциплин в свой учебный план на дальнейшие семестры. Кредиты - это единица измерения, которая определяет объем образовательной деятельности для студентов и преподавателей, значение кредита по времени составляет 3 часа обучения в неделю. 129 кредитов определено Министерством образования и науки как минимальное количество кредитов теоретического материала для получения степени бакалавра, утверждено приказом Министра образования и науки Республики Казахстан от 25 ноября 2014 года № 484 в разделе "Требования к максимальному объему учебной нагрузки обучающихся и воспитанников". Внедрение кредитной системы означает, что студенты должны сами выбирать себе дисциплины для учебного плана, которые они будут проходить в течении восьми семестров. Рабочий учебный план формируется работниками университета для студентов на базе образовательной программы по той или иной специальности.

Одной из целей является исключение бумажного документооборота, как одного из факторов замедления бизнес процесса. Документы приходится печатать, переделывать, где-то хранить и согласовывать содержимое, при этом возникают сложности с форматом документа. Данный проект автоматизирует часть работы офис регистратора, заведующих кафедры и деканата, связанной с рабочим учебным планом для факультетов. Использование этой системы поможет улучшить функционирование высшего учебного заведения, сэкономит время и спасет, как минимум, несколько деревьев. С внедрением информационной системы процессы работы с рабочими учебными планами упрощаются и некоторые действия, требовавшие внимание и отнимающие определенное количество времени, будут автоматизированы, путем создания системы управления рабочей учебной программой с определением действующих лиц и разграничением ролей.

Актуальность данной работы заключается в том, что аналогом может послужить только печать документов и их сопровождение вручную, что занимает много времени и замедляет внутренние бизнес процессы высшего учебного заведения. Данная дипломная работа избавит руководство университета и некоторых его работников от лишней работы.

1 Теоретическая часть.

1.1 Постановка задачи

Необходимо создать автоматизированное рабочее место управления рабочим учебным планом с определением действующих лиц, использующих систему и разграничением ролей. Инструмент должен позволять множеству пользователей одновременно осуществлять деятельность по созданию, утверждению или отклонению, наблюдению РУПа. Исходные данные для информационной системы — это академический год, состоящий из 2-х семестров и образовательная программа бакалавриата, рассчитанная на 4 года обучения. Каждый семестр может содержать в себе не более 24 кредитов, где каждый кредит это 1 академический час в неделю. Дисциплины могут быть как 2-х кредитными, так и 3-х кредитными, с разбивкой на виды контактных часов — лекция, практическое занятие и лабораторное занятие. Дисциплины в общем случае делятся — на общеобразовательные, базовые и профилирующие. Где в общеобразовательные дисциплины входя формирующие предметы высшего образования, такие как философия, культурология, логика, социология, политология, история и иностранные языки. Базовые дисциплины — это дисциплины ядра высшего образования по направлениям — технические, гуманитарные и другие. В рамках университета Satbayev University в них входят математика, физика, химия, а так же дисциплины конкретной образовательной программы, например по специальности Вычислительная техника и программное обеспечение — алгоритмы и структуры данных, архитектура вычислительных систем, базы данных, операционные системы и системное программирование, компьютерные сети. Профилирующие дисциплины — это дисциплины корпуса специализации студента, в рамках обучения которую получает студент, дисциплины — позволяющие определить траекторию развития знаний, навыков и компетенций. В такие дисциплины могут входить, например узко-направленные предметы как разработка мобильных приложений, разработка сетевых приложений, разработка компьютерных игр, машинное обучение и др. В системе так же должны учитываться условия пререквизитности для определения последовательности и взаимозависимости дисциплин, что является критичным при формировании плана развития студента. Система должна поддерживать модель выбора дисциплин по выбору и отдельный каталог дисциплин, которые могут быть выбраны студентом для формирования Индивидуального учебного плана.

Анализ существования подобных систем показал, таких как в США показал, что кредитная технология образования позволяет выпускать более гибких и адаптированных к изменениям специалистов в их сфере. На примере западных университетов с внедренной кредитной системой можно сказать, что она обеспечивает прозрачность функционирующих

национальных образовательных систем и благоприятно влияет на обучающихся. Минусы данной системы объясняются низким уровнем контроля над ходом обучения студентов и малым количеством аудиторных часов [1]. При создании инструмента главными отличительными особенностями должны быть простота и интуитивно-понятный интерфейс для конечных пользователей.

Процесс создания такой информационной системы включает в себя:

- 1) Исследование предметной области
- 2) Написание технического задания
- 3) Проектирование базы данных, в которой будет храниться информация об предметах, элективах, специальностях, РУПах.
- 4) Разработку реляционной базы данных SQL для хранения информации по объектам необходимых для составления рабочего учебного плана;
- 5) Разработку серверной части приложения в сетевом решении для обеспечения корректной работы системы при использовании ее несколькими пользователями;
- 6) Разработку интуитивно-понятного интерфейса для пользователей;
- 7) Обеспечение возможности сохранения документации по работе с формированием РУПа;

Конечными пользователями базы данных являются офис регистратор, заведующие кафедрой и деканат.

1.2 Функциональные возможности

Система должна выполнять следующие функции:

- Хранить рабочие учебные планы в электронном виде;
- Обеспечить доступ к электронным РУПам всем заинтересованным сторонам информационной системы;
- Обеспечить процесс создания, согласования и утверждения РУПа;
- Добавлять, редактировать, просматривать, удалять дисциплины;
- Добавлять, редактировать, просматривать, удалять элективные группы;
- Добавлять, редактировать, просматривать, удалять специальности;
- Добавлять, редактировать, просматривать, удалять РУПы, прототипы РУПов;
- Добавлять, редактировать, просматривать пользователей с разными ролями;

- Печатать завершённые РУПы;
- Откатывать данные – РУП можно изменить на любом этапе работы, с соответствующими правами;
- Вести полный отчет по времени и авторам на все действия.

1.3 Предметная область

Предметная область – деятельность университета, направленная на систематизирование предметов и элективов на восемь семестров для выбора их студентами. В рамках университета существуют институты, которые объединяют несколько кафедр с общим направлением. Каждая кафедра отвечает за выпуск специалистов по определенной специальности согласно общему классификатору. На текущий момент были приняты изменения в закон об образовании и ВУЗы могут строить образовательные программы независимо друг от друга.

Так как образовательные программы для получения соответствующего сертификата строятся независимо друг от друга в высшие учебные заведения была введена кредитная система. Кредитная система обучения - это система при которой обучающиеся студенты в определенных рамках могут планировать свою уникальную последовательность траектории изучения необходимых для окончания цикла дисциплин, которые измеряются в кредитах. Кредиты - это единица измерения, которая определяет объем образовательной деятельности для студентов и преподавателей, значение кредита по времени составляет 3 часа обучения в неделю, таким образом устанавливая для дисциплины объем учебной работы в 3 кредита подразумевает под собой 3 часов изучения этой дисциплины в неделю. Кредиты необходимы для разработки уникальной образовательной программы, путем распределения кредитов по семестрам на период обучения студента. При этом существуют нормативы по уровням образования, регламентирующие минимальное количество кредитов теоретического материала, который должен освоить студент для получения соответствующего сертификата окончания цикла, определенные Министерством образования и науки. Для бакалавров - это 129 кредитов теоретического курса. Где каждый кредит - это 1 час лекционного материала, либо 1 час практического занятия, или же 2 часа лабораторных работ. Таким образом дисциплины могут быть 2-х или 3-х кредитными, за редким исключением дисциплине может быть присвоено 1 или 4 кредита.

Основные цели системы кредитного обучения:

- А) Нормализация объема образовательной деятельности для студентов и преподавателей;
- Б) Унификация объема знаний студента;

В) Обеспечение возможности индивидуального планирования и организации учебного процесса;

Г) Увеличения значения самостоятельной работы и самообучения обучающегося.

