

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет имени
К.И. Сатпаева

Институт информационных и телекоммуникационных технологий

Кафедра кибербезопасность, обработка и хранение информации

Мухпул Мухаббат Айнидинкызы

Разработка информационной системы туристической фирмы

ДИПЛОМНАЯ РАБОТА

Специальность 5В070300 – Информационные системы

Алматы 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет имени
К.И. Сатпаева

Институт информационных и телекоммуникационных технологий

Кафедра кибербезопасность, обработка и хранение информации

ДОПУЩЕН К ЗАЩИТЕ

Заведующий кафедрой
КБОиХИ, канд.техн.наук,
ассистент-профессор.

 Н.А. Сейлова

«16» мая 2019 г.

ДИПЛОМНАЯ РАБОТА


На тему: «Разработка информационной системы туристической фирмы»

Специальность 5В070300 – Информационные системы

Выполнил

Мухпул М.А.

Рецензент,
заместитель руководителя Национальной
научной лаборатории коллективного
пользования информационных и
космических технологий НАО КазННТУ
имени К.И.Сатпаева, PhD

 К. А. Бостанбеков

«15» мая 2019 г.

Научный руководитель,
PhD, сениор лектор

 О.В. Киселева

«15» мая 2019 г.

Алматы 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Казахский национальный исследовательский технический университет имени
К.И. Сатпаева

Институт информационных и телекоммуникационных технологий

Кафедра кибербезопасность, обработка и хранение информации

5B070300 – Информационные системы

УТВЕРЖДАЮ

Заведующий кафедрой
КБОиХИ, канд.техн.наук,
ассистент-профессор.

 Н.А. Сейлова

« 16 » мая 2019 г.

ЗАДАНИЕ

на выполнение дипломной работы

Обучающемуся Мухпул Мухаббат Айнидинкызы

Тема: Разработка информационной системы туристической фирмы.

Утверждена приказом Ректора Университета №116 2-б от 16.10.2018 г.

Срок сдачи законченной работы «17» мая 2019 г.

Исходные данные к дипломной работе: результат литературного обзора современного состояния по данной теме, сбор теоретического материала.

Краткое содержание дипломной работы:

- а) оценка туристических фирм;
- б) разработка сайта;

Перечень графического материала (с точным указанием обязательных чертежей): представлены 10 слайдов презентации работы.

Рекомендуемая основная литература: из 17 наименований



ГРАФИК

подготовки дипломной работы

| Наименование разделов, перечень разрабатываемых вопросов | Сроки представления научному руководителю и консультантам | Примечание |
|--|---|------------|
| Обзор и анализ туристических компаний | 10.01.2019 – 08.03.2019 | |
| Разработка сайта | 05.02.2019 – 10.03.2019 | |
| Настройка ИС для сайта | 11.03.2019 – 28.04.2019 | |

Подписи

консультантов и нормоконтролера на законченную дипломную работу с указанием относящихся к ним разделов работы

| Наименование разделов | Консультанты, И.О.Ф. (уч. степень, звание) | Дата подписания | Подпись |
|-----------------------|--|-----------------|---|
| Разработка ПО | М.Т.Н М.Б. Бауржан | 17.05.19 |  |
| Нормоконтроллер | Сениор-лектор, PhD О. В. Киселева | 15.05.19 |  |

Научный руководитель



Киселева О. В.

Задание принял к исполнению обучающийся



Мухпул М.А.

Дата

" 08 " 01 2019 г.

РЕЦЕНЗИЯ

на дипломный проект

Мухпул Мухаббат Айнидинкызы

Специальности 5В070300 Информационные системы

На тему: «Разработка информационной системы туристической фирмы»

Выполнено:

- а) графическая часть на 5 листах
б) пояснительная записка на 10 страницах

ЗАМЕЧАНИЯ К РАБОТЕ

Данная дипломная работа посвящена разработке информационной системы туристической компании. Система позволяет искать информацию о турах и бронировать их, предоставляет пользователю полную информацию о туре, выборе маршрутов, стоимости и условиях. Информационная система реализована в виде веб приложения, используя веб фреймворк Django на языке Python, и СУБД SQLite.


В первом разделе проводится обзор предметной области и анализ существующих систем бронирования. Во втором разделе описываются функциональные требования к системе и обосновывается выбор СУБД, языка программирования и веб фреймворка. В третьем разделе описана системное программное обеспечение. В данной работе недостаточно развернуто описана программная реализация информационной системы.

ОЦЕНКА РАБОТЫ

В ходе реализации дипломного проекта все поставленные задачи были решены. Разработанная информационная система является завершенной и дипломная работа соответствует всем требованиям. Учитывая выше изложенное, считаю, что дипломная работа на тему: «Разработка информационной системы туристической фирмы» заслуживает оценки не ниже 90 баллов (Отлично), а Мухпул Мухаббат Айнидинкызы присвоения академической степени бакалавра по специальности 5В070300 – Информационные системы.

Рецензент

Заместитель руководителя Национальной
научной лаборатории коллективного пользования
информационных и космических технологий
НАО КазННТУ имени К.И. Сатпаева, PhD

 К. А. Бостанбеков

«15» нояб 2019 г.

ОТЗЫВ

НАУЧНОГО РУКОВОДИТЕЛЯ

на дипломную работу
(наименование вида работы)

Мухпул Мухаббат Айнидинкызы
(Ф.И.О. обучающегося)

5B070300 – Информационные системы
(шифр и наименование специальности)

Тема: Разработка информационной системы для туристической фирмы.

Туризм является отраслью, которая оказывает значительное влияние на социально-экономическое развитие территорий. Данный вид народнохозяйственной деятельности характеризуется мультипликативным эффектом, генерируя образование доходов в смежных отраслях экономики. Среди видов туризма ключевую роль стал играть внутренний туризм как основной источник финансовых поступлений от туристской деятельности. Студент разработал сайт, который предоставляет информацию о туристической фирме. Есть большое количество компаний, которые занимаются различными турами. Тема туризма актуальна, но на данный момент внутренний туризм находится на этапе развития. Сайт разработан корректно, с использованием языка Python.

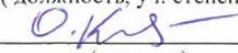
В дипломной работе рассматриваются инструменты для создания веб-сайта, такие как Python, фреймворк Django, текстовый редактор Atom, СУБД SQLite. Во второй главе ведется обоснование выбора программного обеспечения, в третьей главе ведется разработка сайта. Студент также загрузил свой сайт на хостинг.

В процессе написания дипломной работы Мухпул М.А. показала себя самостоятельным, хорошо подготовленным специалистом. Также она проявила высокий уровень знаний.

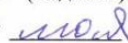
Рекомендую дипломную работу, выполненную Мухпул М.А., к защите. Считаю, что она заслуживает присвоения академической степени бакалавра по специальности 5B070300 – Информационные системы.

Научный руководитель

PhD, сеньор-лектор
(должность, уч. степень, звание)

 Киселева О.В.

(подпись)

«1»  2019 г.

Протокол анализа Отчета подобия

заведующего кафедрой / начальника структурного подразделения

Заведующий кафедрой / начальник структурного подразделения заявляет, что ознакомился(-ась) с Полным отчетом подобия, который был сгенерирован Системой выявления и предотвращения плагиата в отношении работы:

Автор: Мухпул М

Название: Разработка информационной системы туристической фирмы

Координатор: Ольга Киселева

Коэффициент подобия 1:0,5

Коэффициент подобия 2:0

Тревога:0

После анализа отчета подобия заведующий кафедрой / начальник структурного подразделения констатирует следующее:

- обнаруженные в работе заимствования являются добросовестными и не обладают признаками плагиата. В связи с чем, работа признается самостоятельной и допускается 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, работа не допускается к защите.

