МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ МАРИЙ ЭЛ

Государственное бюджетное профессиональное образовательное учреждение Республики Марий Эл «Йошкар-Олинский технологический колледж»

# КУРСОВОЙ ПРОЕКТ

# МДК 06.03 Устройство и функционирование информационной системы

**ПМ.06 Сопровождение информационных систем**

Специальность 09.02.07

Специалист по информационным системам

РАЗРАБОТКА ИНФОРМАЦИОННОЙ СИСТЕМЫ

«**Сведения о Турагентстве «TOURIST»»**»

Студент Долгирев Д.Ю.

Группа И-41

Преподаватель Колесников Е.

Йошкар-Ола, 2021г

**Содержание**

Введение 2

1 Структурный системный анализ 3

1.1 Описание предметной области 5

1.2 Разработка функциональной модели информационной системы 6

2 Реализация и разработка проекта базы данных 10

2.1 Инфологическое проектирование 10

2.2 Логическая модель данных 14

2.3 Физическая структура информационной системы 15

3 Разработка интерфейса 16

3.1 Разработка формы меню 16

Заключение 19

Список информационных источников 20

**Тема «Разработка и проектирование ИС «Сведения о Турагентстве «TOURIST»»»**

**Введение**

Для успешного и эффективного функционирования практически любой системы необходимы ведение контроля, анализа и внедрение автоматизации для отдельных процессов или системы в целом. Автоматизация позволяет повысить производительность и качество системы, оптимизировать процессы управления, снизить затраты.

Целью данной курсовой работы является проектирование информационной системы сети турагентства, которая позволила бы значительно улучшить качество обслуживания клиентов и упростить процессы оформления продаж путёвок, а также достичь максимального результата работы сети туристического агентства.

Для описания предметной области и проектируемой системы будем использовать диаграммы UML. Язык моделирования UML предоставляет выразительные средства для создания визуальных моделей и обладает рядом значимых преимуществ:

* UML объектно-ориентирован, в результате чего методы описания результатов анализа и проектирования семантически близки к методам программирования на современных объектно-ориентированных языках;
* UML позволяет описать систему практически со всех возможных точек зрения и разные аспекты поведения системы;
* Диаграммы UML сравнительно просты для чтения после быстрого ознакомления с его синтаксисом;

UML получил широкое распространение и динамично развивается.

1. **Структурный системный анализ**

Турагентство называется “TOURIST”

В современном обществе информация стала полноценным ресурсом производства, важным элементом социальной и политической жизни общества. Качество информации определяет качество управления.

В последнее время все больше предприятий сталкиваются с проблемой улучшения управляемости компании: улучшение контроля и ускорение бизнес-процессов, улучшение возможности их отслеживания, оптимизация рабочего времени, экономия трудозатрат, повышение производительности труда и так далее. Единственным способом реализации подобных задач является внедрение информационной системы.

Применение современных информационных технологий имеет важное значение для оптимизации внутренних процессов организации, оперативного доведения информации до исполнителей, улучшения взаимодействия подразделений и отдельных исполнителей в процессе работы с документами, контроля исполнения документов и поручений, поиска информации и определения стадии исполнения документов и их местонахождения, то есть, в конечном счете, способствует более оперативному и качественному решению вопросов, которым посвящены документы. Главное при этом - улучшение взаимодействия всех подразделений организации, повышение управляемости, а также достижение более высокой оперативности в работе.

**Актуальность темы** определяется тем, что информационные системы составляют в настоящее время основу компьютерного обеспечения информационных процессов, входящих практически во все сферы человеческой деятельности автоматизировать процесс ведения документации и отчетности.

О своевременности и актуальности рассматриваемой проблемы говорит тот факт, что большую часть своего времени, администратор туристического агентства на оформление различной документации и отчетов.

Выше изложенное в целом определило **цель исследования**: повышение эффективности работы администратора турагентства за счет разработки и внедрения информационной системы.

