

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование кафедры)

09.03.03 Прикладная информатика

(код и наименование направления подготовки, специальности)

Прикладная информатика в социальной сфере

(направленность (профиль)/специализация)

БАКАЛАВРСКАЯ РАБОТА

на тему «Разработка системы управления рекламными рассылками курсов
платного образования»

Студент

А.О. Юдина

(И.О. Фамилия)

(личная подпись)

Руководитель

С.В. Мкртычев

(И.О. Фамилия)

(личная подпись)

Консультанты

М.М. Бажутина

(И.О. Фамилия)

(личная подпись)

Допустить к защите

Заведующий кафедрой к.т.н., доцент, А.В. Очеповский

(ученая степень, звание, И.О. Фамилия)

(личная подпись)

« _____ » _____ 20__ г.

Тольятти 2018

АННОТАЦИЯ

Тема ВКР: «Разработка систем управления рекламными рассылками курсов платного образования».

Целью данной бакалаврской работы является разработка информационной системы управления рекламными рассылками курсов платного образования.

Работа состоит из введения, трех глав, заключения и списка используемой литературы.

В первой главе проводится анализ деятельности Института дополнительного образования, изучению бизнес процессов рассылки платных сообщений, рассмотрены модели «Как есть», «Как должно быть», назначения и требования к информационной системе.

Во второй главе дается краткая характеристика и проводится сравнительный анализ современных методологий и технологий концептуального моделирования информационных систем, разрабатывается логическая структура информационной системы и проектируется база данных.

В третьей главе описан выбор технологии для разработки программного обеспечения информационной системы, разработана физическая модель базы данных, рассмотрены программные модули и интерфейс программы.

В заключении формулируются выводы по проделанной работе, определяется, какие задачи были решены, определяются пути их внедрения и направления дальнейшего совершенствования.

Объем ВКР -48 страниц. Работа содержит 22 рисунка, 1 таблицу, 26 источников литературы.

ABSTRACT

The title of the bachelor's thesis is "Development of Management Systems for Advertising Mailings of Paid Education Courses".

The aim of the work is to create an information system for advertising mailings of paid education courses.

In the first chapter the analysis of the activities of the Institute of Continuing Education, the study of business processes for the distribution of paid messages, the models "AS IS", "TO BE", the purpose and requirements for the information system.

The second chapter provides a brief description and comparative analysis of modern methodologies and technologies for the conceptual modeling of information systems, develops the logical structure of the information system and designs a database.

The third chapter describes the choice of technology for the development of information system software, a physical model of the database is developed, program modules and the program interface are examined.

The conclusion summarizes the results of the work, what tasks have been accomplished, and projects ways to implement them and to guide their further improvement.

The bachelor's thesis contains 22 figures, 1 table, a list of references (26 items). The whole volume of the work is 48 pages.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
Глава 1 Функциональное моделирование предметной области	7
1.1 Анализ деятельности Института дополнительного образования	7
1.2 Обоснование выбора методологии и технологии концептуального моделирования информационной системы	8
1.4 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ»	14
1.5 Анализ известных программных решений	16
1.6 Назначение разработки и требования создания информационной системы.....	17
Выводы по главе 1	20
Глава 2 Логическое проектирование информационной системы.....	22
2.1 Обоснование выбора методологии и технологии логического моделирования информационной системы	22
2.2 Разработка логической модели информационной системы.....	23
2.3 Концепция информационной системы	24
Выводы по главе 2	26
Глава 3 Разработка и внедрение автоматизированной информационной системы управления рекламными рассылками курсов платного образования	27
3.1 Выбор технологии разработки программного обеспечения информационной системы.....	27
3.2 Структурная схема базы данных	29
3.3 Программные модули информационной системы	30
3.4 Интерфейс программы.....	34
Выводы по главе 3	38
ЗАКЛЮЧЕНИЕ	39
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	40
ПРИЛОЖЕНИЕ А Код программы	43

ВВЕДЕНИЕ

В современном мире Интернет имеет важное значение в развитии компании. В частности, организации разрабатывают маркетинговые Интернет-стратегии по привлечению клиентов через социальные сети, через сайт предприятия, таргетинговую рекламу и e-mail рассылки.

Институт дополнительного образования также рассматривает Интернет, как часть стратегии развития компании для увеличения клиентской базы, в частности для распространения почтовых рассылок.

В качестве получателей будут рассматриваться различные предприятия и организации, имеющие в своем штате сотрудников с необходимыми направлениями деятельности.

Актуальность: Формирование рекламных рассылок с учетом специфики определенного предприятия повысит доверие работодателей к Институту дополнительного образования и позволит осуществить индивидуальный подход к каждому из клиентов Института.

Целью автоматизации процесса рекламных рассылок является повышение рейтинга Института дополнительного образования на основе индивидуального подхода к потенциальным и имеющимся клиентам.

Объект исследования – бизнес-процесс рекламной рассылки курсов платного образования.

Предмет исследования – информационная система управления рекламными рассылками курсов платного образования.

Целью работы является разработка система управления рекламными рассылками курсов платного образования.

Для достижения цели сформулированы следующие задачи:

- Проанализировать работу Института дополнительного образования;
- Спроектировать информационную систему для управления рекламными рассылками курсов платного образования;
- Разработать информационную систему для управления рекламными рассылками курсов платного образования.

Методы исследования – методологии структурного и объектно-ориентированного анализа и проектирования ИС.

Соответствие содержания бакалаврской работы профессиональным компетенциям по видам профессиональной деятельности выпускника:

- научно-исследовательская деятельность:
 - способность применять системный подход и математические методы в формализации решения прикладных задач (ПК-23);
 - способность готовить обзоры научной литературы и электронных информационно-образовательных ресурсов для профессиональной деятельности (ПК-24).

Работа состоит из введения, трех глав, заключения, списка используемой литературы и приложение с кодом.

В первой главе проводится анализы деятельности Института дополнительного образования, изучению бизнес процессов рассылки платных сообщений, рассмотрены модели «Как есть», «Как должно быть», назначения и требования к информационной системе.

Во второй главе дается краткая характеристика и проводится сравнительный анализ современных методологий и технологий концептуального моделирования ИС, разрабатывается логическая структура информационной системы и проектируется база данных.

В третьей главе описан выбор технологии для разработки программного обеспечения информационной системы, разработана физическая модель базы данных, рассмотрены программные модули и интерфейс программы.

В заключении формулируются выводы по проделанной работе, определяется, какие задачи были решены, определяются пути их внедрения и направления дальнейшего совершенствования.

Глава 1 Функциональное моделирование предметной области

1.1 Анализ деятельности Института дополнительного образования

Рассмотрим структуру ИДО. ИДО, как правило, имеет в своей структуре несколько отделов: администрация, отдел по работе с клиентами, отдел по учебной работе, отдел маркетинга. На рисунке 1.1 приведена структура Института.

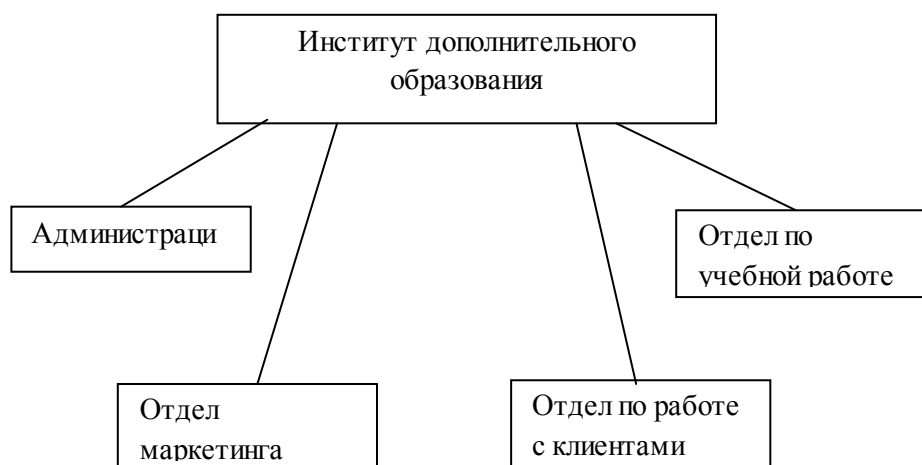


Рисунок 1.1 – Организационная структура ИДО

Важное значение в развитии компании является маркетинг, эффективным инструментом которого являются почтовые рассылки.

Процесс рассылки курсов платного образования Института дополнительного образования (ИДО) можно описать следующим образом.

Отдел по работе с клиентами получает информацию о потенциальных клиентах из различных источников – Центр занятости населения, непосредственно от клиентов, через интернет. Сотрудники отдела ведут картотеку, в которой содержится информация о предприятии, направление его деятельности, кадровый состав предприятия. Данные картотеки периодически актуализируются.

Отдел по учебной работе формирует перечень курсов платного образования, составляет учебные планы курсов, определяет количество учебных часов и стоимость курсов. Сотрудники отдела ведут картотеку курсов.

Отдел маркетинга осуществляет рекламную рассылку перечня курсов платного образования потенциальным клиентам с указанием продолжительности и стоимости курсов, а также документов, подтверждающих прохождение курса.

Подбор перечня курсов для рассылки в различные организации осуществляется вручную.

Основной задачей автоматизированной системы управления рекламными рассылками курсов платного образования будет являться автоматизация подбора перечня курсов под нужды потенциального клиента путем сопоставления направления его деятельности и кадрового состава предприятия с имеющимися направлениями дополнительного образования.

Целью автоматизации является применение индивидуального подхода к каждому из клиентов Института повышения квалификации.

1.2 Обоснование выбора методологии и технологии концептуального моделирования информационной системы

Рассмотрим различные методологии и технологии концептуального моделирования ИС.

В наше время принято выделять:

- Объектно-ориентированный;
- Структурный.

Рассмотрим каждый из этих подходов.