Обоснование:

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

Дата 16.05.19

Подпись заведующего кафедрой /

начальника структурного подразделения





Окончательное решение в отношении допуска к защите, включая обоснование:

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

Допускается к защите

.....
1'

Дата 06.05.2022

.....
Подпись заведующего кафедрой / 

начальника структурного подразделения



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: Мухпул М

Название: Разработка информационной системы туристической фирмы

Координатор: Ольга Киселева

Коэффициент подобия 1:0,5

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:

Взвешивается возможность
не обосновывается

15.05.19

Дата

О.К. —

Подпись Научного руководителя

Краткий отчет



| | |
|---|---|
| Университет: | Satbayev University |
| Название: | Разработка информационной системы туристической фирмы |
| Автор: | Мухпул М |
| Координатор: | Ольга Киселева |
| Дата отчета: | 2019-05-15 07:16:33 |
| Коэффициент подобия № 1: | 0,5% |
| Коэффициент подобия № 2: | 0,0% |
| Длина фразы для козффициента подобия № 2: | 25 |
| Количество слов: | 4 784 |
| Число знаков: | 37 220 |
| Адреса пропущенные при проверке: | |
| Количество завершенных проверок: | 65 |

>>

Самые длинные фрагменты, определенные, как подобные

| № | Название, имя автора или адрес гиперссылки (Название базы данных) | Автор | Количество одинаковых слов |
|---|---|-------|----------------------------------|
| 1 | URL_ http://vestiirk.ru/articles/ | | 6 |
| 2 | URL_ https://fullpicture.ru/obshhestvo/15-prichin-po-kotorym-nam-nuzhno-izbavitsya-ot-yadernogo-oruzhiya.html | | 6 |
| 3 | URL_ http://vestiirk.ru/articles/ | | 5 |
| 4 | URL_ https://www.sipri.org/sites/default/files/2018-05/milex_press_release_rus.pdf | | 5 |

>>

Документы,содержащие подобные фрагменты: Из домашней базы данных

Не обнаружено каких-либо заимствований

>>

Документы,содержащие подобные фрагменты: Из внешних баз данных

Не обнаружено каких-либо заимствований



Документы, содержащие подобные фрагменты: Из интернета

Документы, выделенные жирным шрифтом, содержат фрагменты потенциального плагиата, то есть превышающие лимит в длине коэффициента подобия № 2

| № | Источник гиперссылки | Количество одинаковых слов (количество фрагментов) |
|---|--|--|
| 1 | URL_ http://vestiirk.ru/articles/ | 11 (2) |
| 2 | URL_ https://fullpicture.ru/obshhestvo/15-prichin-po-kotorym-nam-nuzhno-izbavitsya-ot-yadernogo-oruzhiya.html | 6 (1) |
| 3 | URL_ https://www.sipri.org/sites/default/files/2018-05/milex_press_release_rus.pdf | 5 (1) |

АНДАТПА

Бұл дипломдық жұмыс бастапқы тіркеумен туристік компанияның ақпараттық жүйесін әзірлеуге арналған . Түрлі турлармен айналысатын көптеген компаниялар бар. Туризм тақырыбы өзекті, бірақ қазіргі уақытта ішкі туризм даму сатысында тұр.

Ұсынылған сайт Қазақстанның әдемі және ерекше жерлерімен айналысатын компанияның сайты болып табылады. Сайтты әзірлеу Python көмегімен жүзеге асырылды - жалпы мақсаттағы жоғары деңгейлі бағдарламалау тілі және Django фреймворк . Бұл дипломдық жұмыста келесі құралдар қолданылды: Python кітапханалары, Django, Atom, HTML.

АННОТАЦИЯ

Данная дипломная работа посвящена разработке информационной системы туристической компании с первичной регистрацией . Есть большое количество компаний, которые занимаются различными турами. Тема туризма актуальна, но на данный момент внутренний туризм находится на этапе развития.

Разработанный сайт является, сайтом компании которая занимается турами по красивым и необычным местам Казахстана. Разработка сайта осуществлена при помощи Python - высокоуровневый язык программирования общего назначения и фреймворка Django . В данной дипломной работе использованы следующие инструменты: библиотеки Python, Django, Atom, HTML.

ANNOTATION

This thesis is devoted to the development of the information system of the travel company with the primary registration . There are a large number of companies that are engaged in various tours.

The topic of tourism is relevant, but at the moment domestic tourism is at the stage of development.

The proposed site is the site of the company which is engaged in tours of the beautiful and unusual places of Kazakhstan. Development of the site was carried out using Python - a high-level programming language for General purpose and Django framework . In this thesis the following tools are used: Python libraries, Django, Atom, HTML .

СОДЕРЖАНИЕ

| | |
|---|----|
| ВВЕДЕНИЕ | 21 |
| 1 Обзор и анализ существующих информационных систем в туризме | 22 |
| 1.1 Характеристика особенностей туристических предприятий | 22 |
| 1.2 Анализ существенных недостатков туристических фирм | 23 |
| 1.3 Информация – связующее звено туристской отрасли | 26 |
| 1.4 Системы бронирования | 27 |
| 1.5 Постановка задачи | 30 |
| 2 Обоснование выбора программного обеспечения | 32 |
| 2.1 Требования к программе | 32 |
| 2.2 Система управления базами данных SQLite | 32 |
| 2.3 Язык программирования Python | 35 |
| 2.4 Django – фреймворк на Python | 37 |
| 2.5 Текстовый редактор Atom | 39 |
| 3 Разработка программного обеспечения | 41 |
| 3.1 Структура базы данных | 41 |
| 3.2 Архитектура MVC | 45 |
| 3.3 Системное программное обеспечение 28 | 48 |
| 3.3.1 Операционная система Microsoft Windows 7 | 48 |
| 3.4 Прикладное программное обеспечение | 49 |
| 3.4.1 Браузер Google Chrome | 49 |
| 3.5 Инструментальное программное обеспечение 28 | 51 |
| 3.5.1 Среда разработки PyCharm | 51 |
| 3.5.2 Используемые технические средства | 51 |
| ЗАКЛЮЧЕНИЕ | 52 |
| СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ | 53 |

ВВЕДЕНИЕ

Среди многообразия поисков путей развития рынка, средств производства, новых направлений деятельности коммерческо-посреднических организаций и предприятий вызывают значительный интерес научные исследования и практические новации. В течение последних лет бурно развиваются основанные на информатике новые технологии.

На протяжении всего периода применения компьютеров и компьютерных систем существует тенденция создания высоконадежных управляющих комплексов, ориентированных на получение и использование информационных ресурсов. Эта тенденция выразилась в мощном процессе создания различных видов систем, как встроенных в уникальные объекты информационно-технологических комплексов. В наше время информационные технологии стали неотъемлемой частью нашей жизни. Информационные системы, связанные с предоставлением и обработкой информации для всех уровней управления объектами, приобретают особую важность в общественной жизни. На данный момент невозможно представить какую-либо организацию, не применяющую социальные сети и сайты.

Важным этапом разработки информационной системы является углубленное изучение данной области. Выявление сущностей организации, какие функции они выполняют, какие атрибуты входят, в сущности. Определение зависимостей между сущностями. Это все создает визуальное представление исследуемой задачи.

Целью работы является разработка информационной системы турфирмы "Wanderlust". Начальным этапом создания системы является изучение, анализ и моделирование деятельности туризма для возможного улучшения и оптимизации методов работы.

В данной дипломной работе осуществляется проектирование ИС туристической фирмы "Wanderlust". ИС должна быть проста в использовании и предельно понятна для сотрудников. Клиент не может вносить изменения в структуру таблиц и схему данных.

ИС туристической фирмы должна обеспечивать выполнение следующих основных функций:

- информацию о городах, услугах, турах, транспорте, стран;
- вывод информации о стоимости туров;
- вывод информации о турах в определенные страны с определенной длительностью пребывания;
- формирование списка сотрудников, клиентов, туров.

1 Обзор и анализ существующих информационных систем в туризме

В данной дипломной работе необходимо разработать информационную систему туристической фирмы. Главной задачей при разработке данной информационной системы является быстрота нахождения информации, просмотра туров, услуги и т.д.

В качестве предметной области рассматривается туристическая фирма.