Оценивание знаний по каждой дисциплине при кредитном обучении осуществляется по балльно-рейтинговой системе:

А) Текущий контроль - выполнение заданий (самостоятельные работы студента, лабораторные работы, отчеты про проделанной практике), посещаемость занятий и прочее. В среднем текущий контроль оценивается до трети баллов за аттестацию (от 15 до 20 баллов за аттестацию);

Б) Рубежный контроль, он осуществляется путем проведения на 8-ой и 15-ой неделе семестра контрольной работы или теста. В рубежный контроль оценивается до половины баллов за аттестацию (15 баллов за аттестацию);

В) Экзаменационная работа. Для получения допуска к экзаменационной работе требуется иметь минимальный проходной балл за совокупность двух аттестаций и минимальный порог посещаемости. Минимальный проходной балл - половина семестровых баллов (30 баллов), минимальный порог посещаемости - 80% посещения. Экзаменационная работа оценивается до 40% от общего количества баллов на дисциплину (40 баллов), при получении менее 20 баллов за экзаменационную работу дисциплина считается не пройденной и студент должен повторно ее пройти (retake).

После получения общего балла он конвертируется в единицу измерения успеваемости студента - GPA. GPA (Grade Point Average) - это среднее арифметическое значение от итоговых баллов дисциплин, полученных за все пройденные курсы, с учетом затраченного на них времени (количества кредитов).

Внедрение кредитной системы означает, что студенты должны сами выбирать себе дисциплины для учебного плана, которые они будут проходить в течении восьми семестров. Следовательно, для предоставления студентам возможности распределения дисциплин в свой учебный план на дальнейшие семестры сотрудники университета формируют рабочий учебный план.

1.4 Технологии для создания программного обеспечения автоматизации бизнес процесса

Для автоматизации обозначенного бизнес процесса были выбраны:

1) Сетевое решения для обеспечения корректной работы системы при использовании ее несколькими пользователями. Для этого был использован framework ASP.NET core 2 от компании Microsoft для написания серверной составляющей, поддерживающий работу с асинхронными вызовами, что делает его оптимальным решением данной задачи. ASP.NET core имеет эффективный модульный генератор HTTP запросов, возможность добавления пакетов через NuGet, единый набор для разработки веб-приложения с пользовательским интерфейсом (Web UI) и Web API, расширяемость, открытость и кроссплатформенность. NuGet дает доступ к таким пакетам как:

- Entity framework core – инструмент от компании Microsoft для разработки хранилища данных путем создания ее через код (code first), работающий с реляционной базой данных SQLite, используемой как основная база данных для хранения объектов, требующих связи между друг другом. Удобна в сочетании с Entity framework. Главное преимущество Entity framework core от других инструментов отображения данных на объектах в поддержке LINQ запросов, с их помощью можно легко совершать выборку объектов из базы данных, даже с набором ассоциативных связей, тем самым отправляя на клиент через HTTP запрос лишь ту информацию, которая необходима для отображения на сайте, ускоряя его работу;

- ASP.NET Identity – система для аутентификации, авторизации и разграничения доступа внутри проекта. Использование этой системы сильно экономит время и застраховывает разработчика от ошибок, которые он может допустить в системе безопасности. В отличии от того же Membership API, который работал в связке с ASP.NET ранее, ASP.NET Identity не имеет проблем с расширением и сложности написания, также ASP.NET Identity не привязан к реляционным базам данных.

В сравнении с ASP.NET и другими синхронными веб-инфраструктурами ASP.NET core обрабатывает запросы асинхронно. Паттерн `async\await` дает возможность пометить каждый обработчик запросов как асинхронный, вследствие чего работа системы ввода-вывода будет производиться в своем потоке, это позволяет не блокировать основной поток. Если сравнивать ASP.NET core с другими асинхронными веб-инфраструктурами, такими как Node.JS, то у последнего несмотря на обширное сообщество разработчиков отсутствует проверенная команда профессиональных разработчиков, которые имеются у ASP.NET core. Хотя Node.JS более удобен в выборе модулей для разработки, но некачественное обновление одного пакета может привести к поломке приложения. Также ASP.NET core использует строго-типизированный объектно-ориентированный язык программирования C#, который обеспечивает

типовую безопасность переменных и удобство отлаживания. Помимо этого ASP.NET core обладает лучшей производительностью приложений, нежели Node.JS и является одним из лучших решений для объемных приложений, предоставляя разработчику инструментарий высшего качества [2].

2) Angular - framework от компании Google, обладающий широким спектром возможностей для написания реактивного веб-приложения. Angular реализует подход single page application (SPA) для разработки клиентской части, который обеспечивает быстрое действие сайта при переходе между вкладками. Помимо этого, создание проектов на Angular производится на объектно-ориентированном языке программирования TypeScript со строгой типизацией типов данных. На сегодняшний день является наиболее перспективным решением. Выбор Angular очевиден в комбинации с C#, так как синтаксис этих языков крайне схож. Angular имеет встроенный модуль HttpClientModule для работы с протокол HTTP для передачи данных между клиентов и сервером. Hypertext transfer protocol – это протокол прикладного уровня, седьмого уровня сетевой модели OSI, передает данные путем «запрос-ответ». Всего протокол HTTP предоставляет девять методов передачи данных, такие как: OPTIONS, HEAD, GET, POST, PUT, DELETE, PATCH, TRACE, CONNECT. Наиболее распространенные запросы – это GET и POST, первый служит для получения клиентом информации в теле запроса на отправленное сообщение, а второй для отправки от клиента к заданному ресурсу сообщения с произвольными данными и получения результата запроса.

Angular в отличии от своего предшественника AngularJS реализует подход SPA и работает с компонентами. Компонент работает по принципу ввода-вывода: получает данные на вход, обрабатывает их и возвращает отрендеренный шаблон пользовательского интерфейса. Также с компонентами работают Vue и React. Но несмотря на это, у Angular есть преимущества и перед ними в виде строго-типизированный объектно-ориентированный язык программирования TypeScript. React использует шаблон JSX, в то время как Angular использует модифицированный HTML со специальным языком Angular, использующий структурные директивы (ngFor - перебрать в шаблоне элементы массива, ngIf - отображение элемент при определенном условии и т.д.). В противовес AngularJS, Vue и React Angular является фреймворком, что делает приложение структурированным и дает возможность не проводить анализ библиотек [3].

3) Нереляционная база данных MongoDB, используется как вспомогательная БД для хранения объектов, не нуждающихся в связях, в данном случае документ РУПа. В противоположности от реляционных баз данных, таких как SQL, MongoDB осуществляет хранение информации путем документо-ориентированной модели, вследствие чего база данных на MongoDB работает быстрее, проще в использовании и лучше масштабируется. В отличии от MySQL, SQLite, PLSQL, которые использует таблицы, MongoDB работает с коллекциями, отличие коллекций от таблиц в

том, что таблицы хранят только однотипные строго структурированные данные, а коллекции содержат различные объекты с различной структурой и набором свойств. Основное преимущество MongoDB в сравнении с SQL, это то, что он работает с гибкими JSON-формат документами [\[4\]](#).

2 Проектная часть.

2.1 Техническое задание

2.1.1 Наименование системы

«Система автоматизирования работы с рабочим учебным планом».

2.1.2 Назначение системы

Система представляет собой программно-аппаратный комплекс, автоматизирующий рабочее место управления рабочей учебной программой с определением действующих лиц, использующих систему и разграничением ролей. Инструмент должен позволять множеству пользователям системы одновременно осуществлять деятельность по формированию, изменению, удалению, утверждению или отклонению, наблюдению и просмотру, распечатыванию рабочего учебного плана, с возможностью отслеживания проделанной деятельности по нему. Создавать, просматривать, редактировать и удалять дисциплины с различными количеством кредитов (от 1 до 4), типом предмета (общеобразовательные, базовые, профилирующие). Создавать, просматривать, редактировать и удалять специальности.