Данная информационная система предназначена для хранения информации о ценах, о агентствах, сотрудничающих с турагентством и, с возможностью внесения данных, выборки и изменения данных, вывода информации в необходимом формате.

**Объект исследования**: работа администратором турагентства.

**Предмет исследования**: информационная система, автоматизирующая работу администратора турагентства.

В соответствии с поставленной целью в проекте определены следующие задачи исследования:

* 1. На основе теоретического анализа литературы и Internet-источников произвести анализ предметной области турагентства.
  2. Провести функционально-ориентированное проектирование информационной системы.
  3. Разработать инфологическую модель информационной системы.
  4. Спроектировать логическую структуру информационной системы.
  5. Разработать физическую структуру информационной системы.
  6. Разработать запросы и отчеты к информационной системе.
  7. Разработать интерфейс БД.
  8. Создать руководство пользователя.

**1.1 Описание предметной области**

При разработке информационной системы «TOURIST» было проведено обследование деятельности турагентства по следующим источникам:

* Администратором туристического агентства были предоставлены необходимые нормативные документы по правилам оформления путёвок и прочее;
* журнал, содержащий условия заключения договора с различными агентами;
* литература и Internet-источники, описывающие работу турагентства.

Таким образом, в результате обследования предметной области были определены следующие **входные данные**:

* информация о агентах,
* информация о документах,
* информация о платежах,
* информация об условиях оформления путёвок.

К **выходным данным** относятся отчеты об оплатах за путёвки, сведения об агентах

Для создания информационной модели турагентства необходимо осуществить формальное описание его работы. Изучение руководящих документов является первым шагом в изучении процессов работы туристического агентства. Затем, на основе эталонных знаний, исследуется реальная работа администратора туристического агентства и сравнивается с указанием руководящих документов. Многократное изучение руководящих документов, периодическое наблюдение за реальными действиями администратора и комментарии этого работника позволяют получить знания о функционировании работы туристического агентства, разработать модели.

Целью решения данной задачи является сведение к минимуму работы администратора турагентства с бумажными носителями, что ускорит процесс обработки поступающей информации, сократив время ожидания для приёма заявок на оформление путёвок от клиентов, исключит возможную путаницу информации.

# 1.2 Разработка функциональной модели информационной системы

Проектирование информационной системы «TOURIST» начинается с этапа построения бизнес-процессов.

Для описания бизнес-процессов, подлежащих учету в информационной системе «Туристическое агентство», используется функциональное моделирование.

Любое действие может быть декомпозировано, т.е. разложено на более мелкие, которые, в свою очередь могут быть декомпозированы, и т.д. до уровня разумной достаточности.

Диаграмма вариантов использования в UML — диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Основное назначение диаграммы — описание функциональности и поведения, позволяющее заказчику, конечному разработчику и пользователю совместно обсуждать проектируемую или существующую систему.

При моделировании системы с помощью диаграммы прецедентов системный аналитик стремится:

* чётко отделить систему от её окружения;
* определить действующих лиц (акторов), их взаимодействие с системой и ожидаемую функциональность системы;
* определить в глоссарии предметной области понятия, относящиеся к детальному описанию функциональности системы (то есть прецедентов).

Работа над диаграммой может начаться с текстового описания, полученного при работе с заказчиком. При этом нефункциональные требования (например, конкретный язык или система программирования) при составлении модели прецедентов опускаются (для них составляется другой документ).

На диаграмме использования изображаются:

* акторы — группы лиц или систем, взаимодействующих с нашей системой;
* варианты использования (прецеденты) — сервисы, которые наша система предоставляет акторам;
* комментарии;
* отношения между элементами диаграммы.