Туристическая фирма - независимая многофункциональная компания, работающая под собственной торговой маркой, что даёт возможность не ограничивать в выборе турпродукта какого-то одного туроператора.

В идеале, турагентство должно решать для туриста следующие задачи:

- помочь выбрать клиенту тур исходя из его запросов;
- предоставить правдивую и полную информацию по туру;
- правильно оформить все документы для клиента;
- оказывать поддержку своему клиенту на отдыхе, если он оказался в затруднительной ситуации.

1.1 Характеристика особенностей туристических предприятий

Основными задачами туристских предприятий являются организация и проведение путешествий, походов, экскурсий, привлечение к участию в проводимых мероприятиях максимального количества граждан, обеспечение познавательной, воспитательной, оздоровительной ценности мероприятий при высоком качестве обслуживания туристов, а также соблюдение требований, предъявляемых к хозяйственно-финансовой деятельности и безопасности людей.

Цели предприятия определяются в зависимости от выбранных видов туризма и предоставляемых услуг.

Туроператорская деятельность связана с формированием путешествий, экскурсий, их подготовкой и проведением; турагентская деятельность занимается реализацией путевок и подготовкой проездной документации; экскурсионная деятельность предполагает разработку и проведение экскурсий.

Основными целями, которые ставятся перед этими видами деятельности, являются:

- повышение духовного и интеллектуального уровня жителей Казахстана;

- привлечение широких слоев населения к активным занятиям физической культурой и спортом;
- укрепление и развитие культурных дружественных связей с зарубежными странами;
- привитие местному населению патриотизма, чувства уважения иностранными гражданами Республики Казахстан;
- формирование у туристов и экскурсантов бережного отношения к природе;
- привлечение к туризму малообеспеченных социальных групп населения.

Каждое предприятие, осуществляя хозяйственно-финансовую деятельность, выполняет свои обязанности и пользуется предоставленными ему правами. Являясь юридическим лицом, оно должно иметь свои расчетные счета в банковском учреждении, уставный фонд, самостоятельный баланс, печать с наименованием организации, лицензию на ведение туристской деятельности. При создании туристского предприятия работникам предстоит провести огромную организаторскую работу по заключению договоров с транспортными ведомствами, гостиницами, пунктами питания, музеями, театрами и выставочными залами, с внутренними и зарубежными партнерами, страховыми и рекламными компаниями, общественными и другими организациями. Взаимоотношения между предприятиями и физическими лицами, занимающимися оказанием туристско-экскурсионных услуг, строятся на взаимовыгодных и взаимоприемлемых условиях, что отражается в договорах с партнерами. Туризм тесно связан с размещением, питанием, транспортировкой, изданием рекламной и сувенирной продукции, изготовлением снаряжения и других туристских товаров. Создание транспортных агентств по продаже авиа-, авто- и железнодорожных билетов в последние годы стало солидной статьей доходов турфирм.

1.2 Анализ существенных недостатков туристических фирм

В настоящее время в Республике Казахстан насчитывается сотни туристских предприятий, но о высокой культуре обслуживания клиентов пока говорить рано. Обслуживание начинается задолго до того, как мы увидим своего клиента в лицо.

Создаваемая информационная система должна решить данные проблемы. Система управления туристической фирмой должна быть в состоянии централизовать данные в реальном времени, чтобы позволить загружать информации о турах для непосредственного рассмотрения в

реальном времени через целую компанию. Но мы создадим лишь часть из этой системы, облегчающую работу менеджерам по поиску информации.

Ни для кого не секрет, что одной из главных проблем отечественного туристического сектора является недостаточно развитая инфраструктура (гостиницы, дороги, транспорт). Эта проблема известна давно, и её обсуждали не раз на различных уровнях. Ситуация, конечно же, с каждым годом постепенно меняется к лучшему, но предстоит ещё сделать немало.

Другая проблема - дефицит квалифицированных кадров для туристической индустрии. Отечественный туризм остро нуждается в технологически грамотных молодых профессионалах, неординарно мыслящих, владеющих государственным и иностранными языками. Ограниченное использование современных технологий в сфере туризма также доставляет дискомфорт современным туристам. Кроме того, Казахстану не хватает достаточной узнаваемости для увеличения потока иностранных туристов. У выезжающих граждан популярным традиционно является озеро Иссык-Куль (Кыргызстан). Кроме Кыргызстана можно назвать Азербайджан, Узбекистан, Россию и, конечно же, Грузию. Курорты Грузии уже успели завоевать сердца наших соотечественников. В этом году увеличится поток казахстанцев в Россию, в первую очередь, в связи с проведением там ЧМ по футболу. Если говорить о дальнем зарубежье, то наиболее популярными направлениями являются Китай, Турция, ОАЭ, Тайланд, Малайзия, Индия. Очевидно, что обесценивание национальной валюты ещё даёт о себе знать, и казахстанцы выбирают более бюджетные направления. Те же, кто не уехал отдыхать за границу, находят варианты внутри Казахстана. Многие игроки рынка начали успешно продавать внутренний продукт. Наиболее популярными направлениями по-прежнему являются Алматы, Алаколь, Бухтарма, Боровое. На озере Алаколь в 2016-2017 гг. наблюдался настоящий бум. В пик сезона было трудно, а порой и невозможно забронировать места в отелях, расположенных на побережье озера. Предполагается, что и в этом году на Алаколе будет немало отдыхающих, поскольку растёт число туристических объектов и улучшается инфраструктур. Во время недавнего визита президента страны в Алматинскую область, он отметил, что Алаколь должен стать самым посещаемым местом в Казахстане.

Социальные сети во всем мире уже давно стали одним из инструментов увеличения и удержания клиентской базы во многих сферах предпринимательства. Казахстанский туристический бизнес также обратил внимание на это явление. Многие компании завели свои «аккаунты» и пытаются с их помощью привлечь дополнительный поток потенциальных клиентов. Многие туристические компании пытаются привлечь или удержать клиентов скидками и акциями, не обращая должного внимания на

то, что этого уже не достаточно. Мир изменился, каждый этап путешествия созрел для новизны, инноваций. Именно этого ждут «миллениалы», задающие сегодня тренд в туризме. Поэтому крайне важно предлагать свои услуги инновационными методами, используя самые современные технологии.

Технологии помогают турагентам быстрее и лучше выбирать информацию для клиента и о клиенте, обеспечивать потребителя данными и аналитикой, учитывать все его пожелания и быть ценным консультантом. - Очевидно, что рынок туризма Казахстана меняется, онлайн-туризм активно набирает обороты и в последующие годы будет расти. Этому есть логичные объяснения. Жёсткая конкуренция, рыночные тренды, связанные с техническим прогрессом и молодое поколение туристов, так называемые «миллениалы» являются движущей силой этих изменений. Кроме того, скандалы, возникающие в отечественной туристической индустрии, подрывают доверие к «офлайн» и заставляют часть путешественников переходить в «онлайн». Драйвером роста онлайн-туризма Казахстана сегодня является продажа авиа и ж/д билетов. Доля онлайн-покупок авиабилетов, по некоторым оценкам, в 2017 году составила чуть более 10%, в 2016 г. – около 10%, в 2015 г — около 8%, а в 2014 г. – не более 4% (для сравнения, в США уже не менее 70% билетов продается через Интернет, в Европе – не менее 50%). То есть рост очевиден.

В первую очередь мобильные технологии вывели туризм на новый уровень. В настоящее время с увеличением количества пользователей смартфонов и планшетов, растёт спрос и на мобильные тревел-приложения. В западных странах не менее 50% туристов планируют свои поездки посредством мобильных устройств. Современные путешественники предпочитают многофункциональные приложения, облегчающие им жизнь. Эти приложения становятся неотъемлемой частью процесса путешествий.

В наиболее развитых странах игроки туристического бизнеса активно начали использовать современные мобильные технологии, дающие возможность предлагать не только самые выгодные цены, но и персонализированный перечень услуг, зарабатывая тем самым доверие и лояльность клиентов, что критически важно в условиях, когда технологии меняют весь ландшафт туристической отрасли.