2.1.3 Область применения системы

Область применения – деятельность университета, направленная на систематизирование дисциплин на восемь семестров для выбора их студентами. В рамках университета существуют институты, которые объединяют несколько кафедр с общим направлением. Каждая кафедра отвечает за выпуск специалистов по определенной специальности согласно общему классификатору. На текущий момент были приняты изменения в закон об образовании и ВУЗы могут строить образовательные программы независимо друг от друга.

2.1.4 Требования к функциональным характеристикам

Программная часть должна обеспечивать возможность выполнения перечисленных ниже функций:

2.1.4.1 Разграничение функционала для подключаемых пользователей.

А) Заведующие кафедрами:

- Добавлять, редактировать, просматривать, удалять дисциплины;
- Добавлять, редактировать, просматривать, удалять элективные группы;
- Добавлять, редактировать, просматривать, удалять специальности;
- Добавлять, редактировать, просматривать, удалять рабочие учебные планы;
- Наследовать данные из прототипа рабочего учебного плана в свой РУП;
- Отправлять рабочие учебные планы на согласование;
- Выводить сводную таблицу по рабочему учебному плану;
- Печатать завершённые рабочие учебные планы;

Б) Офис регистратора и деканат:

- Просматривать рабочие учебные планы;
- Просматривать подтвержденные рабочие учебные планы;
- Добавлять, редактировать, просматривать, удалять прототипы рабочего учебного плана;
- Добавлять, редактировать, просматривать пользователей с разными ролями;
- Подтверждать рабочие учебные планы;
- Выводить сводную таблицу по рабочему учебному плану;
- Печатать завершённые рабочие учебные планы;
- Откатывать данные по рабочим учебным планам – РУП можно вернуть из состояния «подтвержденного» на любом этапе работы;
- Отправлять документы по подтвержденным рабочим учебным планам в картотеку.

2.1.4.2 Предоставление возможности пользователям одновременно осуществлять деятельность по работе с системой.

2.1.4.3 Обеспечение безопасности личной информации пользователей.

2.2 Логическая и физическая модели базы данных

2.2.1 Описание логической и физической модели

Логическая и физическая модель данных бизнес процесса. Логическая модель показывает какие сущности, атрибуты сущности и связи имеются в базе данных. Логический уровень представляет данные в том виде, в котором они выглядят в реальном мире, сущности имеют те же наименования.

Физическая модель абстрагируют программные продукты. Она зависит от конкретной СУБД, технически описывая системный каталог. В физической модели содержится информация о всех объектах базы данных.

Использование SQL дает возможность не обращать внимание на то, где хранятся эти байты информации и выделенной для них памяти.

2.2.2 Логическая модель базы данных

Логическая модель включает в себя восемь сущностей: пользователь, роль пользователя, предмет, элективная группа, тип дисциплины, специальность, семестр и рабочий учебный план. Она показана на рисунке 1.

А) Сущность «Пользователь» имеет атрибуты id, имя и пароль, где id - это первичный ключ. Сущность «Пользователь» имеет связь один ко многим с таблицами: роль, специальность, рабочий учебный план, предмет, элективная группа.

Б) Сущность «Роль» имеет атрибуты id, название, где id - это первичный ключ. Сущность «Роль» имеет связь один ко многим с таблицей пользователя.

В) Сущность «Предмет» имеет атрибуты id, название, количество кредитов, шифр, кол-во лекционных часов, кол-во лабораторных часов, кол-во часов практики, дата добавления, дата последнего обновления, где id - это первичный ключ. Сущность «Предмет» имеет связь один ко многим с таблицами: тип предмета, пользователь и связь многие ко многим с таблицами семестр, предмет, элективная группа.

Г) Сущность «Элективная группа» имеет атрибуты id, название, количество кредитов, шифр, кол-во часов практики, дата добавления, дата последнего обновления, где id - это первичный ключ. Сущность «Элективная группа» имеет связь один ко многим с таблицами: тип предмета, пользователь и связь многие ко многим с таблицами семестр, предмет.

Д) Сущность «Тип предмета» имеет атрибуты id, название, код, цвет, где id - это первичный ключ. Сущность «Тип предмета» имеет связь один ко многим с таблицами: предмет, элективная группа.

Е) Сущность «Специальность» имеет атрибуты id, название, шифр, дата добавления, дата последнего обновления, где id - это первичный ключ. Сущность «Специальность» имеет связь один ко многим с таблицами: пользователь, рабочий учебный план.

Ж) Сущность «Семестр» имеет атрибут id (первичный ключ), имеет связь многие ко многим с таблицами: предмет, элективная группа, РУП.

3) Сущность «РУП» имеет атрибуты id, год, специальность, дата добавления, дата последнего обновления, где id - это первичный ключ. Сущность «РУП» имеет связь один ко многим с таблицами: специальность, пользователь и связь многие ко многим с таблицей семестра.

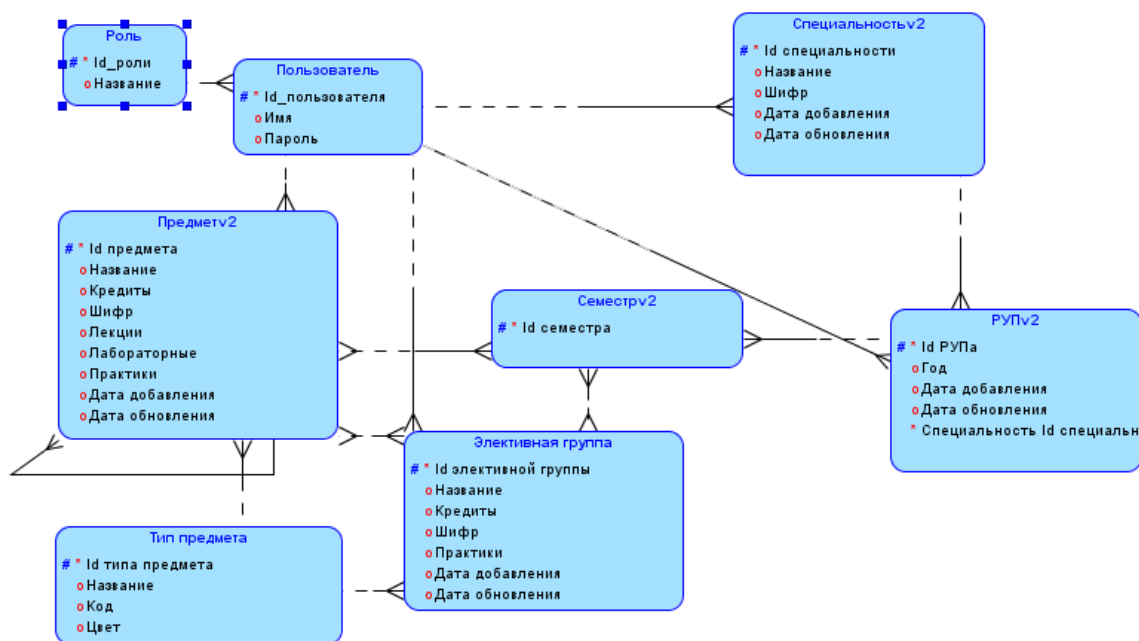


Рисунок 1. Логическая модель базы данных

2.2.3 Физическая модель базы данных

Физическая модель содержит информацию о всех объектах базы данных. В физической модели показываются промежуточные таблицы, внешние ключи и типы данных атрибутов. Физическая модель включает в себя восемь основных таблиц: пользователь, роль пользователя, предмет, элективная группа, тип дисциплины, специальность, семестр и рабочий учебный план, и четыре промежуточные таблицы: предмет-пререквизит, предмет-элективная группа, семестр- элективная группа и семестр-предмет. Она представлена на рисунке 2.

А) Таблица «Пользователь» имеет атрибуты id (number), имя (varchar) и пароль (varchar), где id - это первичный ключ.