Основа структурного подхода заключается в декомпозиции (разбиении): система разбивается на подсистемы, которые также делятся. Декомпозиция продолжается до конкретных процедур, при этом описываемая схема сохраняет целостное представление.

Часто встречающиеся модели:

- DFD (Data Flow Diagrams) диаграммы потоков данных;
- SADT (Structured Analysis and Design Technique) модели и соответствующие функциональные диаграммы;

- ERD (Entity-Relationship Diagrams) диаграммы "сущность-связь".

В совокупности приведенные модели представляют полное определение ИС независимо от того, представляет ли собой она имеющейся или вновь разрабатываемой. В каждом конкретном случае структура диаграмм зависит от необходимой полноты описания системы.

Следующей популярной технологией проектирования ИС является объектно-ориентированный анализ и проектирование (ООАП), которая строится на объектно-ориентированной методологии представления предметной области в виде объектов, являющихся экземплярами соответствующих классов.

В ООАП рассматривались графических нотации:

- Диаграммы "сущность-связь" (ERD);
- Диаграммы функционального моделирования (SADT).

Задачей онтологического моделирования является создание формализованных электронных моделей знаний. Цели применения этих моделей лежат в сфере бизнеса, и могут включать:

- Выполнение имитационного моделирования процессов с целью их оптимизации;
- Быстрое получение логических выводов на основании большого количества информации, с целью поддержки принятия решений;
- Обеспечение доступности для восприятия пользователей больших объемов сложно структурированной информации;
- Решение ряда технических задач, прежде всего в области интеграции информационных систем.

Необходимо заметить, что принятие решений – это волевой акт субъекта, напрямую зависящий от его интересов. Именно поэтому, говоря об информационной системе, рассматривается только поддержка принятия решений, которая заключается в наилучшем обеспечении человека информацией, требуемой для их осознанного принятия, а также предсказании

последствий тех или иных вариантов решения для того, чтобы человек мог оценить их приемлемость с учетом ценностных установок.

По результатам проведения сравнительного анализа в качестве метода исследования была выбрана методология структурного анализа и проектирования ИС.

В качестве графических нотаций для проектирования информационной системы для бакалаврской работы были выбраны нотации:

- IDEF0 – для описания бизнес процессов производства и описание информации об использовании ресурсов;
- DFD - диаграмма потоков данных, описывающая внешние по отношению к системе источники.

Следующее, что нужно выбрать – это CASE-средство для проектирования информационной системы, с помощью которого можно охарактеризовать предметную область в качестве модели.

Основой CASE-технологии служит база данных, хранящая не только информационные объекты, но и правила работы с ними.

CASE-технология представляет собой набор инструментальных средств, которые позволяют моделировать предметную область, анализировать модель на всех стадиях разработки и сопровождения, разрабатывать приложения в соответствии с информационными потребностями пользователей.

То есть при выборе CASE-средства нам надо учитывать возможность проектирования базы данных, выбранные выше нотации. В качестве такого CASE-средства, была выбрана программы Erwin: Erwin Data Modeler для проектирования базы данных и Erwin Process Modeler для графических нотаций.

1.3 Разработка и анализ модели бизнес-процесса «КАК ЕСТЬ»

Для разработки и анализа модели рассматриваемого бизнес-процесса будет использована нотация IDEF0, которая дает исчерпывающее представление о

составляющих бизнес-процесса в графической форме.

На первом этапе разрабатывается контекстная диаграмма бизнес-процесса, которая показывает входящие и исходящие потоки информации, участников процесса, управляющие воздействия.

Контекстная диаграмма бизнес-процесса по рекламной рассылке курсов «как есть» платного образования представлена на рисунке 1.2.

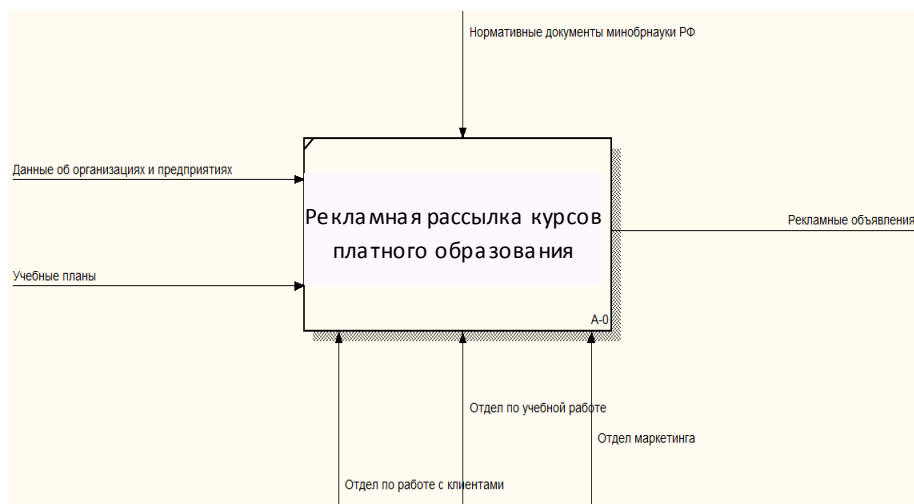


Рисунок 1.2 – Контекстная диаграмма бизнес-процесса «Рекламная рассылка курсов платного образования» «как есть»

Для наиболее четкого понимания процесса формирования рекламных рассылок необходимо провести декомпозицию процесса A0. Декомпозиция процесса A0 представлена на рисунке 1.3.

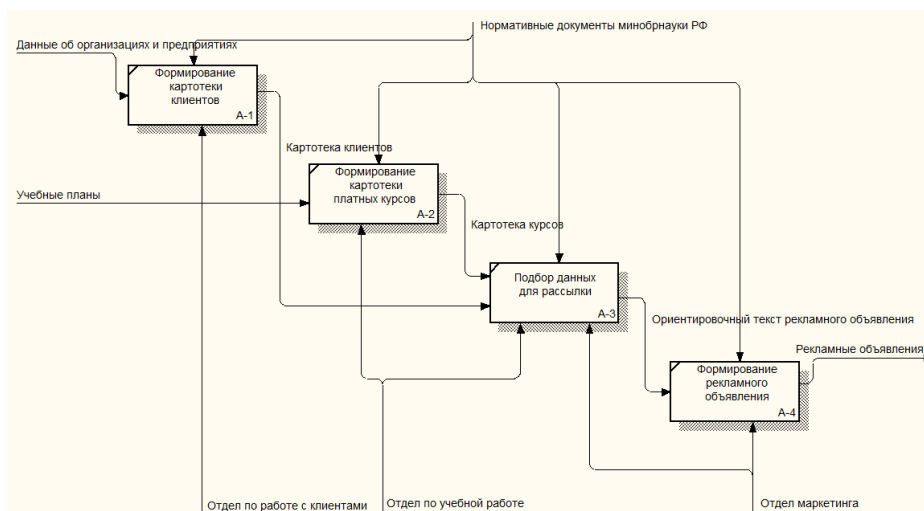


Рисунок 0.3 - Декомпозиция процесса A0

Декомпозиция процесса А0 показывает, что наиболее сложным процессом в цепочке процессов по формированию рекламных объявлений является процесс подбора данных для рассылки (процесс А3).

Данный процесс имеет наибольшее количество участников, а также аккумулирует в себе работу со всей информацией, участвующей в данном бизнес-процессе.

Для наиболее глубокого понимания процесса подбора данных для рассылки необходимо провести декомпозицию процесса А3.

Декомпозиция процесса А3 представлена на рисунке 1.4.

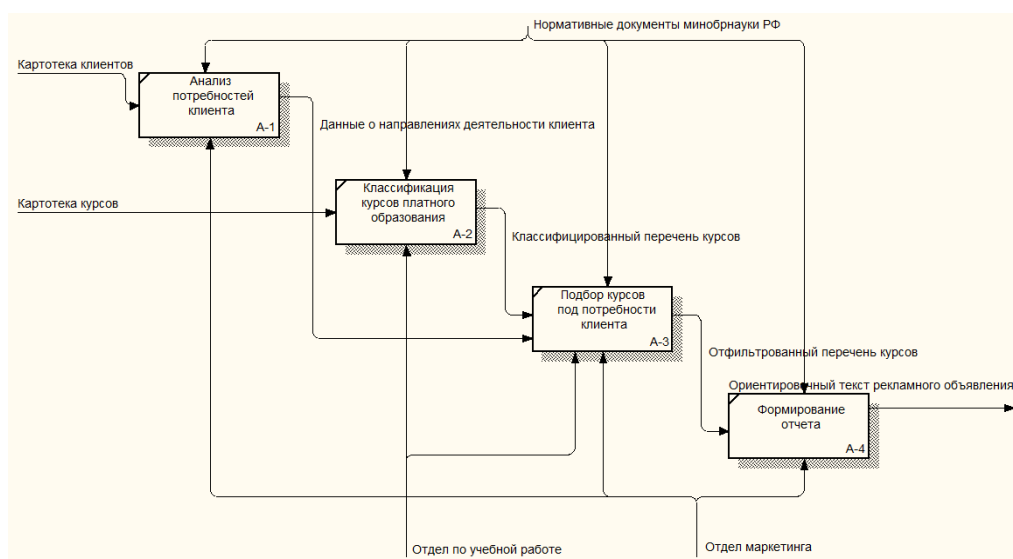


Рисунок 0.4 - Декомпозиция процесса А3

Декомпозиция процесса А3 показывает, что основной задачей для формирования рекламного объявления, которое заинтересует клиента, является подбор курсов под направление деятельности организации или же подходящие для данной организации по ее кадровому составу.

Автоматизация подбора необходимых курсов существенно упростит и увеличит эффективность процесса по рассылке рекламных объявлений.

Рассмотренный бизнес-процесс имеет ряд недостатков. Основным недостатком можно считать ручную классификацию предприятий и курсов платного образования по определенным признакам.

Для автоматизации подбора курсов платного образования под нужды предприятия необходимо, чтобы и потребности клиента, и курсы платного образования имели идентичные классификаторы, определяющие их отношение к тому или иному виду деятельности.