Персонализация сегодня стала одним из ключевых трендов в глобальной туристической сфере. Данные, которые можно будет получать с помощью современных технологий, помогут персонализировать поездку для путешественника. В свою очередь персонализация приводит к лояльности клиентов, которая сейчас важна как никогда. И одним из способов завоевания лояльности туристов является предоставление туристическими агентствами высококачественного комплексного сервиса.

1.3 Информация - связующее звено туристской отрасли

Международный и внутренний туризм представляет собой мощную отрасль по торговле услугами. Основу туристической индустрии составляют фирмы туроператоры и турагенты, занимающиеся туристическими поездками, продажей их в виде путевок и туров; предоставляющие услуги по размещению и питанию туристов (гостиницы, кемпинги и др.), их передвижению по стране, а также органы управления, информации, рекламы по исследованию туризма и подготовке для него кадров, предприятия по производству и продаже товаров туристского спроса. На туризм работают и другие отрасли, для которых обслуживание туристов не является основным видом деятельности (предприятия культуры, торговли и др.). Туризм - информационно насыщенная деятельность. Существует немного других отраслей, в которых сбор, обработка, применение и передача информации были бы настолько же важны для ежедневного функционирования, как в туристической индустрии. Услуга в туризме не может быть выставлена и рассмотрена в пункте продажи, как потребительские или производственные товары. Ее обычно покупают заранее и вдали от места потребления. Таким образом, туризм на рынке почти полностью зависит от изображений, описаний, средств коммуникаций и передачи информации.

Однако выделяется одна особенность - связующим центром, удерживающим различных производителей в рамках туристической отрасли, является информация. Именно информационные потоки, а не товары обеспечивают связи между производителями туристических услуг; они идут не только в виде потоков данных, но выступают также в форме услуг и платежей.

Услуги (например, ночевка в гостинице, аренда автомобиля, комплексные туры и места в самолетах) не пересылаются турагентам, которые, в свою очередь, не хранят их до тех пор, пока не продадут потребителям. Передается и используется информация о наличии, стоимости и качествах этих услуг. Точно так же реальные платежи не переводятся от турагентов турпоставщикам, а комиссионные - от турпоставщиков турагентам.

Можно выделить три характерные черты туризма. Во-первых, это - разнообразная и интегрированная торговля услугами. Во-вторых, это - комплексная услуга, как с точки зрения производителя, так и потребителя. Наконец, это - информационно насыщенная услуга. Поэтому туризм - как международный, так и внутренний, - сфера растущего применения информационных технологий.

Система информационных технологий, используемых в туризме, состоит из компьютерной системы резервирования, системы проведения телеконференций, видеосистем, компьютеров, информационных систем

управления, электронных информационных систем авиалиний, электронной пересылки денег, телефонных сетей, подвижных средств сообщения и т.д. При этом необходимо отметить, что эта система технологий разворачивается не тур агентами, гостиницами или авиакомпаниями каждым в отдельности, а всеми ими.

Более того, использование каждым сегментом туризма системы информационных технологий имеет значение для всех остальных частей. Например, системы внутреннего управления гостиницей могут быть связаны с компьютерными глобальными сетями, которые обеспечивают, в свою очередь, основу для связи с гостиничными системами резервирования, которые, уже в обратном направлении, могут быть доступны турагентам через их компьютеры. Следовательно, мы имеем дело с интегрированной системой информационных технологий, которая распространяется в туризме. Из вышеизложенного становится ясно, что в туристической индустрии распространяются не компьютеры, не телефоны, не видеотерминалы сами по себе - здесь функционирует система взаимосвязанных компьютерных и коммуникационных технологий.

Кроме того, отдельные компоненты туристической отрасли тесно взаимосвязаны друг с другом - ведь многие турпроизводители вовлечены в деятельность друг друга. Все это позволяет рассматривать туризм как высоко интегрированную услугу, что делает его еще более доступным для применения информационных технологий в организации и управлении.

1.4 Системы бронирования

Активно используют информационные технологии турагенты и туроператоры - без их компьютерных систем бронирования, видеосистем, систем взаимодействующих видеотекстов - невозможно представить ежедневное планирование и управление операциями. Компьютерные системы резервирования оказывают огромное влияние на всю туристическую отрасль. Около 90% турагентов в США и Великобритании связаны в компьютерные системы бронирования. Компьютерные системы бронирования предоставляют не только авиауслуги, но также ночевки в гостиницах, аренду автомобилей, круизные поездки, информацию о месте пребывания, курсы валют, сообщения о погодных условиях, автобусное и ж/д сообщение. Такие системы позволяют резервировать все основные сегменты тура - от мест в гостиницах и авиаперелетов до билетов в театр и страховых полисов. Фактически они составляют всеобщую информационную систему, предлагающую важнейшие распределительные сети для всей туристической торговли. Одним соединением через модем с серверами, имеющими соответствующую базу данных, турагенты получают доступ к информации о наличии возможных услуг, стоимости, качестве,

времени прибытия и отправления по разнообразному ряду туристических услуг от своих поставщиков. Более того, турагенты могут связаться с этими базами данных для того, чтобы сделать и подтвердить свой заказ. Функционирование и эффективность этих систем требуют, чтобы поставщики туристических услуг усвоили, по крайней мере, минимальный уровень технологии (например, навыки работы с персональными компьютерами и использования сетевых ресурсов в турагентствах), чтобы получать доступ к таким системам и быть на них представленными.

Крупнейшими компьютерными системами резервирования (бронирования) на международном рынке туризма являются системы AMADEUS, Worldspan и Galileo. Пример работы этой системы можно увидеть на рисунке 1.

Сейчас казахстанским агентствам предлагается 3 варианта подключения к Amadeus. Первый - телефонная версия Dial Up, не требующая дополнительного оборудования кроме компьютера и модема, подходит для малых агентств. Средние агентства больше устроит стандартная версия, устанавливаемая в офисе и включающая, помимо программного обеспечения, особые компьютеры и принтеры для печати билетов, программное обеспечение. Крупным агентствам, располагающим собственными локальными компьютерными сетями, устанавливается система клиент-сервер. Через шлюз, сервер локальной сети, агентства связываются с центральным сервером Amadeus. Таким образом, каждый терминал агентства имеет доступ к системе бронирования. Причем плата за подключение к Amadeus каждого последующего рабочего места весьма незначительна.