Б) Таблица «Роль» имеет атрибуты id (number), название (varchar), где id - это первичный ключ.

В) Таблица «Предмет» имеет атрибуты id (number), название (varchar), количество кредитов (number), шифр (varchar), кол-во

лекционных часов (number), кол-во лабораторных часов (number), кол-во часов практики (number), дата добавления (date), дата последнего обновления (date), где id - это первичный ключ.

Г) Таблица «Элективная группа» имеет атрибуты id (number), название (varchar), количество кредитов (number), шифр (varchar), кол-во часов практики (number), дата добавления (date), дата последнего обновления (date), где id - это первичный ключ.

Д) Таблица «Тип предмета» имеет атрибуты id (number), название (varchar), код (varchar), цвет (varchar), где id - это первичный ключ.

Е) Таблица «Специальность» имеет атрибуты id (number), название (varchar), шифр (varchar), дата добавления (date), дата последнего обновления (date), где id - это первичный ключ.

Ж) Таблица «Семестр» имеет атрибут id (number) - первичный ключ.

З) Таблица «РУП» имеет атрибуты id (number), год (varchar), специальность (varchar), дата добавления (date), дата последнего обновления (date), где id - это первичный ключ.

И) Промежуточная таблица «Предмет_Пререквизит» имеет внешние ключи: id предмета 1 (number), id предмета 2 (number).

К) Промежуточная таблица «Элективная группа_Предмет» имеет внешние ключи: id элективной группы (number), id предмета (number).

Л) Промежуточная таблица «Семестр_Предмет» имеет внешние ключи: id семестра (number), id предмета (number).

М) Промежуточная таблица «Семестр_Элективная группа» имеет внешние ключи: id семестра (number), id элективной группы (number).

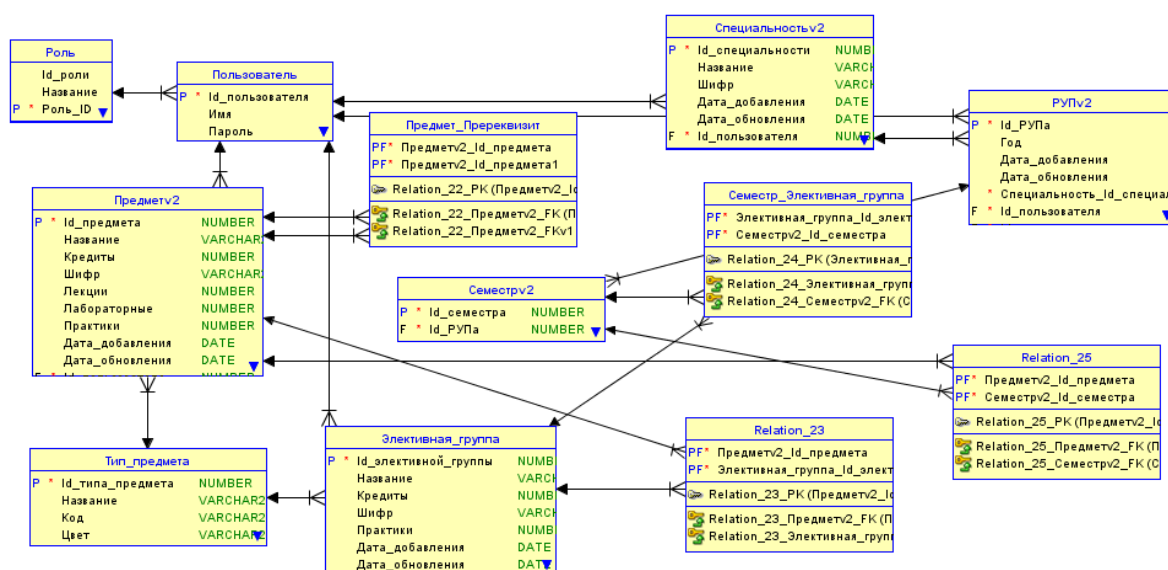


Рисунок 2. Физическая модель базы данных

2.3 Диаграммы активности

2.3.1 Диаграмма активности оформления рабочего учебного плана.

Диаграмма активности оформления рабочего учебного плана, заведующим кафедрой и его подтверждения деканатом, представлена на рисунке 3.

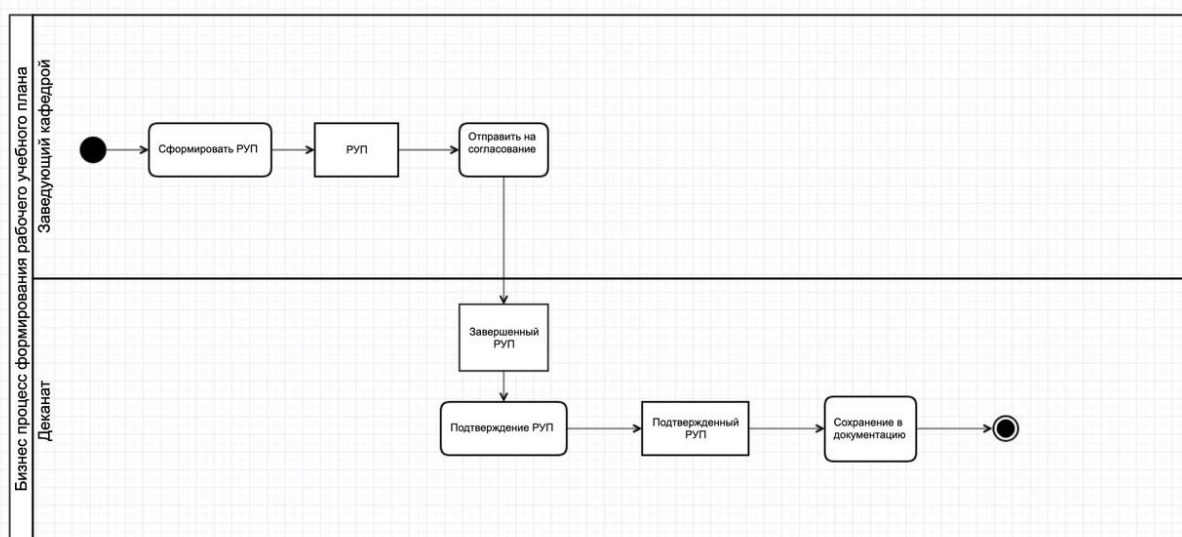


Рисунок 3. Диаграмма активности оформления рабочего учебного плана

2.3.2 Диаграмма активности оформления рабочего учебного плана через прототип РУПа.

Диаграмма активности оформления рабочего учебного плана через прототип РУПа представлена на рисунке 4.

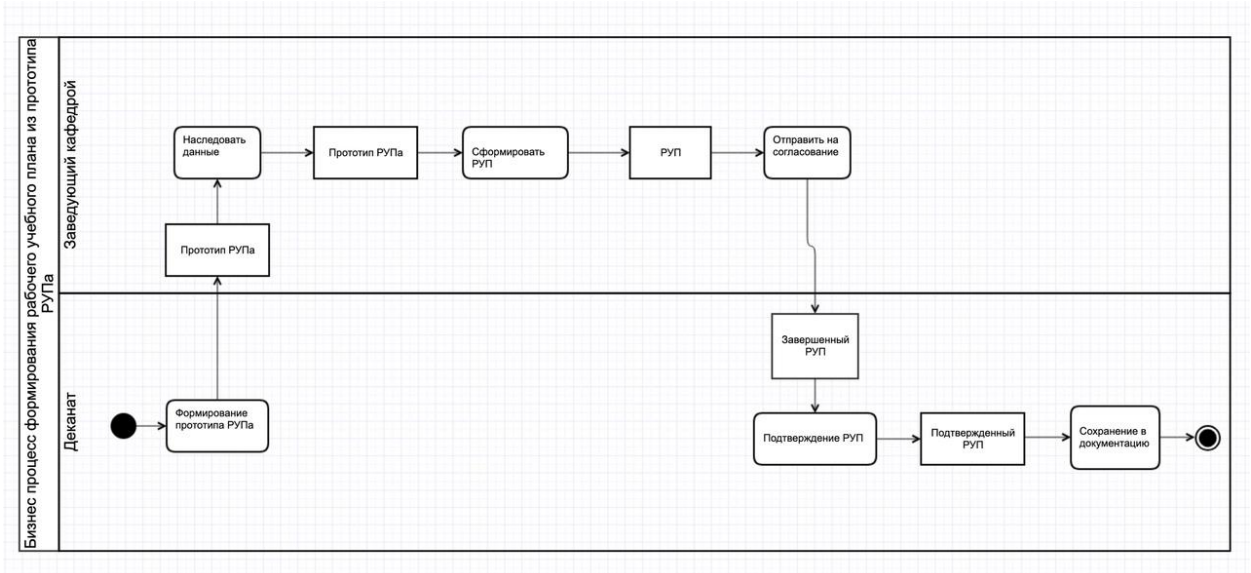


Рисунок 4. Диаграмма активности оформления рабочего учебного плана через прототип РУПа