Например, предприятию по продаже стройматериалов не будут интересны курсы повышения квалификации медицинских работников, однако, если это предприятие является достаточно крупным и имеет собственный штат бухгалтеров, то ему будут интересны курсы повышения квалификации бухгалтера.

Следовательно, при присвоении данной организации признаков, по которым будут подбираться курсы платного образования, следует учесть, что данной организации могут быть интересны курсы по складскому учету материальных ценностей, по бухгалтерии, по повышению эффективности продаж и т.п.

По таким же признакам должны быть классифицированы и сами учебные курсы. Однако здесь нужно заметить, что курсы являются более однонаправленными и не могут обучать сразу и бухгалтерии, и маркетингу.

Следовательно, при классификации предприятий предприятие можно отнести к нескольким классам, а при классификации курсов – один курс можно отнести только к одному классу.

Исходя из вышесказанного, процесс автоматизации рекламных рассылок можно свести к использованию одной таблицы классов для классификации и предприятий, и курсов платного образования.

Подбор необходимых курсов для предприятия будет выполняться автоматически на основании запроса к базе данных. Это позволит отказаться от ручной аналитики данных клиентов и курсов, а администрирование базы данных будет сведено к периодической актуализации данных клиентов и курсов.

1.4 Разработка модели бизнес-процесса «КАК ДОЛЖНО БЫТЬ»

Для разработки модели автоматизированной системы по рассылке информации о курсах платного образования используется нотация DFD.

На первом этапе разрабатывается контекстная диаграмма, которая отражает входные и выходные данные автоматизированной системы.

Контекстная диаграмма представлена на рисунке 1.5.

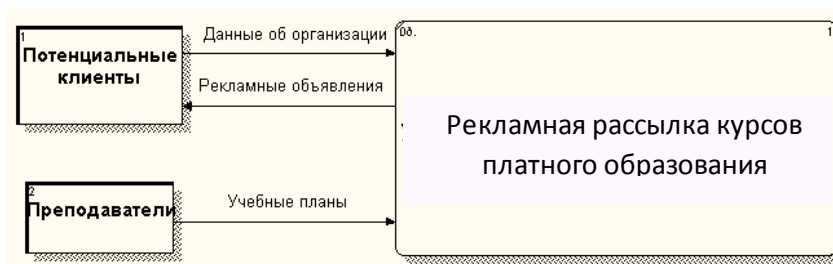


Рисунок 0.5 – Диаграмма потоков данных процесса «Рекламная рассылка курсов платного образования»

Для получения декомпозиции процесса А0 необходимо определить потоки данных, необходимые процесса управления рекламными рассылками.

В данном случае потоками данных будут являться:

- Данные об организациях;
- Классификатор направлений деятельности;
- Данные о курсах платного образования.

В качестве процессов будут рассмотрены процессы отдела по работе с клиентами и бизнес процессы сотрудника отдела по учебной работе.

При составлении модели «КАК ЕСТЬ» в процессе также участвовали сотрудники отдела маркетинга, а выполняемые ими задачи являлись ключевыми в рассматриваемом бизнес-процессе.

Однако благодаря автоматизации данного бизнес-процесса сотрудников отдела маркетинга можно исключить из данного процесса, а функции по автоматическому формированию рекламных объявлений передать отделу по работе с клиентами.

Диаграмма потоков данных на уровне подразделений изображена на рисунке 1.6.

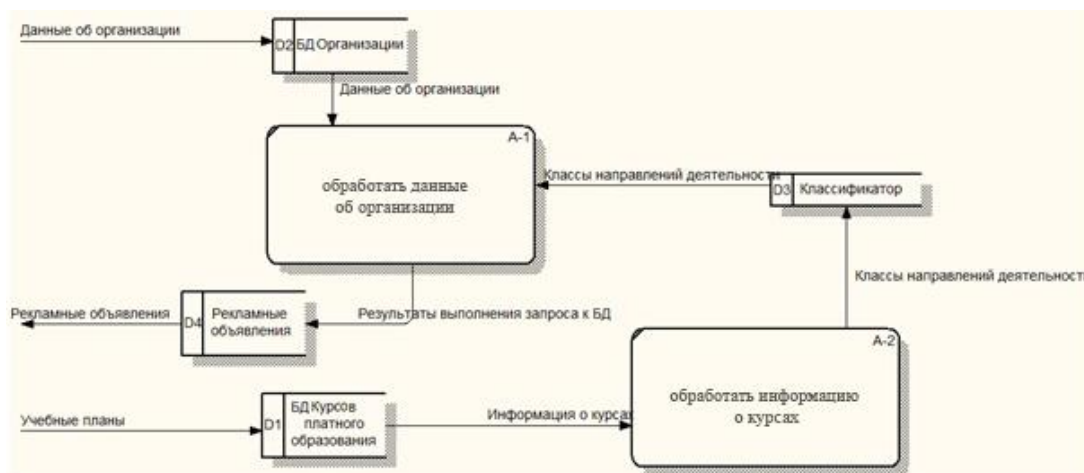


Рисунок 0.6 – Диаграмма процесса потоков данных на уровне подразделений

Рассматриваемый процесс не является сложным, поэтому необходимость в дальнейшей декомпозиции отсутствует.

Сформируем требования к информационной системе в системе FURPS+.

FURPS+ расшифровывается как:

- Функциональность;
- Удобство использования;
- Надежность;
- Производительность;
- Поддерживаемость.

Перечислим требования к информационной системе:

- Система должна иметь возможность отправлять письма.
- Система должна иметь возможность формирования шаблона письма.
- Ведение списка организаций-клиентов компании, которым будут отправлять рекламные рассылки.
- Возможность добавления новой организации получателя.

- Возможность отправить письма с рассылкой с выбранным курсом выбранным организациям (фильтр по курсам и клиентам).
- Возможность ведения списка курсов.
- Возможность ведения подписей лиц.
- Система должна предоставлять доступ на основании введенных логина и пароля пользователя (авторизация).
- Система должна быть проста, надежна.
- У системы должен быть интуитивно-понятный интерфейс.
- У системы должен быть простой и быстрый поиск информации.
- Небольшой бюджет на рассылки.

В разделе ниже рассмотрим эти требования на уже известных программных решениях.

1.5 Анализ известных программных решений

В качестве известных ИТ-решений для рассылки рекламных объявлений будут рассмотрены следующие онлайн сервисы:

- UniSender.com - сервис email рассылок, в котором за 15 минут можно с нуля создать рассылку, сегментация подписчиков.
- Mailerlite.com - англоязычный сервис email рассылок, идеально для малого бизнеса с небольшой базой подписчиков.
- www.getresponse.ru – русскоязычный сервис email рассылок, с возможностью настройки маркетинга, лендингов.

Сравнительный анализ данных сервисов приведен в таблице 1.1.

Таблица 1.1 – Сравнение онлайн сервисов для рассылки объявлений

	UniSender.com	Mailerlite.com	getresponse.ru
Отправка писем	+	+	+
Шаблоны писем объявлений	+	+	+
Ведение организаций клиентов	+	+	+
Добавление новой организации-клиента	+	+	+
Фильтр по курсам и клиентам	-	-	-
Авторизация	+	+	+
Ведение списков курсов	+	+	+

Ведение подписей адресата	+	+	+
Интуитивно-понятный интерфейс	+	+	+
Удобный поиск получателей	+	+	+
Бюджетный вариант	-	-	+
Итого	9	9	10

Анализ готовых ИТ-решений для рассылки объявлений показывает, что имеющиеся сервисы не адаптированы для персональной работы с клиентами.

Подавляющая часть подобных сервисов осуществляет рассылку только на тематические сайты и крупные сайты с объявлениями.

Некоторые сервисы имеют возможность работы с клиентской базой заказчика, однако рассылка осуществляется по принципу «всем сразу», т.е. отсутствует возможность индивидуального подхода к каждому клиенту.

Разрабатываемая ИС должна решить задачу персонализации рекламных рассылок, что позволит клиентам относиться к ним не как к спаму, а как к полезной и нужной информации.

1.6 Назначение разработки и требования создания информационной системы

Для оптимальной работы система должна отвечать следующим требованиям:

1) Назначение и цели создания системы.

ИС предназначена для управления рекламными рассылками курсов платного образования.

Целями создания ИС являются:

- Повышение надежности передачи и хранения данных;
- Автоматизация формирования рекламных рассылок с учетом направлений деятельности клиента;
- Автоматизация индивидуального подхода к клиенту.

2) Характеристика объектов автоматизации.

Объектами автоматизации являются подразделения: отдел по работе с клиентами, отдел по учебной работе.

3) Требования к системе в целом.

Автоматизированная система управления рекламными рассылками курсов платного образования должна содержать следующие разделы : раздел менеджера по работе с клиентами, раздел сотрудника отдела по учебной работе.

Раздел ИС менеджера по работе с клиентами предназначено для ведения базы данных клиентов и формирования рекламных объявлений для последующей рассылки клиентам.

Раздел ИС сотрудника отдела по учебной работе предназначено для ведения базы данных курсов платного образования и ведения классификатора направлений деятельности клиентов и курсов дополнительного образования.

В будущем надо учесть возможность работы других отделов компании.

Режим работы системы - 8 час в день.

Персонал должен иметь подготовку на уровне опытного пользователя персонального компьютера, профессиональную подготовку и навыки работы с информационной системой.

Контроль за подготовку персонала необходимо исполнить на этапе внедрения.

ИС обязана принимать во внимание единые требования к информационной безопасности и учитывать различные средства защиты от несанкционированного доступа:

- Распределение ответственности за информационную безопасность;
- Организация парольного доступа.

Компоненты подсистемы защиты от несанкционированного доступа должны обеспечивать:

- Идентификацию пользователя;
- Проверку полномочий пользователя при работе с системой;

- Разграничение доступа пользователей на уровне решаемых задач и выполняемых функций.