На мой взгляд, наиболее правильный порядок построения диаграммы следующий:

1. выделить группы действующих лиц (работающих с системой по-разному, часто из-за различных прав доступа);
2. идентифицировать как можно больше вариантов использования (процессов, которые могут выполнять пользователи). При этом не следует делить процессы слишком мелко, нужно выбирать лишь те, которые дадут пользователю значимый результат. Например, кассир может «продать товар» (это будет являться прецедентом), однако «ввод штрих-кода товара для получения цены» самостоятельным прецедентом не является;
3. дополнить прецеденты словесным описанием (сценарием):

* для каждого прецедента создать разделы: «главная последовательность» и «альтернативные последовательности»;
* при составлении сценария нужно упорно задавать заказчику вопросы «что происходит?», «что дальше?», «что еще может происходить?» и записывать ответы на них.

Сценарии являются очень важной частью диаграмм использования, хотя их формат и не регламентирован. Ряд авторов предлагает использовать псевдокод для представления сценария и даже сразу строить диаграммы деятельности или взаимодействия, но на мой взгляд, наиболее предпочтительным вариантом на этапе построения use-case диаграмм является текстовый, описывающий систему с точки зрения пользователя (т.к. именно этот формат будет наиболее понятен заказчику, с которым вам предстоит согласовывать техническое задание).

Изображение диаграммы прецедентов представлено на рисунке 1

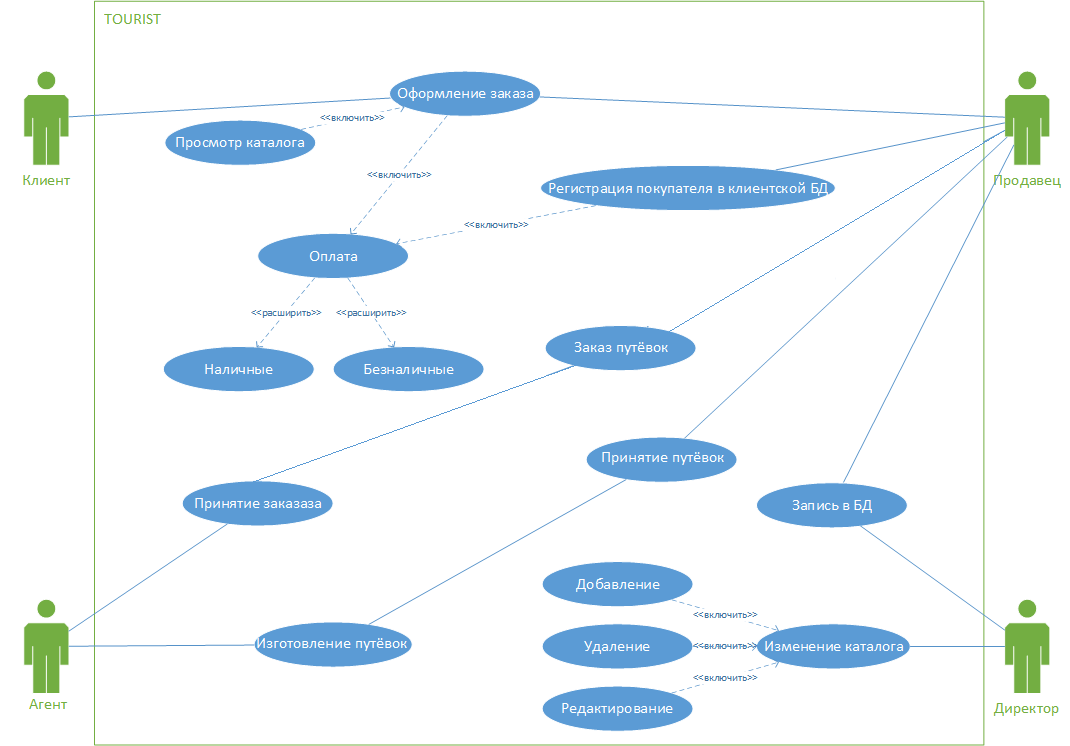


Рисунок 1. Диаграмма прецедентов

**Диаграмма последовательности** *(*[*англ.*](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA)*sequence diagram)* — [UML-диаграмма](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0_(UML)), на которой для некоторого набора объектов на единой временной оси показан жизненный цикл объекта (создание-деятельность-уничтожение некой сущности) и взаимодействие актеров (действующих лиц) информационной системы в рамках [прецедента](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82_(UML)).