2.4 Диаграмма использования

Диаграмма использования бизнес процесса формирования рабочего учебного плана, заведующим кафедрой и деканатом представлена на рисунке 5.

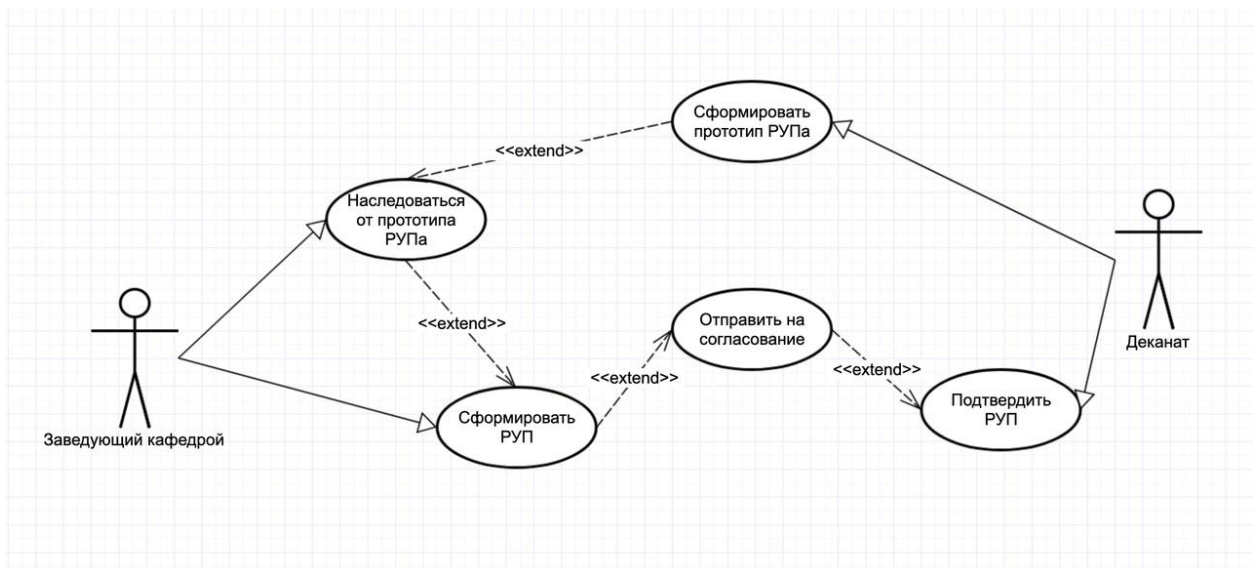


Рисунок 5. Диаграмма использования

2.5 Интерфейс приложения

Интерфейс приложения состоит из страниц, которые включают в себя: страницу авторизации, страницу списков рабочих учебных планов - главную страницу, страницу добавления/редактирования РУПа, страницу списка дисциплин, страницу добавления/редактирования предметов, страницу добавления/редактирования элективных групп, страницу списка специальностей, страницу добавления/редактирования специальностей, страницу списка пользователей, страницу добавления/редактирования пользователей.

1) Страница авторизации включает в себя форму для заполнения с обязательными полями имени пользователя, пароля и кнопку входа в систему. Страница представлена на рисунке 6.

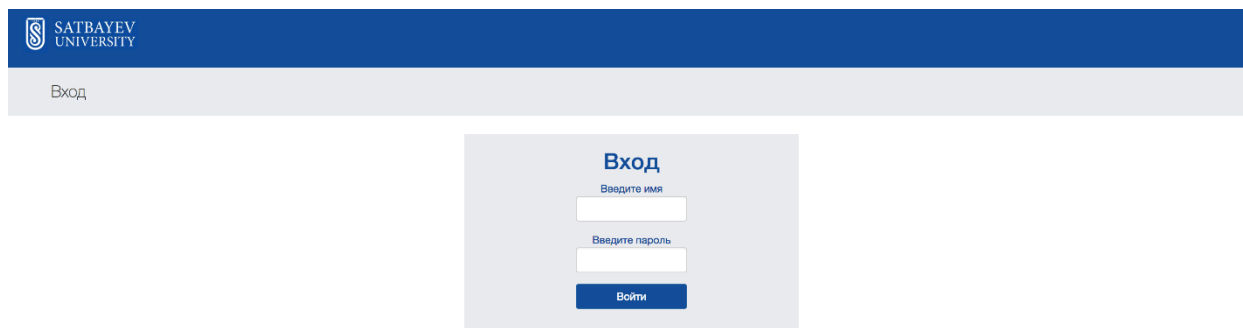


Рисунок 6. Страница при запуске приложения

2) Страница списка рабочих учебных планов содержит 3 таблицы РУПов, которые в зависимости от роли авторизованного пользователя имеют различный набор опций. Отображение страницы и ее функционала для заведующих кафедрами представлено на рисунке 7. Отображение страницы и ее функционала для деканата представлено на рисунке 8.

SATBAYEV UNIVERSITY РУПы Предметы Специальности operator Выйти

РУПы [+](#) Добавить РУП

Список прототипов РУПов			
Шифр сп.	Специальность	Год	Операции
5B070200	Информатика	2018-2019	Наследоваться
5B070400	Вычислительная техника и программное обеспечение	2018-2019	Наследоваться

Список РУПов			
Шифр сп.	Специальность	Год	Операции
5B070400	Вычислительная техника и программное обеспечение	2019-2020	Изменить Удалить Подтвердить

Рисунок 7. Список РУПов для заведующих кафедрой.

SATBAYEV UNIVERSITY РУПы Пользователи mg Выйти

РУПы [+](#) Добавить прототип РУПа

Список прототипов РУПов			
Шифр сп.	Специальность	Год	Операции
5B070200	Информатика	2018-2019	Изменить Удалить
5B070400	Вычислительная техника и программное обеспечение	2018-2019	Изменить Удалить

Список РУПов			
Шифр сп.	Специальность	Год	Операции
5B070400	Вычислительная техника и программное обеспечение	2019-2020	-

Список закрытых РУПов			
Шифр сп.	Специальность	Год	Операции
5B070400	Вычислительная техника и программное обеспечение	2018-2019	-
5B070400	Вычислительная техника и программное обеспечение	2018-2019	-
5B070400	Вычислительная техника и программное обеспечение	2018-2019	-
5B070400	Вычислительная техника и программное обеспечение	2018-2019	-
5B070400	Вычислительная техника и программное обеспечение	2018-2019	Открыть

Рисунок 8. Список РУПов для деканата.

3) Страница добавления РУПа/прототипа РУПа содержит форму с обязательными полями специальность и учебного года, таблицу с отсортированным по типам общее количество кредитов на период обучения, форма для распределения дисциплин по восьми семестрам, содержащая кнопку «Добавить дисциплину», которое открывает модальное окно со списком дисциплин, с возможностью поиска по названию дисциплины, для добавления в выбранный семестр. Процесс проиллюстрирован на рисунках 9-11.