В случае возникновения аварии или сбоя в процессе выполнения пользовательских задач должно быть обеспечено восстановление базы данных до состояния на момент последней завершенной системой транзакции.

В случае повреждения журналов транзакций СУБД должно обеспечиваться восстановление состояния системы на момент создания последней резервной копии данных, но не более чем за сутки до момента сбоя.

4) Требования к функциям (задачам), выполняемым системой.

Раздел ИС менеджера по работе с клиентами должно обеспечивать:

- Ввод и редактирование данных справочника «Организации»;
- Формирование рекламных объявлений для рассылки.

ИС сотрудника отдела по учебной работе должно обеспечивать:

- Ввод и редактирование данных справочников «Учебные курсы».

5) Требования к математическому обеспечению.

Раздел менеджера по работе с клиентами должно обеспечивать формирование рекламных объявлений по направлению деятельности клиента путем выполнения запросов к базе данных.

6) Требования к информационному обеспечению.

Ввод данных осуществляется при помощи клавиатуры. Вывод данных производится при помощи принтера.

Должна быть обеспечена возможность сохранения данных в случае авариях или внезапном отключении электропитания.

Для контроля данных можно применить ограничения диапазонов, соотношения между значениями и другие свойства данных. Данные за прошлый год необходимо хранить в архиве. Утерянные данные восстанавливаются из архива АРМ или из копии БД.

7) Требования к лингвистическому обеспечению.

Для организации взаимодействия с пользователем все прикладное ПО системы должно быть на русском языке, для формирования запросов использовать язык SQL.

Для выбора пользовательских функций использовать меню, горячие клавиши и манипулятор типа «мышь».

8) Требования к техническому обеспечению.

Должно быть оборудовано рабочее место для реализации возможностей ИС с техническими характеристиками не ниже следующих:

- Процессор с частотой 2.2 ГГц;
- Монитор 18,5 дюймов;
- Оперативная память 2 ГБ;
- Жесткий диск 500 GB.

9) Требования к организационному обеспечению.

Организационное обеспечение системы должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей при осуществлении автоматизированных и связанных с ними неавтоматизированных функций системы.

К работе с системой должны допускаться сотрудники, прошедшие обучение работе с информационной системой.

10) Требования к методическому обеспечению.

Необходимо разработать «Руководство пользователя» и «Систему помощи» для ИС.

Выводы по главе 1

В первой главе был проведен анализ деятельности Института дополнительного образования, описаны цели организации и задач управления исследуемым бизнес процессом, дана краткая характеристика и проведен сравнительный анализ современных методологий и технологий концептуального моделирования ИС. Также в соответствии с концепцией

реинжиниринга на основе выбранной методологии строится модель бизнес-процесса «КАК ЕСТЬ» (AS-IS) и проводится ее анализ, перечисляются недостатки существующего бизнес-процесса. Предлагаются решения для его усовершенствования путем внедрения в него в качестве одного из механизмов управления - ИС. На основе выбранной методологии строится развернутая модель бизнес-процесса «КАК ДОЛЖНО БЫТЬ» (TO-BE). Выполняется декомпозиция модернизированных процессов модели. Формулируются требования к новой информационной системе. На основании построенных моделей проводится разработка требований к создаваемой информационной системе.

Глава 2 Логическое проектирование информационной системы

2.1 Обоснование выбора методологии и технологии логического моделирования информационной системы

Рассмотрим процесс моделирования информационной системы, который начинается с определения требований пользователей базы данных.

Концептуальная модель отображает предметную область в виде взаимосвязанных объектов без указания методов их физического хранения. Концептуальная модель представляет интегрированные концептуальные требования всех пользователей к БД данной предметной области.

При выборе важно, чтобы усилия были направлены на структуризацию данных, принадлежащих пользователям базы данных, и выявление между ними взаимосвязей. Не исключено, что показанные взаимосвязи между объектами получатся нереализуемыми или труднореализуемыми средствами выбранной СУБД. В таком случае потребуются подвергнуть изменениям концептуальную модель. Версия, реализуемая конкретной СУБД, является логической моделью.

Логическая модель рассматривает логические связи между атрибутами объектов независимо от их содержания и среды хранения и может быть любой моделью СУБД. Следовательно, логическая модель показывает логические связи между информационными данными.

Различным пользователям в ИС соответствуют различные подмножества ее логической модели, называемые внешними моделями пользователей. Таким образом, внешняя модель пользователя представляет собой отображение концептуальных требований этого пользователя в логической модели и соответствует получаемым представлениям пользователя о предметной области на основе логической модели.

Логическая модель отображается в физическую память, которая может быть исполнена на магнитных, электронных, биологических, оптических или других принципах.

Внутренняя модель предметной области рассматривает методы доступа, размещение данных и технику индексирования в данной логической модели и называется физической моделью.

Данные всех пользователей в БД должны быть независимы друг от друга, т. е. не должны оказывать влияния на существующие внешние модели. Это положение отражает первый уровень независимости данных. Внешние модели пользователей никак не связаны с типом физической памяти, в которой будут храниться данные, и с физическими методами доступа к этим данным. Это второй уровень независимости данных.

Таким образом, для разработки информационной системы на этапе концептуального моделирования является логическая модель ИС.

2.2 Разработка логической модели информационной системы

Для разработки логической модели ИС необходимо определить объекты, которые необходимы для корректной работы проектируемой системы.

Данными объектами будут являться:

- Клиенты;
- Шаблоны писем;
- Рассылки;
- Пользователи;
- Роли;
- Курсы.

Логическая модель проектируемой информационной системы представлена на рисунке 2.1.

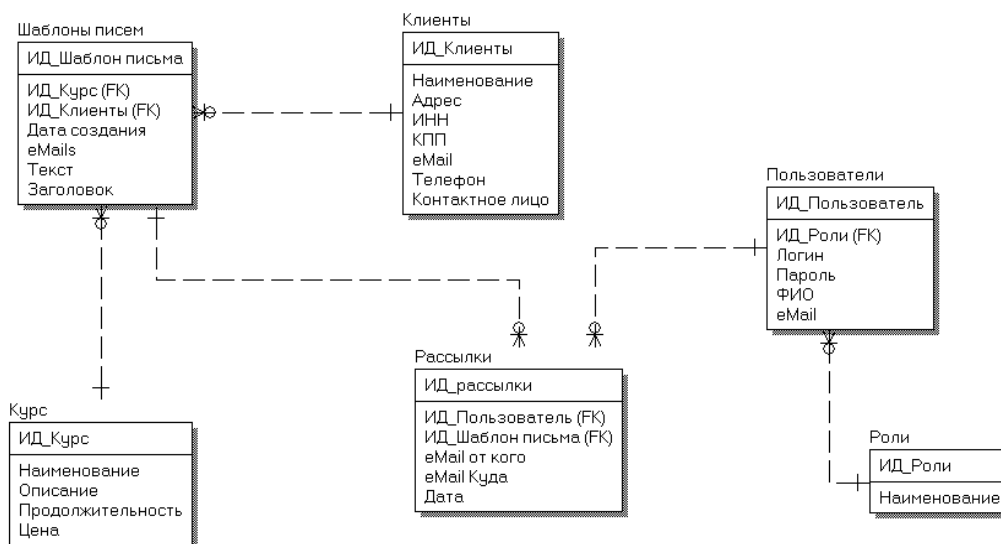


Рисунок 0.1 – Логическая модель данных системы

Разработанная логическая модель данных позволит эффективно хранить и обрабатывать данные для рассылки.

2.3 Концепция информационной системы

Информационная система будет содержать функциональные возможности:

- Авторизация пользователей – авторизация будет осуществляться на основании логина и пароля пользователя, в соответствии его роли ему будут предоставлены соответствующие права.
- Работа с организациями-клиентами: ввод новой организации, редактирование уже имеющейся записи, удаление организации.
- Настройка курсов: введение списка курсов, их описания и цен.
- Настройка шаблонов писем, которые впоследствии смогут сформировать структуру письма.
- Настройка подписей адресатов для последующей вставки в письмо.
- Основное окно для создания рассылки писем с возможностью выбора списка получателей, шаблона и курса, на основании которых будет формироваться структура письма.

Для лучшего понимания информационной системы разработаем рабочий макет интерфейса.

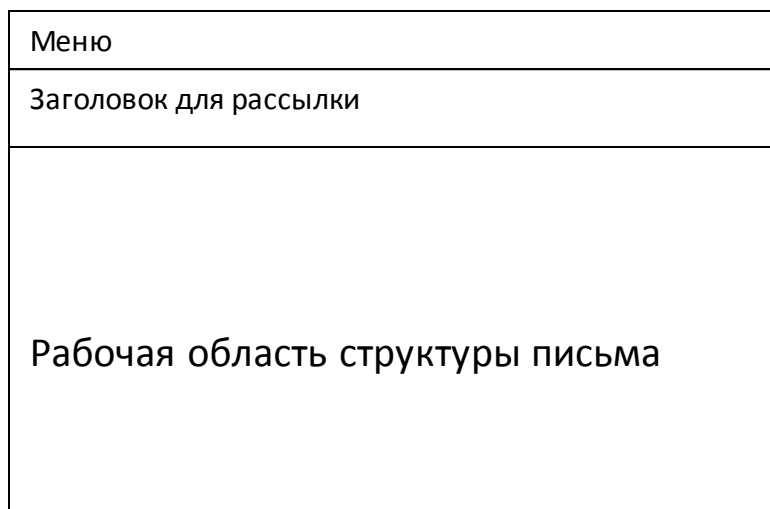


Рисунок 2.2 – Рабочий макет интерфейса системы

Меню будет содержать пункты:

- Данные системы:
 - Учебные курсы – список курсов ИДО;
 - Организации - список организаций клиентов;
 - Банки – список банков клиентов;
 - Подписи лиц – список подписей адресатов для автоматической вставки при формировании рассылки.
- Почта:
 - Шаблоны писем – для автоматического формирования писем нужной структуры для рассылки.
- Настройка – настройки для отправки почты.