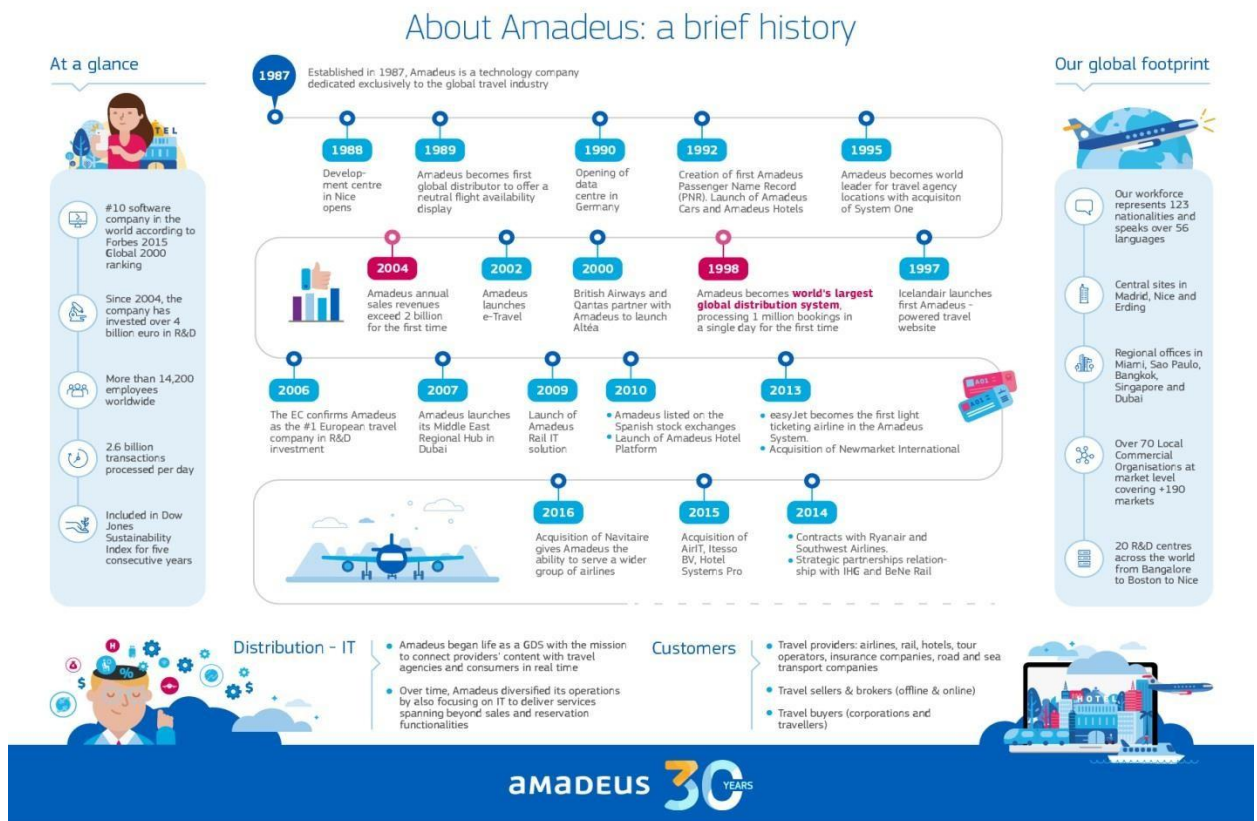


Рисунок 1 – Структура работы системы Amadeus

Worldspan считается самой динамично развивающейся компьютерной системой бронирования в мире. Вот уже почти три года, как эта система бронирования представлена и на нашем рынке .

Galileo - одна из самых передовых систем бронирования в техническом отношении. Именно Galileo одной из первых предоставила агентствам Windows-версию системы резервации. Программа Premier позволяет агентствам полностью автоматизировать работу по обслуживанию клиентов. Например, вести клиентские базы данных, создавать собственные экранные формы и меню, сохранять наиболее часто повторяющиеся запросы.

Хотя стоимость бронирования с помощью компьютера составляет несколько долларов, с учетом комиссии турагентам, представительских расходов, расходов по обслуживанию кредитных карточек и других видов расходов, итоговая сумма по бронированию может достигать 10-15 американских долларов. Такая стоимость бронирования вполне уместна в высокодоходной гостинице, но чрезмерно велика для небольших гостиничных предприятия. В результате возможность применения новой технологии бронирования для частных индивидуальных гостиниц ограничена скромными финансовыми ресурсами. Опасаясь монополии глобальных систем, небольшие гостиничные компании и гостиницы

вынуждены проектировать затраты на внедрение нового оборудования и подготовку персонала при выработке планов на будущее.

1.5 Постановка задачи

Основной задачей дипломной работы является разработка информационной системы туристической компании.

В данной дипломной работе используются последние технологии в сфере разработки веб-приложения, которые позволяют создать масштабируемое, безопасное и гибкое приложения с минимальными затратами.

Требования при разработке информационной системы(веб-приложения) туристической компании:

- приложение должно обладать удобным и понятным интерфейсом для пользователей;
- исходный код приложения должен комментироваться на этапе разработки для будущих изменений или дополнений;
- приложение должно предусматривать контроль вводимой информации;
- для удобства и контроля версии должен использоваться git;
- приложение должно включать хранение и обработку данных о турах и туристах;
- на веб-ресурсе должна быть информация о бронировании и записи на туры;
- в приложении должна присутствовать административная часть, где происходит контроль за всеми происходящими действиями в приложения;
- приложение должно сообщать пользователю о некорректных действиях при работе с сервисом, с выдачей сообщений об ошибках;
- приложение должно предоставлять полную информацию пользователю о туре, выбора маршрутов, стоимость и условия тура;
- приложение должно обладать удобным и понятным интерфейсом для пользователей;
- исходный код приложения должен комментироваться на этапе разработки для будущих изменений или дополнений;
- приложение должно предусматривать контроль вводимой информации;
- для удобства и контроля версии должен использоваться git;
- приложение должно включать хранение и обработку данных о данных клиента;
- на веб-ресурсе должна быть информация о компании;

- в приложении должна присутствовать административная часть, где происходит контроль за всеми происходящими действиями в приложения;
- приложение должно иметь окно регистрации, для того чтобы пользователь мог выбрать и добавить в корзину, подходящие ему туры.

2 Обоснование выбора программного обеспечения

2.1 Требования к программе

Исходя из специфики операций, к разрабатываемой системе должны быть предъявлены следующие требования:

- функциональные возможности ИС должны позволять выполнение в автоматизированном режиме основных операций, производимых менеджерами туристической компании;
 - базовое (системное) программное обеспечение должно позволять работу в реальном времени, и допускать проведения в больших количествах операций ввода/вывода, чтения, записи, вывод на печать; – базовые программные средства должны обеспечивать перенос пакета прикладных программ на новые аппаратные платформы, без каких-либо в них изменений;
 - прикладные программные средства должны обеспечить взаимодействие с существующей БД;
 - управление ИС должно быть простым и наглядным, а работа должна снижать количество допускаемых администратором ошибок;
- Реализация вышеперечисленных требований позволит создать простую и эффективную автоматизированную систему, которая существенно повысит производительность труда администраторов и улучшит качество обслуживания клиентов.

Функциональная модель данных, входящая в состав программы написана как веб- приложение с использованием языка программирования Python и фреймворка Django, также текстового редактора Atom. Информационная система состоит из двух составляющих: – управляющего модуля; – модулей экранных форм.

Управляющий модуль реализует пользовательский интерфейс, создает запросы к серверу и обрабатывает ответы от него.

Модуль экранных принимает запрос от клиента, производит обработку, после, формирует и отправляет страницу клиенту с использованием протокола HTTP.

2.2 Система управления базами данных SQLite

СУБД позволяет пользователям создавать, считывать, обновлять и удалять данные из баз данных. СУБД выступает в качестве интерфейса между базой данных и конечным пользователем или приложением, обеспечивая согласованную организацию данных и легкий доступ.

SQL (язык структурированных запросов) - это стандартизированный язык программирования для управления реляционными базами данных и выполнения различных задач обработки данных. Использование SQL включает изменение таблицы базы данных и структуры индекса; добавление, обновление и удаление строк данных; и извлечение подмножеств информации из базы данных для приложений обработки транзакций и аналитики. Запросы и другие операции SQL принимают форму команд, написанных как операторы. Обычно используемые операторы SQL включают выбор, добавление, вставку, обновление, удаление, создание, изменение и усечение. в SQL существует специальный набор общих команд — таких, как SELECT, INSERT, UPDATE и DELETE.

SQLite - СУБД с открытым исходным кодом. SQLite работает практически на всех платформах, включая Linux, UNIX и Windows. Его можно использовать со многими другими приложениями.

SQLite является важным элементом корпоративного стека с открытым исходным кодом, который называется LAMP. LAMP - это платформа вебразработки, которая использует операционную систему Linux и использует Apache в качестве веб-сервера, SQLite в качестве СУБД и в качестве объектно-ориентированного языка программирования, например PHP. (В

некоторых случаях используются Perl, Python, PHP.)

Основан на модели клиент-сервер. Ядром SQLite является сервер SQLite, который управляет всеми командами базы данных. Сервер спроектирован как каталог, который может быть собран из другой программы в отдельное приложение (клиент-сервер для использования в сетевой среде).

SQLite работает с несколькими утилитами, которые поддерживают управление базами данных SQLite. Команды устанавливаются на компьютер и отправляются в SQLite Server через клиента SQLite.