SATBAYEV UNIVERSITY РУПы Предметы Специальности operator Выйти

РУП Специальность: 5B070200 - Информатика Уч. год: 2018-2019

5B070200 - Информатика 2018-2019 уч. год

Форма обучения: Дневная Срок обучения: 4 года Академическая ступень: Бакалавр технических наук

Количество кредитов за весь период обучения

циклы дисциплины	ОБЯЗАТЕЛЬНЫЕ	ЭЛЕКТИВНЫЕ	ВСЕГО
● Цикл общеобразовательных дисциплин (О)	0	0	0
● Цикл базовых дисциплин (Б)	0	0	0
● Цикл профилирующих дисциплин (П)	0	0	0
● Профессиональная практика (ДВО)	6	—	6
● Итоговая аттестация (ИА)	3	—	3
Итого	9	0	9

1 СЕМЕСТР		2 СЕМЕСТР		3 СЕМЕСТР		4 СЕМЕСТР		5 СЕМЕСТР		6 СЕМЕСТР		7 СЕМЕСТР		8 СЕМЕСТР	
№	Код	Наименование дисциплины				Комп.	Кред.	лк/лб/лр	Пререквизит						
Всего кредитов за семестр: 0															

[+](#) добавить дисциплину

[Вернуться к списку](#) [Сохранить изменения](#)

Рисунок 9. Страница добавление РУПа/прототипа РУПа.

Количество кредитов за весь период обучения

циклы дисциплины	ОБЯЗАТЕЛЬНЫЕ	ЭЛЕКТИВНЫЕ	ВСЕГО
● Цикл общеобразовательных дисциплин (О)	0	0	0
● Цикл базовых дисциплин (Б)	0	0	0
● Цикл профи			0
● Профессио			6
● Итоговая а			3
Итого			9

Добавление дисциплины

Основные дисциплины Элективные группы

Поиск

Код	Наименование дисциплины	Комп.	Кред.	лк/лб/пр
123	Subj1	○	3	1/1/1
123	Subj2	○	3	2/1/0

Отменить Добавить

Всего кредитов за семестр: 0 + ДОБАВИТЬ ДИСЦИПЛИНУ

Вернуться к списку
Сохранить изменения

Рисунок 10. Окно выбора дисциплин в семестр.

Количество кредитов за весь период обучения			
циклы дисциплины	ОБЯЗАТЕЛЬНЫЕ	ЭЛЕКТИВНЫЕ	ВСЕГО
● Цикл общеобразовательных дисциплин (О)	9	9	18
● Цикл базовых дисциплин (Б)	3	3	6
● Цикл профилирующих дисциплин (П)	7	0	7
● Профессиональная практика (ДВО)	6	—	6
● Итоговая аттестация (ИА)	3	—	3
Итого	28	12	40

1 СЕМЕСТР		2 СЕМЕСТР		3 СЕМЕСТР		4 СЕМЕСТР		5 СЕМЕСТР		6 СЕМЕСТР		7 СЕМЕСТР		8 СЕМЕСТР	
№	Код	Наименование дисциплины			Комп.	Кред.	лк/лб/пр	Пререквизит							
1	123	Subj3			О	3	2/0/1	да		✕					
2	123	Subj4			Б	3	1/1/1	да		✕					
3	123	Subj5			П	4	2/1/1	да		✕					
4	123	Еlec3			О	3	0/0/3	да		✕					
5	123	Еlec1			Б	3	0/0/3	да		✕					

Всего кредитов за семестр: **16** ⊕ ДОБАВИТЬ ДИСЦИПЛИНУ

Вернуться к списку
Сохранить изменения

Рисунок 11. Форма семестра после выбора дисциплин.

4) Страница списка дисциплин, на ней представлены 2 таблицы с функциональными кнопками: таблица предметов, таблица элективных групп. Отображение страницы представлено на рисунке 12.

SATBAYEV UNIVERSITY РУПы Предметы Специальности operator Выйти

Предметы ⊕ Добавить предмет ⊕ Добавить элективную группу

Список предметов

№	Код	Наименование дисциплины	Комп.	Кред.	лк/лб/пр	Пререквизит	Уд.	Ред.
1	123	Subj1	О	3	1/1/1	нет		
2	123	Subj2	О	3	2/1/0	нет		
3	123	Subj3	О	3	2/0/1	да		
4	123	Subj4	Б	3	1/1/1	да		
5	123	Subj5	П	4	2/1/1	да		
6	123	Subj6	П	3	1/2/0	да		

Список элективных групп

№	Код	Наименование дисциплины	Комп.	Кред.	лк/лб/пр	Пререквизит	Уд.	Ред.
1	123	Еlec1	Б	3	0/0/3	да		
2	123	Еlec2	О	3	0/0/3	нет		
3	123	Еlec3	О	3	0/0/3	да		
4	123	Еlec5	О	3	0/0/3	нет		

Рисунок 12. Список предметов и элективных групп.

5) Страница добавления предмета содержит форму с обязательными полями: название предмета, шифр предмета, кол-во лекционных часов, кол-во лабораторных часов, кол-во часов практики, тип дисциплины (базовый, общеобразовательный, профилирующий), список пререквизитов – дисциплины, которые должны быть пройдены до выбора этого предмета. Отображение страницы представлено на рисунке 13.

SATBAYEV UNIVERSITY РУПы Предметы Специальности operator Выйти

Добавить предмет

Название

Шифр

Количество лекций

Количество лабораторных

Количество практик

Тип

Пререквизиты

[⊕ ДОБАВИТЬ ПРЕРЕКВИЗИТ](#)

Рисунок 13. Добавление предмета.

6) Страница добавления элективной группы содержит форму с обязательными полями: название, шифр, список предметов – дисциплины, составляющие элективную группу. В зависимости от выбранного предмета будут определены количество кредитов и тип дисциплины для элективной группы и последующего выбора предметов в группу. Отображение страницы представлено на рисунке 14.

Скриншот веб-интерфейса для добавления элективной группы. Вверху логотип SATBAYEV UNIVERSITY и меню: РУПы, Предметы, Специальности. В правом верхнем углу 'operator' и 'Выйти'. Заголовок страницы: 'Добавить элективную группу'. Форма содержит:

- Поле 'Название' (Name)
- Поле 'Шифр' (Code)
- Кнопка '+ ДОБАВИТЬ ЭЛЕКТИВ' (Add Elective)
- Кнопка 'Сохранить' (Save)
- Кнопка 'К списку' (To list)

Рисунок 14. Добавление элективной группы.

7) Страница списка специальностей включает в себя таблицу с функциональными кнопками, отображающую основную информацию по специальностям. Изображение страницы представлено на рисунке 15.

Скриншот веб-интерфейса для списка специальностей. Вверху логотип SATBAYEV UNIVERSITY и меню: РУПы, Предметы, Специальности. В правом верхнем углу 'operator' и 'Выйти'. Заголовок страницы: 'Специальности'. Кнопка '+ Добавить специальность' (Add Specialty). Таблица 'Список специальностей':

Номер сп.	Шифр	Специальность	Операции
1	5B070400	Вычислительная техника и программное обеспечение	Изменить Удалить
2	5B070200	Информатика	Изменить Удалить

Рисунок 15. Список специальностей.

8) Страница добавления специальности содержит форму с обязательными полями: название, шифр. Отображение страницы представлено на рисунке 16.