Заголовок для рассылки будет содержать поля отбора для формирования рассылки:

- Выбор шаблона, с которого будет формироваться структура письма;
- Получатели – список организаций-клиентов.

Рабочая область нужна для формирования структуры письма.

Для более полного понимания ИС изобразим ее с помощью UML. На рисунке 2.3 изображена UML диаграмма классов системы.

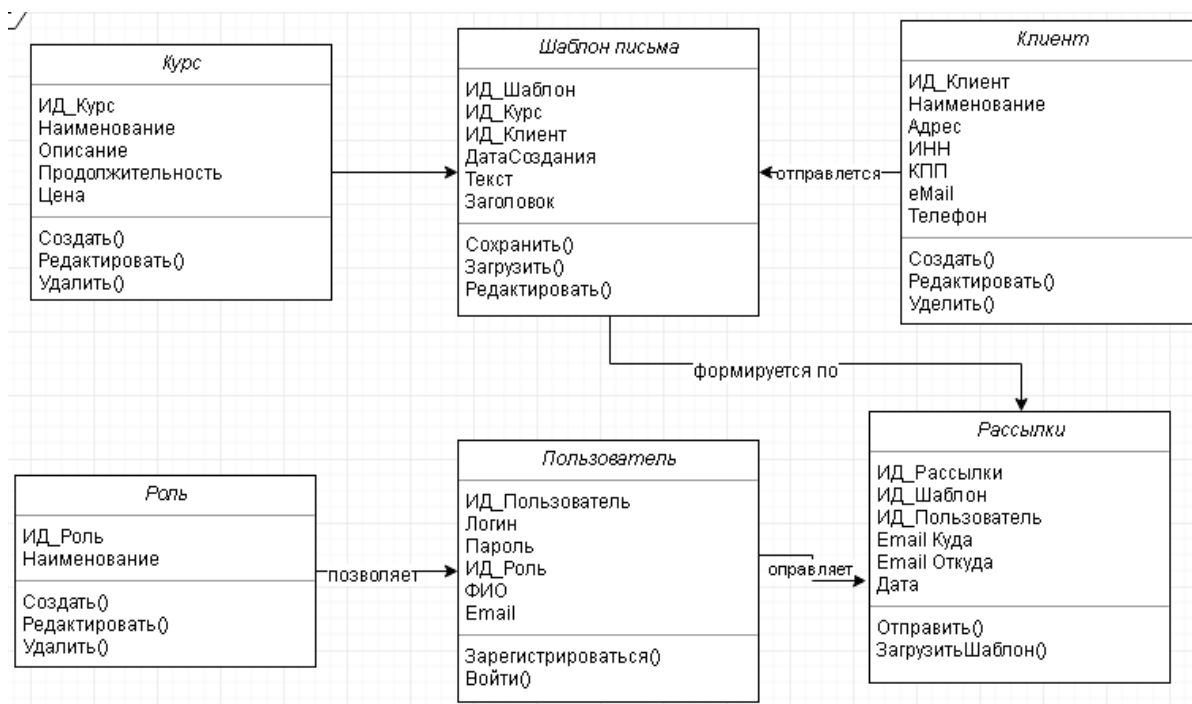


Рисунок 2.3 – UML Диаграмма классов системы

Выводы по главе 2

Во второй главе дается краткая характеристика и проводится сравнительный анализ современных методологий и технологий логического моделирования ИС. Описываются методологии и средства разработки модели данных информационных систем.

Глава 3 Разработка и внедрение автоматизированной информационной системы управления рекламными рассылками курсов платного образования

3.1 Выбор технологии разработки программного обеспечения информационной системы

Для начала определим язык программирования для разработки информационной системы управления персоналом. Здесь важно выделить критерии оценки:

- Кроссплатформенный язык, который позволяет работать разработанной системе на различных операционных системах;
- Легкость и быстрота освоения;
- Бесплатный или условно бесплатный;
- Доступность литературы;
- Объектно-ориентированный;
- Надежность.

В результате вышеописанного в качестве языка программирования был выбран Java.

Теперь выберем базу данных, для начала определим критерии отбора:

- Кроссплатформенная;
- Возможность взаимодействия с java, соответствие стандарту jdbc (англ. Java database connectivity — соединение с базами данных на java);
- Надежная, существует возможность резервного копирования;
- Возможность нескольких подключений;
- Бесплатная, условно-бесплатная.

В качестве такой СУБД можно рассмотреть: MySQL и Oracle, обе подходят по вышеописанным критериям, с одним лишь отличием Oracle – платная СУБД для крупных компаний. Поэтому, идеально подходит MySQL.

MySQL – бесплатная реляционная система управления базами данных от компании Oracle.

Далее выберем среду обработки для программирования, которая имеет возможность взаимодействовать с Java и MySQL и которая упростит процесс обработки. В качестве такой среды разработки была выбрана IntelliJ Idea - интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains.

В качестве технологий в разработке будут использованы следующие:

Maven - фреймворк для автоматизации сборки проектов на основе описания их структуры в файлах на языке POM, являющемся подмножеством XML. Взаимодействует со средой IntelliJ idea, которая была выбрана в качестве среды обработки.

Spring boot - инструмент от компании Pivotal, исключая необходимость прямого подключения встроенных библиотек, имеет внушительный список инструментов для работы с базами данных.

JavaFX - платформа для создания приложений с насыщенным графическим интерфейсом, основанная на Java.

Также необходимо учесть тот факт, что рассылка будет отправляться через корпоративный почтовый ящик, который расположен на домене gmail, поэтому при отправке писем будет использоваться протокол защищенного соединения SSL - (англ. Secure Sockets Layer — уровень защищённых сокетов) — криптографический протокол, подразумевающий более безопасную связь. Он использует асимметричную криптографию для аутентификации ключей обмена, симметричное шифрование для сохранения конфиденциальности, коды аутентификации сообщений для обеспечения целостности сообщений. Для отправки данных будет использоваться протокол SMTP, порт 465.

3.2 Структурная схема базы данных

Для проектирования физической базы данных будет использована программа CA ERwin Data Modeler - CASE-средство для проектирования и документирования баз данных, которое позволяет создавать, документировать и сопровождать базы данных, хранилища и витрины данных.

На рисунке 3.1 приведена физическая схема базы данных.

В качестве СУБД для разработки в бакалаврской работе взята система MySQL, для которой характерен синтаксис Sql. В основу заложен структурированный синтаксис языка Sql, язык напоминает человеческий ЯЗЫК.

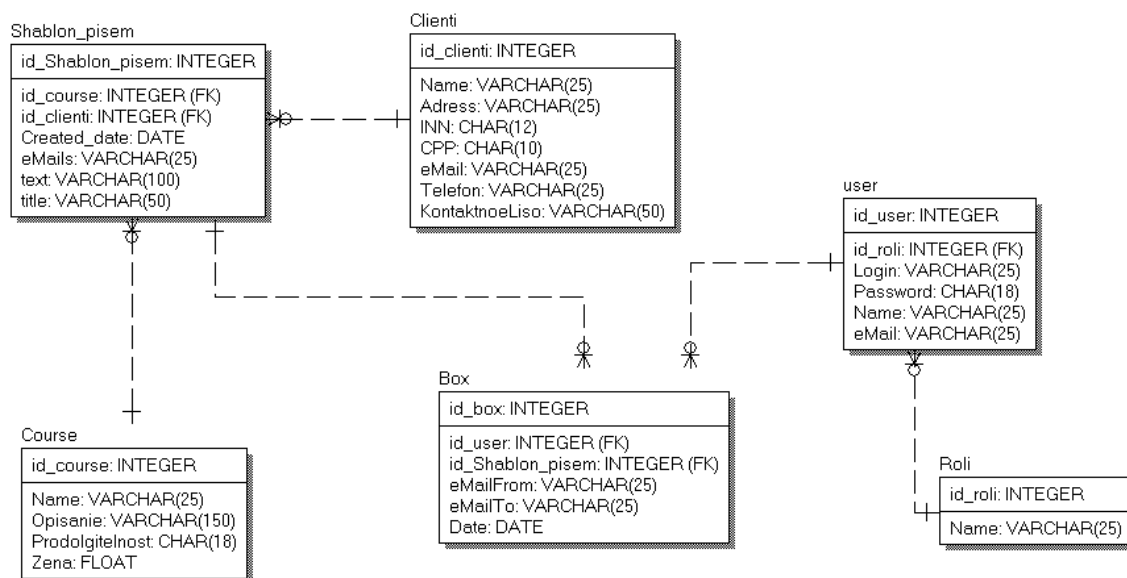


Рисунок 3.1 - Физическая модель данных системы

Команды языка называют запросами, то есть скрипт отправляет запрос на запрашиваемые данные, в результате получает ответ с данными.

Первое слово Sql запроса – глагол, выражающий действие, которое должно выполнить ядро СУБД. Следующая за глаголом часть всегда соответствует формату: имя объекта (таблица), над которым осуществляется действие, а затем – используемые при действии данные.

Как уже описывалось ранее, при разработке web приложения используются фреймворк Spring, содержащий в себе набор библиотек для работы с базами данных, где и обрабатываются запросы через написанный контроллер LoginController. Запросы к базе данных содержатся в слое Repository.

3.3 Программные модули информационной системы

Приложение содержит controller – классы, которые обрабатывают пользовательские запросы с возвратом результата. В контроллере идет обращение к domain, которые получают объекты запросы, а также реализуют определенный функционал программы.

Также есть Repository – слой, где выполняются операции с базой и fxml, где содержатся файлы с внешними шаблонами страниц информационной системы.

На рисунке 3.2 приведена схема пакетов разрабатываемой программы для отправки почтовых сообщений. Рассмотрим более подробно структуру программных модулей.