SQLite предназначен для быстрой обработки исходной большой базы данных. Обычно устанавливается только на одном компьютере, но можно обращаться к базе данных через различные клиентские интерфейсы SQLite, поэтому можно расположить базу данных в нескольких местах.

Основные особенности SQLite.

SQLite позволяет хранить и получать доступ к данным через несколько механизмов хранения, включая InnoDB, CSV и NDB. SQLite также может реплицировать данные и таблицы разделов для повышения производительности и долговечности. Пользователям SQLite не нужно изучать новые команды. Доступ к данным можно получить с помощью стандартных команд SQL. Логическая архитектура базы данных SQLite приведена на рисунке 2.

SQLite имеет формат C и C ++ и может использоваться на более чем 20 платформах, включая Mac, Windows, Linux и Unix. СУБД поддерживает большое количество баз данных с миллионами записей и поддерживает множество типов данных, включая целые числа со знаком, без знака, 1, 2, 3, 4 и 8 байтов. FLOAT; Double; CHAR; VARCHAR; VARBINARY; STRING; BLOB; DATE. Подсчет и OpenGIS пространственных типов. Он также поддерживает строковые типы фиксированной и переменной длины.

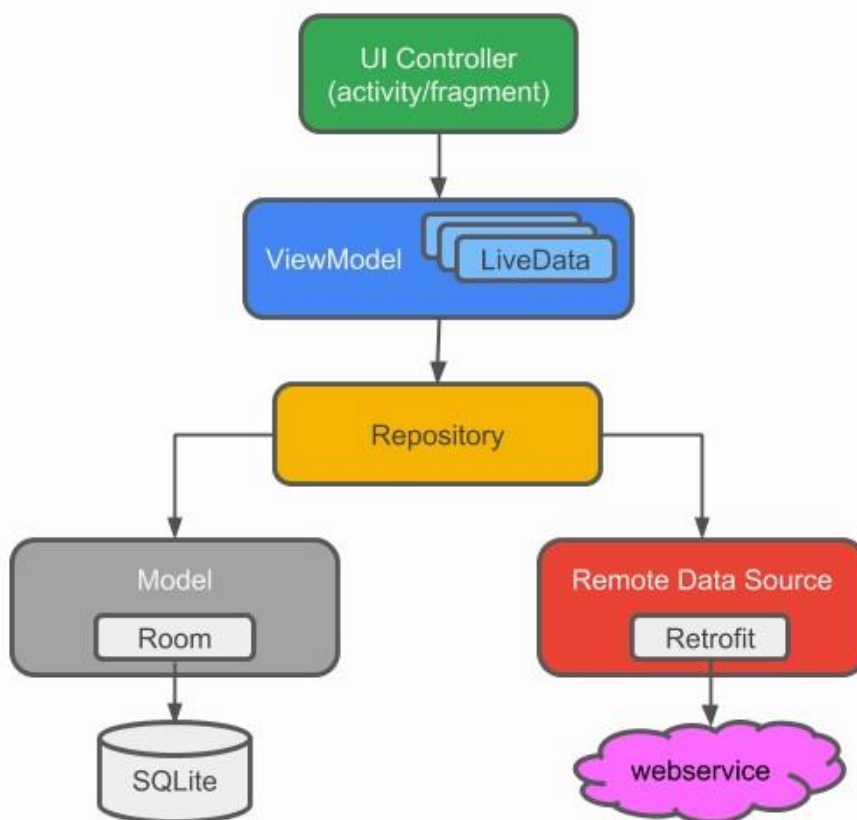


Рисунок 2 – Архитектура базы данных SQLite

Из соображений безопасности SQLite использует зашифрованный пароль и систему доступа, которая обеспечивает контроль на основе хоста;

Клиент SQLite может использовать несколько протоколов для сервера SQLite, включая соединители TCP / IP для всех платформ. SQLite также поддерживает различные клиентские и служебные программы, программы командной строки и инструменты администрирования, такие как SQLite Workbench.

Кроме того, SQLite позволяет программам использовать несколько механизмов хранения для каждой таблицы, позволяя им выбирать наиболее подходящий механизм хранения для указанных таблиц. Одним из двигателей SQLite является InnoDB. InnoDB разработан для высокой доступности. Из-за этого он не так быстр, как другие двигатели. SQL использует свое собственное пространство хранения, но поддерживает ряд

мер для предотвращения потери данных. Он может работать в кластере для обеспечения высокой доступности обеих систем.

2.3 Язык программирования Python

Python – это один из наиболее популярных современных языков программирования. Он пригоден для решения разнообразных задач и предлагает те же возможности, что и другие языки программирования: динамичность, поддержку ООП и кросс-платформенность. Разработку Python начал Гвидо Ван Россум (Guido Van Rossum) еще в середине 1990-х годов, поэтому к настоящему времени удалось избавиться от стандартных «детских» болезней, существенно развить лучшие стороны языка и привлечь множество программистов, использующих Python для реализации своих проектов.

Если же программист только начинает свой путь в области разработки ПО, то Python станет идеальным «вводным» языком программирования. Благодаря своей лаконичности он позволит быстрее овладеть синтаксисом языка, а отсутствие «наследства» в виде сформировавшихся на протяжении многих лет аксиом поможет быстро освоить ООП. В силу этих факторов «кривая обучения» Python будет довольно короткой, и программист сможет перейти от учебных примеров к коммерческим проектам.

Архитектура Python. Любой язык, неважно – для программирования или общения, состоит как минимум из двух частей – словаря и синтаксиса. Язык Python организован точно так же, предоставляя синтаксис для формирования выражений, образующих исполняемые программы, и словарь – набор функциональности в виде стандартной библиотеки и подключаемых модулей.

Как уже упоминалось, синтаксис Python достаточно лаконичный, особенно если сравнивать с Java или C++. С одной стороны – это хорошо, так как чем проще синтаксис, тем проще его изучать и тем меньше ошибок можно совершить в процессе его использования. Однако у подобных языков есть недостаток – с их помощью можно передавать самую простую информацию и нельзя выражать сложные конструкции.

К Python это не относится, так как это язык простой, но упрощенный. Дело в том, что Python является языком с более высоким уровнем абстракции, выше, например, чем у Java и C++, и позволяет передать такое же количество информации в меньшем объеме исходного кода.

Также Python является языком общего назначения, поэтому может применяться практически в любой области разработки ПО (standalone,

клиент-сервер, Web-приложения) и в любой предметной области. Кроме того, Python легко интегрируется с уже существующими компонентами, что позволяет внедрять Python в уже написанные приложения.

Другая составляющая успеха Python – это его модули расширения, как стандартные, так и специфические. Стандартные модули расширения Python – это отлично спроектированная и неоднократно проверенная функциональность для решения задач, возникающих в каждом проекте по разработке ПО, обработка строк и текстов, взаимодействие с операционной системой, поддержка Web-приложений. Эти модули также написаны на языке Python, поэтому обладают его важнейшим свойством – кроссплатформенностью, позволяющей безболезненно и быстро переносить проекты с одной операционной системы на другую.

Если необходимой функциональности не оказалось в стандартной библиотеке Python, то можно создать собственный модуль расширения для его последующего неоднократного использования. Здесь стоит отметить, что модули расширения для Python можно создавать не только на самом языке Python, но и с помощью других языков программирования. В этом случае появляется возможность более эффективной реализации ресурсоемких задач, например сложных научных вычислений, однако теряется преимущество кросс-платформенности, если язык модуля расширения не является сам по себе кросс-платформенным, как Python.

Как известно, все кросс-платформенные языки программирования построены по одной модели: это действительно переносимый исходный код и среда исполнения (runtime environment), которая не является переносимой и специфична для каждой конкретной платформы. В эту среду исполнения обычно входит интерпретатор, который исполняет исходный код, и различные утилиты, необходимые для сопровождения приложения – отладчик, обратный ассемблер и т.д.