Основными элементами диаграммы последовательности являются обозначения [объектов](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82_(%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5)) (прямоугольники с названиями объектов), вертикальные «линии жизни» ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *lifeline*), отображающие течение времени, прямоугольники, отражающие деятельность объекта или исполнение им определенной функции (прямоугольники на пунктирной «линии жизни»), и стрелки, показывающие обмен сигналами или сообщениями между объектами.

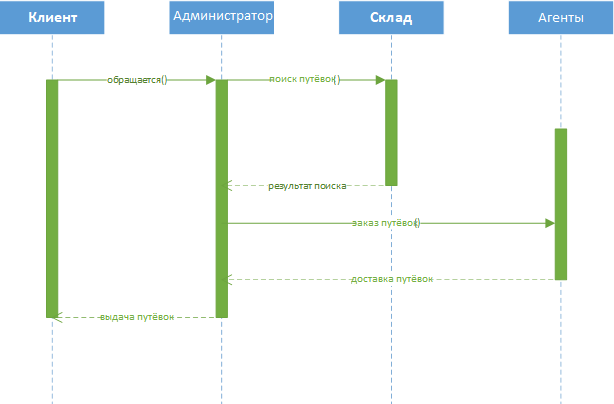


Рисунок 2. Диаграмма последовательности

# Разработка и реализация проекта базы данных

* 1. **Инфологическое проектирование**

В разрабатываемой информационной системе, чтобы не допустить избыточность данных были спроектированы следующие сущности в соответствии с определенными входными данными:

Сущность «Туры» содержит информацию о турах

Сущность «Агенты» содержит информацию об агентах

товаров.

**ER-модель** (от [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Entity-Relationship model*, *модель «*[*сущность*](https://ru.wikipedia.org/wiki/%D0%A1%D1%83%D1%89%D0%BD%D0%BE%D1%81%D1%82%D1%8C_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0))*— связь»*) — [модель данных](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), позволяющая описывать [концептуальные схемы](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BD%D1%86%D0%B5%D0%BF%D1%82%D1%83%D0%B0%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F_%D1%81%D1%85%D0%B5%D0%BC%D0%B0) [предметной области](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%B5%D0%B4%D0%BC%D0%B5%D1%82%D0%BD%D0%B0%D1%8F_%D0%BE%D0%B1%D0%BB%D0%B0%D1%81%D1%82%D1%8C).

ER-модель используется при высокоуровневом (концептуальном) [проектировании баз данных](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%B1%D0%B0%D0%B7_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85). С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями.

Во время [проектирования баз данных](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%B1%D0%B0%D0%B7_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) происходит преобразование схемы, созданной на основе ER-модели, в конкретную [схему базы данных](https://ru.wikipedia.org/wiki/%D0%A1%D1%85%D0%B5%D0%BC%D0%B0_%D0%B1%D0%B0%D0%B7%D1%8B_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) на основе выбранной модели данных ([реляционной](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85), [объектной](https://ru.wikipedia.org/w/index.php?title=%D0%9E%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BD%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85&action=edit&redlink=1), [сетевой](https://ru.wikipedia.org/wiki/%D0%A1%D0%B5%D1%82%D0%B5%D0%B2%D0%B0%D1%8F_%D0%BC%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) или др.).

ER-модель представляет собой формальную конструкцию, которая сама по себе не предписывает никаких графических средств её визуализации. В качестве стандартной графической нотации, с помощью которой можно визуализировать ER-модель, была предложена *диаграмма «сущность-связь»* ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Entity-Relationship diagram, ERD*, *ER-диаграмма*).