Папка controller содержит файлы:

- ClientException – класс получения данных пользователей;
- ControllerConfiguration – класс, в котором соединяются шаблоны для страниц из каталога fxml. и базовые контролеры, а именно RegistrationController, MailboxController и других;
- CrudController – класс, который осуществляет проверка полученных из RegistrationController данных;
- FormController – класс, соединяет данные полученные из базы данных с шаблонами форм;

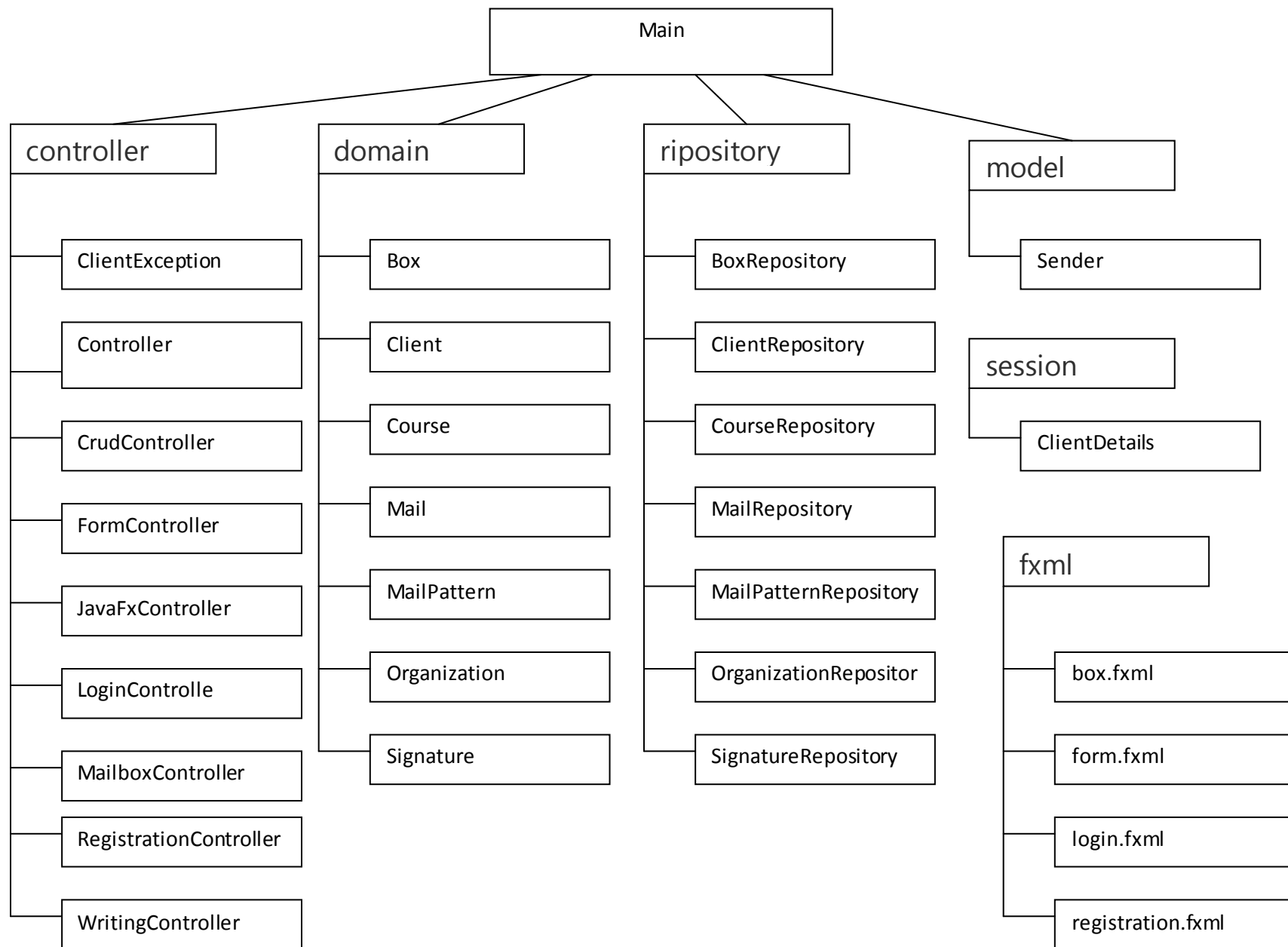


Рисунок 3.2 - Схема пакетов приложения системы

- `JavaFxController` – контроллер с классами для построения диалоговых форм для отправки писем;
- `LoginController` – класс, который подгружает форму авторизации и обрабатывает отправку данных по нажатию кнопки по авторизации данных. Также осуществляет проверку на пустые значения;
- `MailboxController` – класс с mail формы;
- `RegistrationController` - класс, который подгружает форму регистрации и обрабатывает отправку данных по нажатию кнопки по регистрации данных. Также осуществляет отправку данных в `CrudController` для проверки на правильное заполнение полей регистрационных данных;
- `WritingController` - класс, который является расширением `FormController`, служит для обработки данных по созданию шаблона писем и отправку данных в базу данных для сохранения созданного шаблона. А также обработку полученных данных в случае ошибки, при заполнении полей формы, и в случае успеха, выдавая соответствующее сообщение.

Папка `domain` содержит файлы:

- `Box` – класс, генерирующий последовательность полей и колонок для рассылки в базы данных;
- `Client` – класс, генерирующий последовательность полей и колонок клиентов в базы данных;
- `Course` - класс, генерирующий последовательность полей и колонок курсов в базы данных;
- `Mail` - класс, генерирующий последовательность полей и колонок email в базы данных;
- `MailPattern` - класс, генерирующий последовательность полей и колонок рассылок в базы данных;
- `Organization` - класс, генерирующий последовательность полей и колонок организаций в базы данных.

Каждому классу из папки domain соответствует файл с классом каталога repository. Классы этого каталога осуществляют отправление запросов в базу данных и получение ответов для последующей обработки.

В папке model содержится файл Sender, в котором содержится класс для отправки писем через безопасное соединение SSL по протоколу SMTP.

Общий файл для приложения Main позволяет работать пользователю в случае успешной авторизации активирует ControllerConfiguration осуществляет запуск и остановку приложения.

В файле application.properties прописаны общие настройки для подключения к базе данных MySQL.

Файл main.xml содержит подключаемые библиотеки для работы приложения.

Приведем листинги ключевых программных кодов. Файл model\ Sender содержит класс для отправки писем.

```
public class Sender {
    private String username;
    private String password;
    private Properties props;

    public Sender(String username, String password) {
        this.username = username;
        this.password = password;
        props = new Properties();
        props.put("mail.smtp.host", "smtp.gmail.com");
        props.put("mail.smtp.socketFactory.port", "465");
        props.put("mail.smtp.socketFactory.class", "javax.net.ssl.SSLSocketFactory");
        props.put("mail.smtp.auth", "true");
        props.put("mail.smtp.port", "465");
    }

    public void send(String subject, String text, String to) {
        Session session = Session.getInstance(props, new Authenticator() {
            protected PasswordAuthentication getPasswordAuthentication() {
                return new PasswordAuthentication(username, password);
            }
        });
        try {
            Message message = new MimeMessage(session);
            message.setFrom(new InternetAddress(username));
            message.setRecipients(Message.RecipientType.TO, InternetAddress.parse(to));
            message.setSubject(subject);
            message.setText(text);
            Transport.send(message);
        } catch (MessagingException e) {
            throw new RuntimeException(e);
        }
    }
}
```

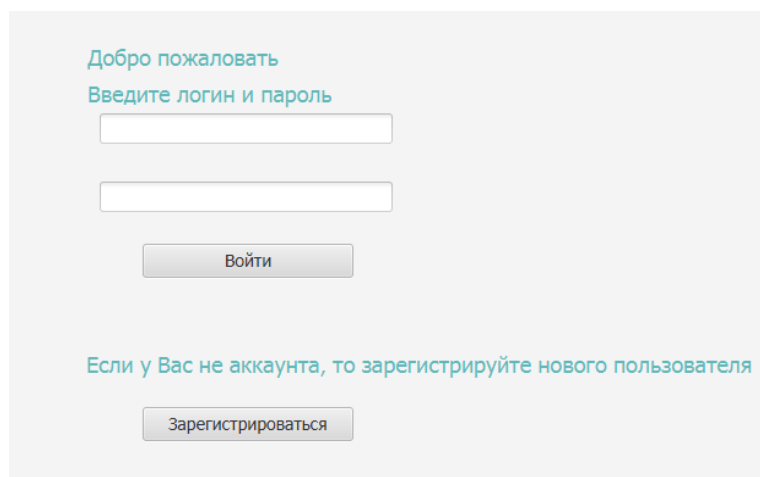
Рисунок 3.3 – класс для отправки писем

Дополнительно код программы приведен в приложении А.

3.4 Интерфейс программы

Рассмотрим интерфейс программы.

В начале запуска программы появляется окно с выбором авторизации и регистрации в случае отсутствия записи.