В среду исполнения Java дополнительно входит компилятор, так как исходный код необходимо скомпилировать в байт-код для виртуальной Джамашины. В среду исполнения Python входит только интерпретатор, который одновременно является и компилятором, однако компилирует исходный код Python непосредственно в машинный код целевой платформы.

На данный момент существуют три известных реализации среды исполнения для Python: CPython, Jython и Python.NET. Как можно догадаться из названия, первая среда реализована на языке C, вторая на языке Java, а последняя – на платформе .NET.

Среда исполнения CPython обычно называется просто Python, и когда говорят о Python, то чаще всего имеется в виду именно эта реализация. Эта реализация состоит из интерпретатора и модулей расширения, написанных на языке C, и может использоваться на любой платформе, для которой доступен стандартный компилятор C. Кроме того, существуют уже

скомпилированные версии среды исполнения для различных операционных систем, включая различные версии ОС Windows и различные дистрибутивы Linux. В этой и последующих статьях будет рассматриваться именно CPython, если иное не оговаривается отдельно.

Среда исполнения Jython – это реализация Python для работы с виртуальной Java-машиной (JVM). Поддерживается любая версия JVM, начиная с версии 1.2.2 (текущая версия Java – 1.6). Для работы с Jython требуется установленная Java-машина (среда исполнения Java) и определенное знание языка программирования Java. Уметь писать исходный код на языке Java не обязательно, однако придется иметь дело с JAR-файлами и Java-апплетами, а также документацией в формате JavaDOC.

Какую версию среды выбрать – зависит исключительно от предпочтений программиста, вообще же рекомендуется держать на компьютере и CPython, и Jython, так как они не конфликтуют между собой, а взаимно дополняют друг друга. Среда CPython работает быстрее, так как нет промежуточного уровня в виде JVM; кроме того, обновленные версии Python сначала выпускают именно в виде среды CPython. Однако Jython может использовать любой класс Java в качестве модуля расширения и работать на любой платформе, для которой существует реализация JVM. Преимущества Python:

- скорость выполнения программ написанных на Python очень высока. Это связано с тем, что основные библиотеки Python написаны на C++ и выполнение задач занимает меньше времени, чем на других языках высокого уровня;
- в связи с этим вы можете писать свои собственные модули для Python на C или C++;
- в стандартных библиотеках Python вы можете найти средства для работы с электронной почтой, протоколами Интернета, FTP, HTTP, базами данных;
- скрипты, написанные при помощи Python выполняются на большинстве современных ОС. Такая переносимость обеспечивает Python применение в самых различных областях;
- Python подходит для любых решений в области программирования, будь то офисные программы, вэб-приложения, GUI-приложения ; – над разработкой Python трудились тысячи энтузиастов со всего мира.

2.4 Django — фреймворк на Python

Django — это программный каркас с богатыми возможностями, подходящий для разработки сложных сайтов и веб-приложений, написанный на языке программирования Python.

Django проектировался для работы под управлением Apache (с модулем `mod_python`) и с использованием PostgreSQL в качестве базы данных. В настоящее время, помимо PostgreSQL, Django может работать с другими

СУБД: SQLite (MariaDB), SQLite, Microsoft SQL Server, DB2, Firebird, SQL Anywhere и Oracle. Для работы с базой данных Django использует собственный ORM, в котором модель данных описывается классами Python, и по ней генерируется схема базы данных.

Архитектура Django похожа на «Модель-Представление-Контроллер» (MVC). Контроллер классической модели MVC примерно соответствует уровню, который в Django называется Представление (View), а презентационная логика Представления реализуется в Django уровнем Шаблонов (Templates). Из-за этого уровневую архитектуру Django часто называют «Модель-Шаблон-Представление» (MTV).

Первоначально разработка Django велась для обеспечения более удобной работы с новостными ресурсами, что достаточно сильно отразилось на архитектуре: фреймворк предоставляет ряд средств, которые помогают в быстрой разработке веб-сайтов информационного характера. Например, разработчику не требуется создавать контроллеры и страницы для административной части сайта, в Django есть встроенное приложение для управления содержимым, которое можно включить в любой сайт, сделанный на Django, и которое может управлять сразу несколькими сайтами на одном сервере. Административное приложение позволяет создавать, изменять и удалять любые объекты наполнения сайта, протоколируя все совершённые действия, и предоставляет интерфейс для управления пользователями и группами (с пообъектным назначением прав).

Веб-фреймворк Django используется в таких крупных и известных сайтах, как Instagram, Disqus, Mozilla, The Washington Times, Pinterest, lamoda и др.

Некоторые возможности Django:

- ORM, API доступа к БД с поддержкой транзакций;
- встроенный интерфейс администратора, с уже имеющимися переводами на многие языки;
- диспетчер URL на основе регулярных выражений;
- расширяемая система шаблонов с тегами и наследованием;
- система кеширования;
- интернационализация;
- подключаемая архитектура приложений, которые можно устанавливать на любые Django-сайты;
- «generic views» — шаблоны функций контроллеров;

- авторизация и аутентификация, подключение внешних модулей аутентификации: LDAP, OpenID и прочие;
- система фильтров («middleware») для построения дополнительных обработчиков запросов, как например включённые в дистрибутив фильтры для кеширования, сжатия, нормализации URL и поддержки анонимных сессий;
- библиотека для работы с формами (наследование, построение форм по существующей модели БД);
- встроенная автоматическая документация по тегам шаблонов и моделям данных, доступная через административное приложение.

Некоторые компоненты фреймворка между собой связаны слабо, поэтому их можно достаточно просто заменять на аналогичные. Но с некоторыми (например, с ORM) это сделать не очень просто. Кроме возможностей, встроенных в ядро фреймворка, существуют пакеты, расширяющие его возможности.

На базе Django разработано достаточно много готовых решений, распространяемых под свободной лицензией, среди которых системы для управления интернет-магазинами, универсальные системы управления содержанием, а также более узконаправленные проекты.

2.5 Текстовый редактор Atom

Atom (в прошлом Atomicity) — бесплатный текстовый редактор с открытым исходным кодом для Linux, macOS, Windows с поддержкой плагинов, написанных на Node.js, и встраиваемых под управлением Git. Большинство плагинов имеют статус свободного программного обеспечения, разрабатываются и поддерживаются сообществом.

Atom основан на Electron (ранее известный как Atom Shell) — фреймворке кросс-платформенной разработки с использованием Chromium и io.js. Редактор написан на CoffeeScript и LESS.

Версия 1.0 была выпущена 25 июня 2015 г.

Поддерживаются следующие языки (со стандартными плагинами) в версии v1.5.1. C/C++, C#, Clojure, CSS, CoffeeScript, Markdown (GitHub Flavored), Go, Git, HTML, JavaScript, Java, JSON, Julia, Less, Make, Mustache, Objective-C, PHP, Perl, Property List (Apple), Python, Ruby on Rails, Ruby, Sass, Shell script, Scala, SQL, TOML, XML, YAML.

Язык html - это основа web сайтов, с его помощью создается каркас страницы, которую вы видите в браузере. Заголовки, абзацы и другие блоки можно выделить и на странице сайта. Это делается с помощью html тегов.

Html теги - это специальные команды для браузера. Они говорят ему, что, к примеру, следует считать заголовком страницы, а что абзацем. Теги строятся по такому принципу: <имя тега>. Имя тега может состоять из английских букв и цифр. Примеры тегов: <h1>, <p>, .

Теги обычно пишутся парами - открывающий тег и соответствующий ему закрывающий. Разница между открывающим и закрывающим тегами в том, что в закрывающем теге после уголка < стоит слеш /.

К примеру, <p> - так я открыл тег p, а так - </p> - я его закрыл. Все, что попадает между открывающим и закрывающим тегами, подпадает под воздействие нашего тега.

Бывают теги, которые не нужно закрывать, например,
 или . В тегах также могут размещаться атрибуты - специальные команды, которые расширяют действие тега. Атрибуты размещаются внутри открывающего тега в таком формате: <тег атрибут1="значение" атрибут2="значение">.