Понятия «ER-модель» и «ER-диаграмма» часто не различают, хотя для визуализации ER-моделей могут быть использованы и другие графические нотации, либо визуализация может вообще не применяться (например, использоваться текстовое описание).

Модель была предложена в [1976 году](https://ru.wikipedia.org/wiki/1976_%D0%B3%D0%BE%D0%B4) [Питером Ченом](https://ru.wikipedia.org/wiki/%D0%A7%D0%B5%D0%BD,_%D0%9F%D0%B8%D1%82%D0%B5%D1%80), им же предложена и самая популярная графическая нотация для модели.

* **Концептуальная модель данных** — схема наивысшего уровня с минимальным количеством подробностей. Достоинство этого подхода заключается в возможности отобразить общую структуру модели и всю архитектуру системы. Менее масштабные системы могут обойтись и без этой модели. В этом случае можно сразу переходить к логической модели.
* **Логическая модель данных** содержит более подробную информацию, нежели концептуальная модель. На этом уровне определяются более подробные операционные и транзакционные сущности. Логическая модель не зависит от технологии, в которой она будет применяться.
* **Физическая модель данных**: на основе каждой логической модели данных можно составить одну или две физических модели. В последних должно присутствовать достаточно технических подробностей для составления и внедрения самой базы данных.

Обращаем ваше внимание на тот факт, что похожие уровни масштаба и детализации встречаются и в других видах схем (например, в диаграммах DFD), однако данная классификация отличается от трехсхемного подхода в разработке ПО, где деление информации осуществляется по несколько иному принципу. Правда, иногда разработчики применяют ER-диаграммы с дополнительными иерархиями, если дизайн базы данных требует больше информационных уровней. К примеру, разработчик может добавить новые группы по принципу расширения вверх (суперклассы) и вниз (подклассы).

* **Только реляционные данные**. Следует четко понимать, что цель ER-диаграмм — показать связи и отношения между элементами, поэтому они отображают только реляционную структуру.
* **Только для структурированных данных**. Данные должны быть четко разбиты на поля, столбцы и строки, иначе пользы от ER-диаграммы будет мало. Это касается и частично структурированных данных, так как только некоторые из них будут пригодны для работы.
* **Сложность интеграции с существующей базой данных**. Применение ER-моделей для интеграции с существующей базой данных — непростая задача по причине различия в архитектурах.

Области применения диаграмм

* **Проектирование баз данных**. ER-диаграммы применяются для моделирования и проектирования реляционных баз данных, причем как в плане логических и бизнес-правил (логические модели данных), так и в плане внедрения конкретных технологий (физические модели данных). В сфере разработки программного обеспечения ER-диаграмма, как правило, служит первым шагом в определении требований проекта по созданию информационных систем. На дальнейших этапах работы ER-диаграммы также применяются для моделирования конкретных баз данных. Реляционная база данных сопровождается соответствующей реляционной таблицей и при необходимости может быть представлена в этом формате.
* **Отладка баз данных**. ER-диаграммы применяются для анализа уже имеющихся баз данных с целью выявить и устранить ошибки в логике или развертывании. Диаграмма позволяет выявить, где именно закрались ошибки.
* **Информационные системы для бизнеса**. ER-схемы используются для проектирования и анализа реляционных баз данных, применяемых в бизнес-процессах. Реляционные базы данных могут пригодиться в любом бизнес-процессе, где задействованы данные, разбитые на поля, включая сущности, действия и взаимосвязи. Базы данных помогают оптимизировать процессы, извлекать данные и повышать качество результатов.
* **Реорганизация бизнес-процессов (BPR)**. ER-диаграммы помогают анализировать базы данных, применяемые при реорганизации бизнес-процессов и моделировании новых баз данных.
* **Образование**. Базы данных — широко распространенный в наши дни способ хранения реляционной информации, применяемой в целях образования и для последующего извлечения данных, поэтому ER-диаграммы играют не последнюю роль в планировании подобных структур данных.
* **Исследовательская деятельность**. Поскольку исследовательская работа во многом опирается на четко структурированные данные, ER-диаграммы играют ключевую роль в построении оптимальных баз данных для анализа информации.