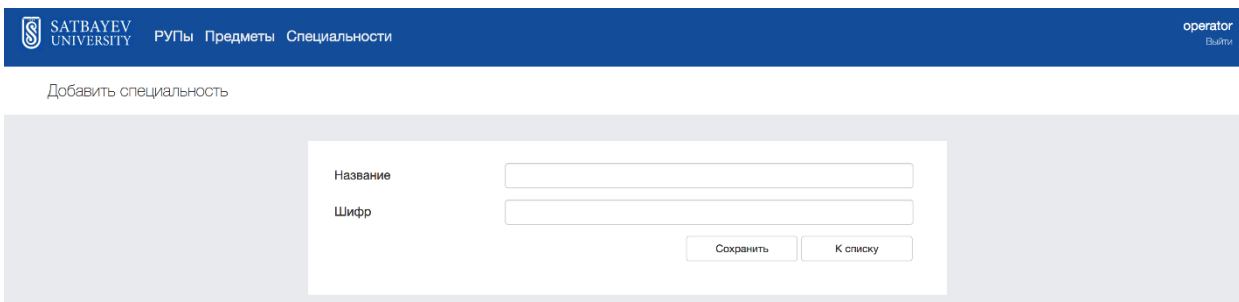
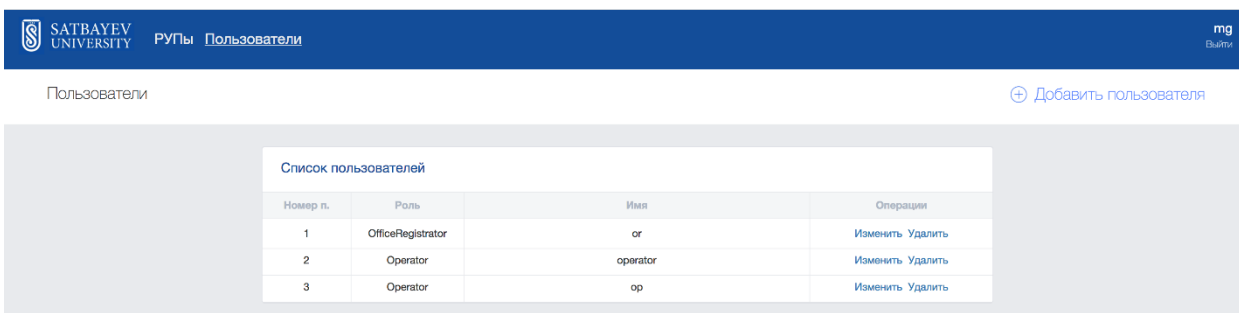


Рисунок 16. Добавление специальности.

9) Страница списка пользователей, доступна только для деканата, содержит таблицу пользователей с информацией по их роли, имени пользователя и операциями над ними. Отображение страницы представлено на рисунке 17.



Номер п.	Роль	Имя	Операции
1	OfficeRegistrar	or	Изменить Удалить
2	Operator	operator	Изменить Удалить
3	Operator	op	Изменить Удалить

Рисунок 17. Список пользователей.

10) Страница добавления пользователя содержит форму с обязательными полями: имя, пароль, роль пользователя. Отображение страницы представлено на рисунке 18.

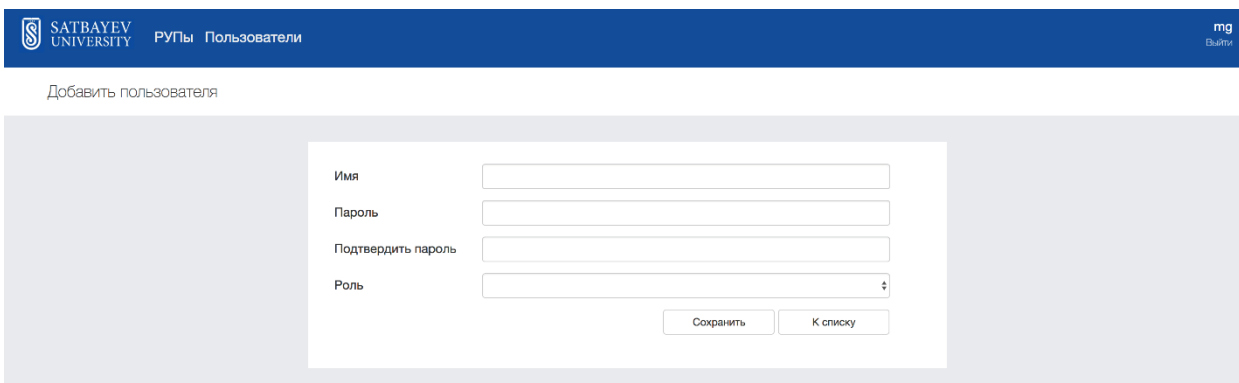



Рисунок 18. Добавление пользователя.

11) Страница сводной таблицы. На ней изображен рабочий учебный план в виде документа формата А4, содержащий подробную информацию по РУПу и каждому его семестру, такие как количество кредитов и информация по выбранным дисциплинам. Нужна для дальнейшей распечатки РУПа. Страница показана на рисунке 19. Распечатка показана на рисунке 20.

УЧЕБНЫЙ ПЛАН СПЕЦИАЛЬНОСТИ для набора на 2018-2019 уч. год						5В070200 - Информатика							
Форма обучения: Дневная			Срок обучения: 4 года			Академическая степень: Бакалавр техники и технологий							
Код	Наименование дисциплины		Ком-по-нент	Кре-диты	лк/лаб/пр	Пререк-визит	Код	Наименование дисциплины		Ком-по-нент	Кре-диты	лк/лаб/пр	Пререк-визит
1 семестр						2 семестр							
123	Subj1		О	3	1/1/1	нет	123	Subj3		О	3	2/0/1	да
123	Subj2		О	3	2/1/0	нет	123	Subj4		Б	3	1/1/1	да
123	Subj7		О	2	1/1/0	нет	123	Subj5		П	4	2/1/1	да
Всего:				8			Всего:				8		
3 семестр						4 семестр							
123	Subj6		П	3	1/2/0	да	123	Elec2		О	3	0/0/3	да
123	Elec1		Б	3	0/0/3	да	123	Elec3		О	3	0/0/3	да
Всего:				6			Всего:				6		
5 семестр						6 семестр							
Всего:				0			Всего:				0		
7 семестр						8 семестр							
Всего:				0			Всего:				0		
Дополнительные виды обучения					Количество кредитов за весь период обучения								
Год	Код	Наименование	Кре-диты	Семестр	Наименование	Обяз.	Элект.	Всего					
1	ААР106	Физическая культура I	0	1	Цикл общеобразовательных дисциплин (О)	20	0	20					
1	ААР108	Физическая культура II	0	2	Цикл базовых дисциплин (Б)	6	0	6					
1	ААР101	Учебная практика	2	2	Цикл профилирующих дисциплин (П)	7	0	7					
2	ААР109	Производственная практика I	1	4	Всего по теоретическому обучению:	42	0	42					
3	ААР103	Политехническая практика II	1	4									

Рисунок 19. Сводная таблица.

01.05.2018
RU
mg
1


mg

РУПы (f) Пользователи (userList)

УЧЕБНЫЙ ПЛАН СПЕЦИАЛЬНОСТИ для набора на 2018-2019 уч. год
 08070200 - Информатика

Формы обучения: Дневная Срок обучения: 4 года

Код	Наименование дисциплины	Экз.	Зач.	Средний балл	Период	Всего	Информатика		
							Экз.	Зач.	
1 семестр									
123	Инф1	0	3	3,0000	лет				
123	Инф2	0	3	2,0000	лет				
123	Инф3	0	3	3,0000	лет				
Всего:							0	9	
2 семестр									
123	Инф4	0	3	3,0000	лет				
123	Инф5	0	3	3,0000	лет				
Всего:							0	6	
3 семестр									
123	Инф6	0	3	3,0000	лет				
123	Инф7	0	3	3,0000	лет				
Всего:							0	6	
4 семестр									
123	Инф8	0	3	3,0000	лет				
123	Инф9	0	3	3,0000	лет				
Всего:							0	6	
5 семестр									
6 семестр									
7 семестр									
8 семестр									

Академические степени: Бакалавр техники и технологий

Код	Наименование дисциплины	Экз.	Зач.	Средний балл	Период	Всего	Информатика		
							Экз.	Зач.	
2 семестр									
123	Инф10	0	3	3,0000	лет				
123	Инф11	0	3	3,0000	лет				
123	Инф12	0	3	3,0000	лет				
Всего:							0	9	
3 семестр									
4 семестр									
5 семестр									
6 семестр									
7 семестр									
8 семестр									

<http://ksaibon44200.ru/mg> 1/3

Печать
Всего: 3 стр.