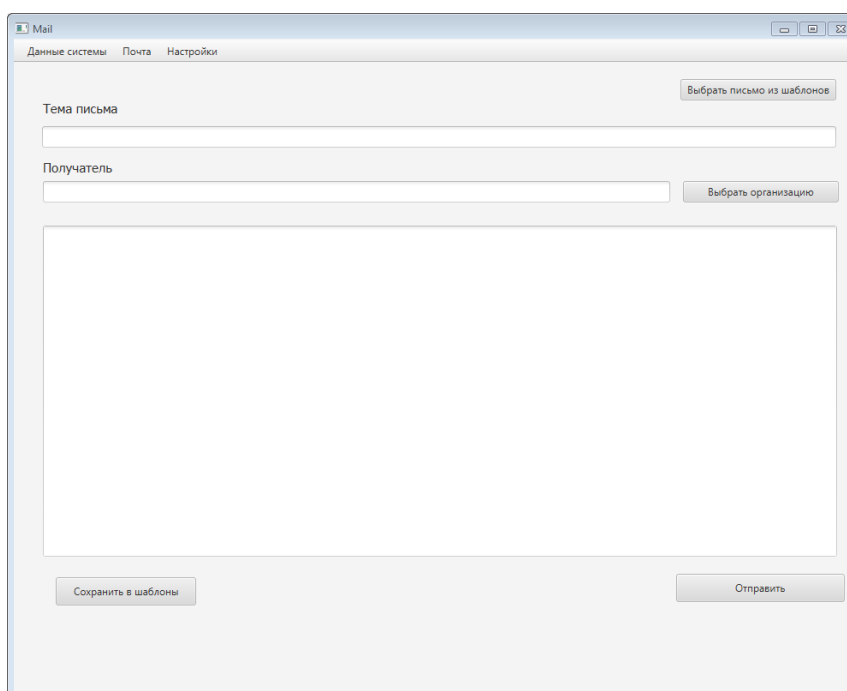


The screenshot shows a light gray window with the following elements:

- Text: "Добро пожаловать" (Welcome) in blue.
- Text: "Введите логин и пароль" (Enter login and password) in blue.
- Two white input fields for login and password.
- A gray button labeled "Войти" (Login).
- Text: "Если у Вас не аккаунта, то зарегистрируйте нового пользователя" (If you don't have an account, register a new user) in blue.
- A gray button labeled "Зарегистрироваться" (Register).

Рисунок 3.4 - Окно запуска программы

В случае успешного входа в программу открывается ее основной интерфейс, приведенный на рисунке 3.5.



The screenshot shows a window titled "Mail" with a menu bar containing "Данные системы", "Почта", and "Настройки". The main area contains:

- A "Тема письма" (Subject) field with a "Выбрать письмо из шаблонов" (Choose email from templates) button to its right.
- A "Получатель" (Recipient) field with a "Выбрать организацию" (Choose organization) button to its right.
- A large empty text area for the email body.
- At the bottom, there are two buttons: "Сохранить в шаблоны" (Save to templates) on the left and "Отправить" (Send) on the right.

Рисунок 3.5 - Внешний вид программы

Затем для добавления получателей нужно выполнить следующее: выбираем пункт: Данные системы – Организации, затем нажимаем кнопку «Добавить».

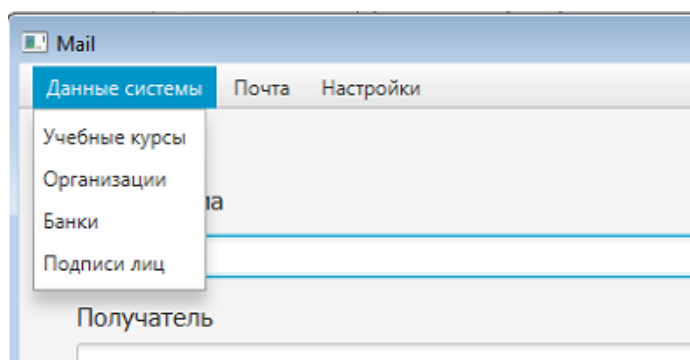


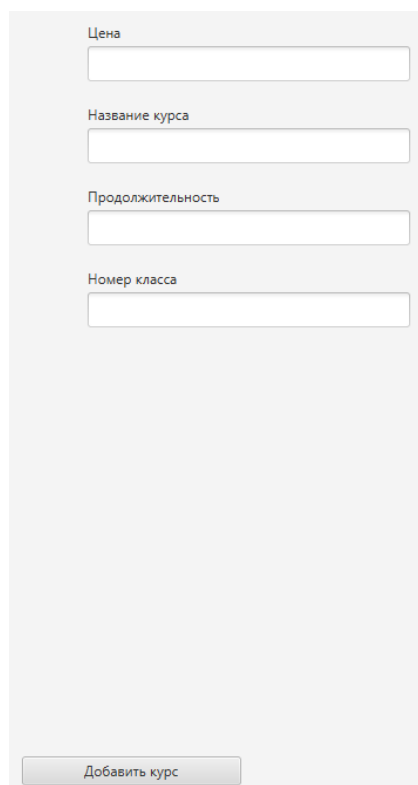
Рисунок 3.6 - Пункт меню «Данные системы»

Затем необходимо внести данные организации и нажать кнопку «Добавить организацию».

A form for adding an organization. It consists of several input fields and a dropdown menu, all arranged vertically. The fields are labeled: 'Владелец', 'Расчетный счет', 'Адрес', 'Телефон', 'Название организации', 'КПП', 'Почта', and 'Введите имя банка'. At the bottom of the form is a button labeled 'Добавить организацию'.

Рисунок 3.7 - Форма добавления организации

Рассмотрим другие формы программы.



Form for adding a course. It contains four input fields:

- Цена (Price)
- Название курса (Course Name)
- Продолжительность (Duration)
- Номер класса (Class Number)

At the bottom, there is a button labeled "Добавить курс" (Add course).

Рисунок 3.8 - Форма добавления курса

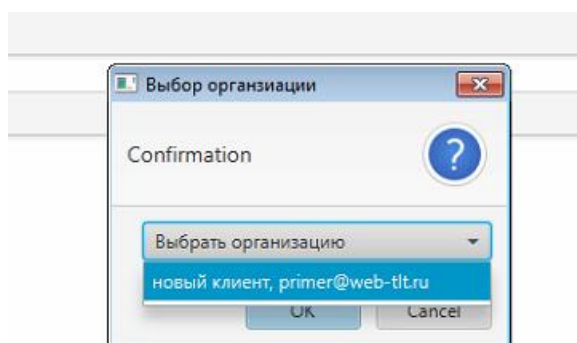


Рисунок 3.9 - Окно выбора получателей

Также можно выбрать шаблон письма, по которому будет определена структура письма, нажав на соответствующую кнопку. На рисунке 3.10 представлена кнопка для загрузки шаблона.



Рисунок 3.10 - Выбрать письмо из шаблонов

На рисунке 3.11 приведен выбранный шаблон со структурой письма.

Данные системы Почта Настройки

Тема письма Выбрать письмо из шаблонов

Получатель Выбрать организацию

Здравствуйте, [Организация.КотактноеЛицо]!

Мы рады пригласить вашу компании. [Организация] на курс [Курс].

[Курс.Описание].

Дата: [Курс.НачалоКурса].

Цена: [Курс.Цена].

С уважением,
[ПодписьТекущегоПользователя]

Сохранить в шаблоны Отправить

Рисунок 3.11 – Шаблон письма с выбранной структурой

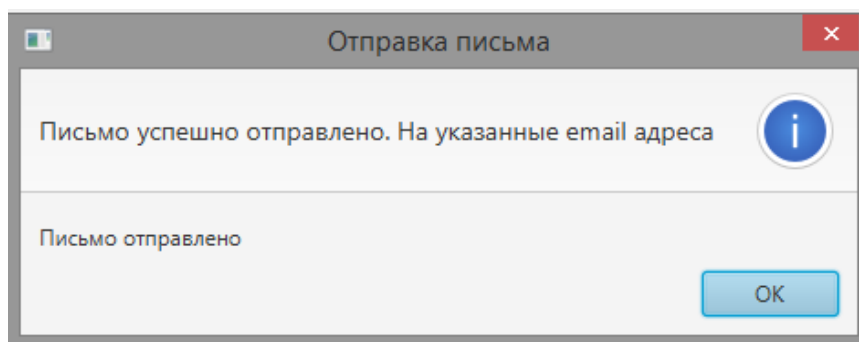


Рисунок 3.12 - Окно сообщения в случае успешного отправления письма

Искать

Мария Шаптала
pr@confetti-city.ru

Мария Шаптала
malisina@yandex.ru

Мастерская Печати и Штампы
pechati27@yandex.ru

Могилев Константин
mka@trubotorg-irk.ru

Молодежный кадровый центр

Рисунок 3.13 - Окно выбора контактных лиц организация

Выводы по главе 3

В 3 главе были выбраны программные средства и технологии для разработки программного обеспечения по рассылке электронных писем. В качестве средств реализации выбраны язык Java, СУБД MySQL, среда программирования IntelliJ Idea, разработан программный код программы, интерфейс и спроектирована физическая модель базы данных. В результате 3 главы удалось создать систему, которая отвечала требованиям и задачам разработки программы, описанными в главе 1.

ЗАКЛЮЧЕНИЕ

Целью данной бакалаврской работы является разработка информационной системы управления рекламными рассылками курсов платного образования.

Структуризация рекламных рассылок и формирование рекламных рассылок под нужды определенного предприятия повышает доверие работодателей к организации, предоставляющей образовательные услуги, и сводит выбор курсов платного образования к необходимому минимуму.

В процессе проектирования проведен анализ предметной области с целью выявления основных объектов автоматизации. Были проанализированы недостатки имеющегося бизнес-процесса по рассылке рекламных объявлений и предложена модель автоматизации рассматриваемого процесса.

Также был проведен анализ аналогичных автоматизированных систем, рассмотрены их достоинства и недостатки.

По результатам исследований были разработаны требования к информационной, позволяющие автоматизировать процесс рассылки рекламных объявлений, используя минимум объектов.

Разработаны логическая и физическая модели данных системы.

Выполнена реализация системы в технологии Java+MySQL.