Диаграммы «сущность-связь» (или ERD) — неотъемлемая составляющая процесса моделирования любых систем, включая простые и сложные базы данных, однако применяемые в них фигуры и способы нотации могут запросто ввести в заблуждение любого.

Это руководство поможет вам стать настоящим экспертом по нотации ER-диаграмм и уверенно взяться за моделирование собственных баз данных!

Концептуальные модели данных дают общее представление о том, что должно входить в состав модели. Концептуальные ER-диаграммы можно брать за основу логических моделей данных. Их также можно использовать для создания отношений общности между разными ER-моделями, положив их в основу интеграции.

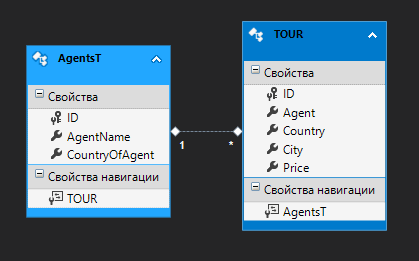


Рисунок 3. ER-диаграмма

# 2.2 Логическая модель данных.

Преобразование ER–диаграммы в схему БД выполняется путем сопоставления каждой сущности и каждой связи, имеющей атрибуты, отношения (таблицы БД).

В данной информационной системе, разработаны связи один ко многим (1:М), которые указаны в таблице 1

|  |  |  |
| --- | --- | --- |
| **Родительская таблица** | **Дочерняя таблица** | **Тип связи** |
| Products | Clients | 1:M |
| Products | Images | 1:M |
| Products | ProductSales | 1:M |
| Products | Manufacturers | 1:M |

Таблица 1. Классификация связей информационной системы (пример)

# Физическая структура информационной системы

Для создания информационной системы используются средства MySQL Workbench

В таблице «Туры» представлена информация о турах.

Структура таблицы: поля, их типы, и размер представлены в таблице 2**.**

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** |
| ID | int |  |
| Agent | nvarchar | MAX |
| Country | nvarchar | MAX |
| City | nvarchar | MAX |
| Price | int |  |

Таблица 2. «Туры»

|  |  |  |
| --- | --- | --- |
| **Имя поля** | **Тип данных** | **Размер** |
| ID | Int |  |
| AgentName | Varchar | MAX |
| CountryOfAgent | Varchar | MAX |

Таблица 3. «Агенты»

# 3. Разработка интерфейса

**3.1. Разработка формы-меню**

|  |  |
| --- | --- |
| **Команда** | **Назначение** |
| Добавить | Открывается окно с добавлением нового тура |
| Редактировать | Открывается окно с возможностью редактирования тура |
| Удалить | Удаляет выбранную запись |
| Обновить таблицу | Обновляется список |
| Выход | Выход из приложения |