Отмена Сохранить

Сохранить как PDF
Изменить

Принтер

Страницы: Все
 Например: 1-5, 8, 11-13

Раскладка:

+ Дополнительные настройки

Печатать с помощью системного диалогового окна (Ctrl+P)
[Открыть PDF для просмотра](#)

Рисунок 20. Печать РУПа.

ЗАКЛЮЧЕНИЕ

В дипломном проекте разработан веб-сервис для сотрудников университета для работы с рабочим учебным планом. Система предоставляет возможность для заведующих кафедрами, офис регистратора и деканата добавлять, просматривать, редактировать, удалять записи, связанные с рабочим учебным планом. Так же он может выполнять печать, что значительно упрощает подгонку под один формат, так как все распечатки будут создаваться по определенному шаблону.

В системе были продуманы такие детали как, обновление дисциплины, входящей в реальный учебный план: при обновлении дисциплины, которая входит в не подтвержденный рабочий учебный план, она обновляется и внутри него, но если рабочий учебный план на определенный год и специальность уже подтвержден, дисциплина внутри него останется неизменной. Это сделано для того, чтобы изменение дисциплины на последующие года не влияли на рабочие учебные планы, по которым студенты уже обучаются или закончили обучение. Также было продумано, что при удалении объекта на него устанавливается флаг, с помощью которого они не выводятся в список для отображения. Данная установка была проделана с целью обеспечения отказоустойчивости системы.

Главное достоинство этого проекта в том, что при необходимости его можно масштабировать, так как в нем уже реализованы основные принципы и функционал работы ВУЗа с внедренной кредитной системой образования.

Считаю, что проект соответствует всем современным требованиям, благодаря использованию передовых технологий для его разработки: ASP.NET core и entity framework сделали проект структурированным, отказоустойчивым и надежным.

Использование этой системы поможет улучшить функционирование высшего учебного заведения, сэкономит много времени и спасет, как минимум, несколько деревьев. С внедрением информационной системы процессы работы с РУПами упрощаются и некоторые действия, требовавшие внимание и отнимающее много времени, будут автоматизированы.

СЛОВАРЬ СОКРАЩЕНИЙ

1. РУП - рабочий учебный план
2. ВУЗ – высшее учебное заведение
3. БД - база данных
4. SPA - single page application
5. ASP - active server pages

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Сервер ASP.NET core - <https://metanit.com/sharp/aspnet5/>
2. Основная база данных entity framework core - <https://metanit.com/sharp/entityframeworkcore/>
3. Пользовательский интерфейс Angular - <https://metanit.com/web/angular2/>
4. Нереляционная база данных MongoDB - <https://metanit.com/nosql/mongodb/>

Приложение А

Весь проект с базами данных и кодом можно скачать по ссылке:

- Клиент: https://github.com/ArsenKaratayev/ryp_client, для запуска откройте репозиторий в консоле (терминале) и напишите команду `npm start`;
- Сервер: <https://github.com/ArsenKaratayev/project>, для запуска откройте Project.sln с помощью Visual studio или отрыв репозиторий в консоле (терминале) и написать команду `dotnet run`.

Приложение Б

Здесь представлены модели базы данных системы.

Тип предмета

```
public class SubjectType
{
public int Id { get; set; }
public string Name { get; set; }
public string Code { get; set; }
public string Color { get; set; }

public virtual List<Subject> Subjects { get; set; }
}
```

Предмет_пререквизит

```
public class SubjectPrerequisiteSubject
{
public int Id { get; set; }
public int PrimaryId { get; set; }
public int RelatedId { get; set; }

public Subject Primary { get; set; }
public Subject Related { get; set; }
}
```

Предмет_элективная_группа

```
public class SubjectElectiveGroup
{
public int SubjectId { get; set; }
public int ElectiveGroupId { get; set; }

public Subject Subject { get; set; }
public ElectiveGroup ElectiveGroup { get; set; }
}
```

Предмет

```
public class Subject
{
```



```

public int Id { get; set; }
public string Name { get; set; }
public int Credits { get; set; }
public string Shifr { get; set; }
public int Lec { get; set; }
public int Lab { get; set; }
public int Pr { get; set; }
public string Date { get; set; }
public string UpdateDate { get; set; }
public int Deleted { get; set; } = 0;

```

```

public string UserId { get; set; }
public int TypeId { get; set; }
public SubjectType Type { get; set; }

```

```

public virtual List<SubjectPrerequisiteSubject> RelatedItems { get; set; } = new
List<SubjectPrerequisiteSubject>();
public virtual List<SubjectPrerequisiteSubject> RelatedTo { get; set; } = new
List<SubjectPrerequisiteSubject>();

```

```

public virtual List<SubjectElectiveGroup> SubjectElectiveGroups { get; set; } =
new List<SubjectElectiveGroup>();
public virtual List<SemesterSubject> SemesterSubjects { get; set; }
}

```

Предмет_семестр.

```

public class SemesterSubject
{
public int SemesterId { get; set; }
public int SubjectId { get; set; }

public Semester Semester { get; set; }
public Subject Subject { get; set; }
}

```

Семестр_элективная_группа.

```

public class SemesterElectiveGroup
{
public int SemesterId { get; set; }
public int ElectiveGroupId { get; set; }

public Semester Semester { get; set; }
}

```

```
public ElectiveGroup ElectiveGroup { get; set; }  
}
```

Элективная группа

```
public class ElectiveGroup  
{  
    public int Id { get; set; }  
    public string Name { get; set; }  
    public int Credits { get; set; }  
    public string Shifr { get; set; }  
    public int Pr { get; set; }  
    public string Date { get; set; }  
    public string UpdateDate { get; set; }  
    public int Deleted { get; set; } = 0;  
  
    public int TypeId { get; set; }  
    public SubjectType Type { get; set; }  
  
    public string UserId { get; set; }  
  
    public virtual List<SemesterElectiveGroup> SemesterElectiveGroups { get; set; }  
  
    public virtual List<SubjectElectiveGroup> SubjectElectiveGroups { get; set; } =  
    new List<SubjectElectiveGroup>();  
}
```

Специальность

```
public class Specialty  
{  
    public int Id { get; set; }  
    public string Name { get; set; }  
    public string Shifr { get; set; }  
    public string Date { get; set; }  
    public string UpdateDate { get; set; }  
    public int Deleted { get; set; } = 0;  
  
    public string UserId { get; set; }  
  
    public virtual List<Ryp> Ryps { get; set; }  
}
```

Семестр.

```
public class Semester
{
public int Id { get; set; }

public int RypId { get; set; }
public Ryp Ryp { get; set; }

public virtual List<SemesterSubject> SemesterSubjects { get; set; }
public virtual List<SemesterElectiveGroup> SemesterElectiveGroups { get; set; }
}
```

РҮП

```
public class Ryp
{
public int Id { get; set; }
public string Name { get; set; }
public string Year { get; set; }
public string Date { get; set; }
public string UpdateDate { get; set; }
public int OperatorCheck { get; set; } = 0;
public int FullCheck { get; set; } = 0;
public int Deleted { get; set; } = 0;
public int Prototype { get; set; } = 0;

public string UserId { get; set; }

public int SpecialtyId { get; set; }
public Specialty Specialty { get; set; }

public virtual List<Semester> Semesters { get; set; }

}
```