В результате работы была спроектирована информационная система, которая решает поставленную цель обеспечения индивидуального подхода к клиентам и повышения рейтинга Института дополнительного образования.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

Научная и методическая литература

1. Альфред, В. Ахо Компиляторы. Принципы, технологии и инструментарий / Альфред В. Ахо и др. - М.: Вильямс, 2015. - 689 с.
2. Брюс Эккель. Философия Java. . -- Питер, 2017. – 1168 с.
3. Гарнаев, Андрей WEB-программирование на Java и JavaScript / Андрей Гарнаев , Сергей Гарнаев. - Москва: СПб. [и др.] : Питер, 2017. - 718 с.
4. Герберт Шилдт. Java 8. Полное руководство. - Вильямс, 2017. – 1376 с.
5. Гонсалвес, Энтони Изучаем Java EE 7 / Энтони Гонсалвес. - М.: Питер, 2016. – 640 с.
6. Гупта, Арун Java EE 7. Основы / Арун Гупта. - М.: Вильямс, 2014. - 336 с.
7. Дж. Клейнберг Дж., Е. Тардос. Алгоритмы. Разработка и применение. - Питер, 2106. – 800 с.
8. Кей С. Хорстманн, Гари Корнелл. Java. Библиотека профессионала. Том 1. Основы.- Вильямс, 2015 г.- 864 с.
9. Крис Шефер, Кларенс Хо, Роб Харроп. Pro Spring.- Вильямс, 2017. – 752 с.
10. Кэти Сиерра, Берт Бейтс. Изучаем Java.- Эксмо, 2017. – 720 с.
11. Монахов, В. Язык программирования Java и среда NetBeans (+ CD-ROM) / В. Монахов. - М.: БХВ-Петербург, 2012. - 720 с.
12. Роберт Лафоре. Структуры данных и алгоритмы на Java. -- Питер, 2017. – 704 с.
13. Савитч, Уолтер Язык Java. Курс программирования / Уолтер Савитч. - М.: Вильямс, 2015. - 928 с.
14. Хабибуллин, Ильдар Самоучитель Java / Ильдар Хабибуллин. - М.: БХВ-Петербург, 2016. - 768 с.

Электронные ресурсы

15. Автоматизированные корпоративные системы - необходимый инструмент эффективного ведения бизнеса [Электронный ресурс]. /Составитель: Малькова О.И. Режим доступа: <http://info.tatcenter.ru/economy/16148.htm/>, свободный.
16. Бегунков В.Н. BPwin - ведущий инструмент визуального моделирования бизнес-процессов [Электронный ресурс]./Составитель: Кармилов Е.В. Режим доступа: <http://www.interfase.ru/ca/bpwin.htm#01>, свободный.
17. Ермаков Р.Н. Описание отдельных концепций IDEF0 [Электронный ресурс]/ Составитель: Костин М.А. Режим доступа: <http://www/idef0/ru/idef0/>, свободный.
18. Моделирование компонентов ПО [Электронный ресурс]. /Составитель: Игорев Д.О. Режим доступа: <http://www.interfase.ru/ca/cacase.htm#5>, свободный.
19. Няньков К.Р. Как выбрать АСУ? [Электронный ресурс]. /Составитель: Богатырева У.Д./Дата издания: Режим доступа: <http://www.technet.ru/index.php?r=12&article=1151&archive=1>, свободный.
20. Ребрин Ю.И. Основы экономики и управления производством. [Электронный ресурс]. Режим доступа: <http://www.prof.in.kz/cgi-bin/pro.pl?infa&sho=26988&how=4&cat=6>,. Свободный
21. [Цели использования автоматизированной системы административного управления](#) [Электронный ресурс]. /Составитель: Гогалев Е.Р. Режим доступа: <http://www.amas.ru/docs/amas.htm#1>, свободный.

Литература на иностранном языке

22. Brian Goetz, Tim Peierls, Joshua Bloch, Joseph Bowbeer, David Holmes, Doug Lea, Java Concurrency in Practice, 2016
23. Herbert Schildt, Core Java Volume I: Fundamentals (9th Edition), Core Java Volume II: Advanced Features (9th Edition), 2013

24. Joshua Bloch, "Effective Java (2nd Edition)", 2015
25. Joshua Marinacci, Chris Adamson, Swing Hacks. Tips and Tools for Killer GUIs, 2017
26. Martijn Verburg, The Well-Grounded Java Developer: Vital techniques of Java 7 and polyglot programming. Benjamin J Evans, 2016

ПРИЛОЖЕНИЕ А Код программы

Файл controller\ WritingController расширение класса FormController, контроллер, служит для обработки данных по созданию шаблона писем и отправку данных в базу данных для сохранения созданного шаблона. А также обработку полученных данных в случае ошибки, при заполнении полей формы, и в случае успеха, выдавая соответствующее сообщение.

```
package sample.controller;

import javafx.fxml.FXML;
import javafx.scene.control.*;
import lombok.SneakyThrows;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Qualifier;
import sample.Main;
import sample.domain.Box;
import sample.domain.Mail;
import sample.domain.MailPattern;
import sample.domain.Organization;
import sample.model.Sender;
import sample.repository.*;
import sample.session.ClientDetails;

import java.time.LocalDateTime;
import java.util.*;

public class WritingController extends FormController {
    private static final String USERNAME = "kursy@gmail.com ";
    private static final String PASSWORD = "Ljkbyf123";

    @FXML private TextArea text;
```

```
@FXML private TextField receiver;
@FXML private Button chooseOrg;
@FXML private Button send;
@FXML private Button saveToPatterns;
@FXML private Button choosePattern;
@FXML private TextField title;

@FXML private MenuItem organization;
@FXML private MenuItem course;
@FXML private MenuItem signature;
@FXML private MenuItem bank;
@FXML private MenuItem about;
@FXML private MenuItem changing;
@FXML private MenuItem write;
@FXML private MenuItem logout;

@Autowired private OrganizationRepository repositoryOrg;
@Autowired private BankRepository bankRepository;
@Autowired private CourseRepository courseRepository;
@Autowired private SignatureRepository signatureRepository;
@Autowired private MailPatternRepository mailPatternRepository;
@Autowired private BoxRepository boxRepository;
@Autowired private MailRepository mailRepository;
@Autowired
@Qualifier("login")
private ControllerConfiguration.View login;
@Autowired
@Qualifier("crud")
private ControllerConfiguration.View crud;
@Autowired
```

```

@Qualifier("writing")
private ControllerConfiguration.View writing;
@FXML
@SneakyThrows(Exception.class)
public void initialize() {

    organization.setOnAction(event -> {
        crudController().setTable(repositoryOrg);
        JavaFxController.changing(crud.getView());
    });
    course.setOnAction(event -> {
        crudController().setTable(courseRepository);
        JavaFxController.changing(crud.getView());
    });
    signature.setOnAction(event -> {
        crudController().setTable(signatureRepository);
        JavaFxController.changing(crud.getView());
    });
    bank.setOnAction(event -> {
        crudController().setTable(bankRepository);
        JavaFxController.changing(crud.getView());
    });
    write.setOnAction(event ->
        JavaFxController.changing(writing.getView()));

    changing.setOnAction(event -> {});
    about.setOnAction(event -> JavaFxController.alert("Mail program 1.0v",
        "Автор: ", "Программа для рассылки писем"));
    logout.setOnAction(event -> JavaFxController.changing(login.getView()));
}

```

```

changing.setOnAction(event -> crudController().changeEmail());
/* Writing controller */
chooseOrg.setOnAction(event -> {
    List<Organization> organizations = repositoryOrg.findAll();
    List<String> choices = new ArrayList<>();
    organizations.forEach(e -> choices.add(e.getName() + ", " +
e.getEmail()));
    Dialog<String> dialog = new ChoiceDialog<>("Выбрать
организацию", choices);
    dialog.setTitle("Выбор организации");
    Optional<String> response = dialog.showAndWait();
    response.ifPresent(this::takeEmailFromString);
});

saveToPatterns.setOnAction(event -> {
    if (!title.getText().isEmpty() && title.getText() != null) {
        MailPattern mailPattern = new MailPattern();
        mailPattern.setTitle(title.getText());
        mailPattern.setText(text.getText());
        mailPattern.setEmails(receiver.getText());
        mailPattern.setCreatedDate(LocalDate.now());
        mailPatternRepository.save(mailPattern);
        JavaFxController.alert("Сохранение", "Сохранение шаблона", "Ваш
шаблон был успешно сохранен!");
        return;
    }
    JavaFxController.alert("Сохранение", "Сохранение шаблона",
"Ваш шаблон не был сохранен! " +
"Как минимум заполните поле, тема!");
});

```

```

choosePattern.setOnAction(event -> {
    MailPattern empty = new MailPattern();
    List<MailPattern> patterns = mailPatternRepository.findAll();
    Dialog<MailPattern> dialog = new ChoiceDialog<>(empty, patterns);
    dialog.setTitle("Выбор шаблона");
    Optional<MailPattern> response = dialog.showAndWait();
    response.ifPresent(this::fillLetter);
});

```

```

send.setOnAction(event -> {
    ClientDetails clientDetails = (ClientDetails) Main.scenes.get("client");
    String title = this.title.getText();
    String emails = this.receiver.getText();
    String context = this.text.getText();
    if (title != null && emails != null && context != null) {
        Sender sender = new Sender(USERNAME, PASSWORD);
        String[] receivers = emails.split(",");
        Mail letter = new Mail();
        letter.setMsg(context);
        letter.setTitle(title);
        List<Box> letters = new ArrayList<>();
        for (String receiver : receivers) {
            sender.send(title, context, receiver);
            Box let = Box.builder()
                .from(clientDetails.getEmail())
                .to(receiver)
                .mail(letter)
                .sentDate(LocalDateTime.now())
                .client(clientDetails.buildClient())

```

```

        .build();
        letters.add(let);
    }
    mailRepository.save(letter);
    boxRepository.saveAll(letters);
}
JavaFxController.alert("Отправка письма",
    "Письмо отправлено",
    "Письмо успешно отправлено. На указанные email адреса");
});
}
private void fillLetter(MailPattern pattern) {
    receiver.setText(pattern.getEmails());
    title.setText(pattern.getTitle());
    text.setText(pattern.getText());
}
private void takeEmailFromString(String stringWithEmail) {
    String email = stringWithEmail.trim().split(",")[1].trim();
    String receiverEmails = receiver.getText();
    if (receiverEmails == null || receiverEmails.isEmpty()) {
        receiver.setText(email);
    } else {
        receiver.setText(receiverEmails + "," + email);
    }
}
private CrudController crudController() {
    return (CrudController) crud.getController();
}
}

```