Таблица 4. Основное меню

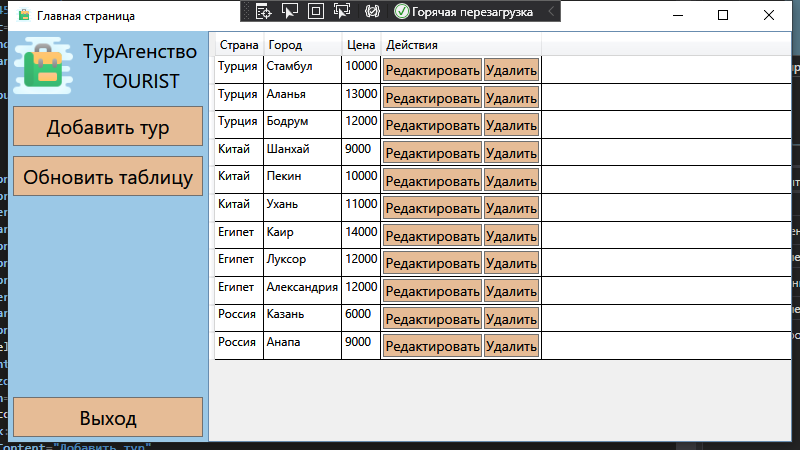


Рисунок 4. Главная страница

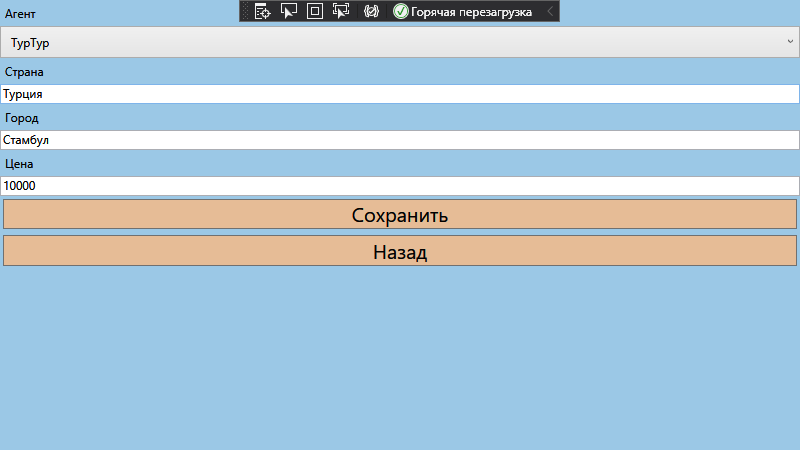


Рисунок 5. Страница **добавления нового** тура

|  |  |
| --- | --- |
| **Команда** | **Назначение** |
| Сохранить | Происходит сохранение добавленного тура |
| Назад | Происходит возврат на главную страницу |

Таблица 5. Страница **добавления нового** тура

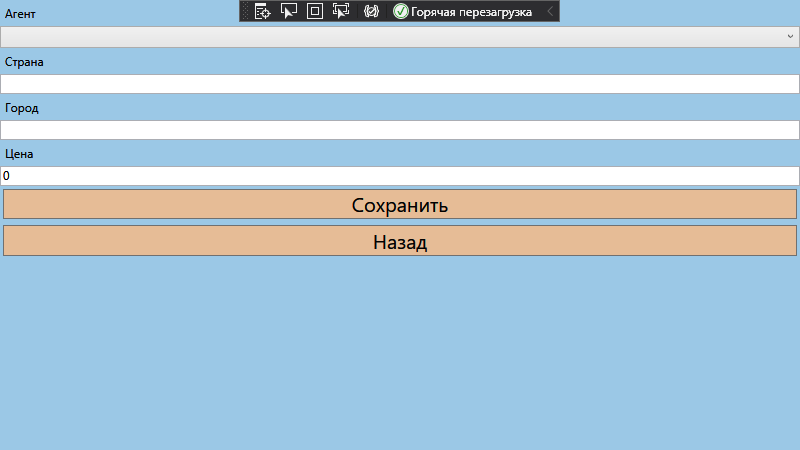


Рисунок 5. Страница **редактирования старого** тура

|  |  |
| --- | --- |
| **Команда** | **Назначение** |
| Сохранить | Происходит сохранение отредактированного тура |
| Назад | Происходит возврат на главную страницу |

Таблица 5. Страница **редактирования старого** тура

**Заключение**

В ходе выполнения курсового проекта средствами MySQL Workbench была разработана информационная система «Турагентство «TOURIST»», предназначенная для автоматизации работы администратора турагентства.

Концептуальная модель данных представлена диаграммой «сущность- связь». На ее основании разработана логическая структура базы данных, в ходе реализации которой задействованы механизмы обеспечения целостности данных.

Создание главной формы доступа к данным позволяет пользователям легко просматривать, обновлять или анализировать данные.

Разработанное приложение просто в применении и может быть использовано в любом турагентстве.

Формы отображают информацию из таблиц в удобном виде, а также служат для ввода данных в таблицы.

Сопоставление результатов проекта с поставленными задачами позволяет заключить следующее:

* + 1. На основе теоретического анализа литературы и Internet-источников произведен анализ предметной области турагентства.
    2. Проведено функционально-ориентированное проектирование информационной системы.
    3. Разработана инфологическая модель данных.
    4. Спроектирована логическая структура информационной системы.
    5. Разработана физическая структура информационной системы.
    6. Разработаны запросы, отчеты к информационной системе.
    7. Разработан интерфейс БД.

**Список информационных источников**

1. Бурков А.В. Проектирование информационных систем [Текст]/ Бурков А.В.- Йошкар-Ола, МарГУ 2009 г. – 312с.
2. ВикрамВасвани. Полный справочник по MySQL. Москва, 2006.
3. Гвоздева Т.В., Баллод Б.А. Проектирование информационных систем DJVU [Текст]/ Твоздева Т.В., Баллод Б.А. -Ростов н/д.: Феникс, 2009. – 512 с.
4. Горохова Т.Н. Разработка и эксплуатация информационных систем Учебное пособие для студентов среднего профессионального образования. [Текст]/ Горохова Т.Н. – Санкт-Петербург: Санкт-Петербургский колледж управления и экономики "Александровский лицей", 2010. – 84с.
5. Громаков Е.И. Проектирование интегрированных компьютерных систем управления: Учебно-методическое пособие PDF [Текст]/ Громаков Е.И – Томск: Изд-во Томского политехнического университета, 2012 г.– 168 с
6. Джон КоггЗолл. РНР 5. Полное руководство. Диалектика, 2006
7. Казаченко, Колобашкина Т.В. и др. Безопасность жизнедеятельности. Промышленная и экологическая безопасность. Методические указания к дипломному проектированию, СПб-ГУАП, 2001
8. Коваленко В.В. Проектирование информационных систем Учеб.пособие [Текст]/ Коваленко В.В.-Рязань, Рязанский государственный радиотехнический университет, 2006г. – 184 с.
9. Люк Веллинг, Лора Томсон. Разработка Web – приложений с помощью РНР и MySQL.Вильямс, 2005
10. Мазупкевич А. РНР. Настольная книга программиста. Новое издание,2003
11. Определение эффективности инвестиций. Методические указания к выполнению дипломного проекта. Спб, 2002
12. Прогнозирование элементов бизнес-плана проектов. Методические указания к выполнению дипломных проектов. СПб 2002.
13. Роберт Дж. Мюлер. Базы данных и UML. Справочник в электронном виде, 2001
14. Смирнов Н.В. Проектирование информационных систем [Текст]/ Смирнов Н.В. – Санкт -Петербург, БГТУ "ВОЕНМЕХ", 2008 г. – 146 с.
15. Справочное руководство пользователя по пакету RationalRose.
16. Трофимов С.А. Case – технологии. Практическая работа в RationalRose. Бином 2001, 272с.
17. УндиБоггс, Майкл Боггс. UML и RationalRose. Лори, 2004
18. Виды диаграмм [Электронный ресурс] – Режим доступа: http://www.intuit.ru/studies/courses/1007/229/lecture/5954?page=2 – 15.11.2016г.
19. http://www.exponenta.ru/soft/others/mvs/stud2/23.asp – 20.11.2016г. – Компьютерный лабораторный практикум "Моделирование"[Электронный ресурс]
20. Режим доступа: http://citforum.ru/database/advanced\_intro/31.shtml – 16.11.2016 г. – Кузнецов С. Базы данных. Вводный курс [Электронный ресурс]
21. Репозитории преподавателя github.com/kolei