Язык CSS расширяет возможности языка HTML. Он позволяет менять цвета, шрифты, фон, в общем заниматься красотой сайта. А html, соответственно, отвечает за структуру сайта.

Страница сайта - это обычный текстовый файл с расширением .html. Внутри этого файла и хранится текст html страницы вместе с тегами. Этот файл обязательно должен иметь следующие теги: тег <html>, который должен содержать в себе текст всего сайта, а внутри него должны быть еще два тега: тег <head> для служебного содержимого страницы и тег <body> - для основного текста, который и виден на экране браузера.

3 Разработка программного обеспечения

3.1 Структура базы данных

База данных разработана в СУБД SQLite, в ней имеются 5 таблиц. Система управления базами данных основан на модели клиент-сервер. Ядром SQLite является сервер SQLite, который управляет всеми командами базы данных. Сервер спроектирован как каталог, который может быть собран из другой программы в отдельное приложение. Ниже представлены разработанные структуры таблиц базы данных.

- в таблице 1 – данные о клиентах;
- в таблице 2 – данные о путевках;
- в таблице 3 – курорты;
- в таблице 4 – номера;

– в таблице 5 – услуги. SQLite не использует парадигму клиент-сервер, то есть движок SQLite не является отдельно работающим процессом, с которым взаимодействует программа, а представляет собой библиотеку, с которой программа компонуется, и движок становится составной частью программы. Таким образом, в качестве протокола обмена используются вызовы функций (API) библиотеки SQLite. Такой подход уменьшает накладные расходы, время отклика и упрощает программу. SQLite хранит всю базу данных (включая определения, таблицы, индексы и данные) в единственном стандартном файле на том компьютере, на котором исполняется программа. Простота реализации достигается за счёт того, что перед началом исполнения транзакции записи весь файл, хранящий базу данных, блокируется; ACID-функции достигаются в том числе за счёт создания файла журнала.

Несколько процессов или потоков могут одновременно без каких-либо проблем читать данные из одной базы. Запись в базу можно осуществить только в том случае, если никаких других запросов в данный момент не обслуживается; в противном случае попытка записи оканчивается неудачей, и в программу возвращается код ошибки. Другим вариантом развития событий является автоматическое повторение попыток записи в течение заданного интервала времени.

В комплекте поставки идёт также функциональная клиентская часть в виде исполняемого файла `sqlite3`, с помощью которого демонстрируется реализация функций основной библиотеки. Клиентская часть является кроссплатформенной утилитой командной строки.

Благодаря архитектуре движка возможно использовать SQLite как на встраиваемых системах, так и на выделенных машинах с гигабайтными массивами данных.

SQLite поддерживает динамическое типизирование данных. Возможные типы значений: INTEGER, REAL, TEXT и BLOB. Так же поддерживается специальное значение NULL.

Размеры значений типа TEXT и BLOB не лимитированы ничем, кроме константы SQLITE_MAX_LENGTH в исходниках sqlite, равной миллиарду (10⁹).

Каждое значение в любом поле любой записи может быть любого из этих типов, независимо от типа, указанного при объявлении полей таблицы. Указанный при объявлении поля тип хранится для справки в его исходном написании, и используется в качестве основы для выбора предпочтений (так называемое «type affinity»: это подход, редко встречающийся в других СУБД) при выполнении неявных преобразований типов на основании схожести этого названия типа на что-либо, знакомое SQLite. В этот алгоритм зашит обширный перечень практикуемых в других СУБД вариантов названий типов данных.

Если безопасного преобразования записываемого значения в предпочитаемый тип не получается, SQLite записывает значение в его исходном виде. Для получения значений из базы есть ряд функций для каждого из типов, и если тип хранимого значения не соответствует запрашиваемому, оно тоже, по возможности, преобразуется.

Старые версии SQLite были спроектированы без каких-либо ограничений, единственным условием было то, чтобы база данных умещалась в памяти, в которой все вычисления производились при помощи 32-разрядных целых чисел. Это создавало определённые проблемы. Из-за того, что верхние пределы не были определены и соответственно должным образом протестированы, часто обнаруживались ошибки при использовании SQLite в достаточно экстремальных условиях. Поэтому в новых версиях SQLite были введены пределы, которые теперь проверяются вместе с общим набором тестов.

Таблица 1 - Атрибуты сущности "Клиенты"

| Имя атрибута | Описание |
|----------------------------|--|
| Код клиента | Первичный ключ, однозначно идентифицирующий клиента туристической фирмы. |
| Фамилия Имя Отчество | Фамилия Имя Отчество клиента. |
| Серия и номер паспорта | Документ (в данном случае паспорт), его серия и номер. |

| | |
|--------------------------|--|
| Дата оформления договора | Дата, когда клиент оформил путевку. |
| Код путевки | Внешний ключ к сущности "Путевки". Код путевки, которая оформлена на клиента. |
| Комментарии | Характеристика клиента по своевременной оплате путевки или услуг. Возможна и другая информация. Например, особенности по здоровью или его пожелания. |

Таблица 2 - Атрибуты сущности "Путевки"

| Имя атрибута | Описание |
|-------------------------|--|
| Код путевки | Первичный ключ, идентифицирующий путевку. |
| Код курорта | Внешний ключ к сущности "Курорты". Код курорта, на который оформлена путевка. |
| Код номера | Внешний ключ к сущности "Номера". Код номера, забронирован по путевке. |
| Код услуги | Внешний ключ к сущности "Услуги". Код набора услуг, которые предоставляются по этой путевке. |
| Продолжительность | Продолжительность пребывания на курорте. |
| Дата начала | Дата, с которой путевка вступает в силу. |
| Стоимость путевки всего | Стоимость путевки, включая стоимость услуг, стоимость билетов. |

Таблица 3 - Атрибуты сущности "Курорты"

| Имя атрибута | Описание |
|----------------|---|
| Код курорта | Первичный ключ, идентифицирующий курорт. |
| Название | Полное название курорта. |
| Расположение | Вблизи, какого города расположен курорт. |
| Страна | Страна, в которой находится этот курорт. |
| Характеристика | Описание климата, погодных условий, пляжей. Наличие и количество бассейнов, сравнительная характеристика цен, а также другие преимущества и возможные недостатки. |

Таблица 4 - Атрибуты сущности "Номера"

| Имя атрибута | Описание |
|--------------|---|
| Код номера | Первичный ключ, идентифицирующий номер. |

| | |
|-------------------|---|
| Количество мест | Двухместные или одноместные номера. |
| Тип номера | Номера президентские, люкс, обычные, эконом класса. |
| Стоимость в сутки | Стоимость проживания в номере за сутки. |
| Адрес гостиницы | Полный адрес гостиницы. |
| Характеристика | Описание номера. Наличие бытовых приборов, бара, джакузи или душ, ортопедический матрас, вид из окна, солнечная или теневая сторона |

Таблица 5 - Атрибуты сущности "Услуги"

| | |
|----------------|--|
| Имя атрибута | Описание |
| Код услуги | Первичный ключ, идентифицирующий набор услуг. |
| Загранпаспорт | Если агентство оформляет данный документ, то в этом поле проставляется его цена. |
| Виза | В это поле вписывается страна, в которую необходимо оформить визу, если таковой нет. |
| Стоимость визы | Стоимость визы, если ее оформляет агентство. |

Таблица 7 - Состав базы данных информационной системы.

| № | Сущности концептуальной модели | Таблицы физической модели | |
|---|--------------------------------|---------------------------|--|
| | | Название | Информация |
| | "Клиенты" | Klient | Сведения о клиентах фирмы. |
| | "Путевки" | putevka | Сведения о предлагаемых путевках фирмы. |
| | "Курорты" | kurort | Сведения о курортах. |
| | "Номера" | nomer | Номера, которые можно забронировать. |
| | "Услуги" | uslugi | Набор услуг, которые предлагает агентство. |

Первым шагом при создании логической модели БД является построение диаграммы ER, которая состоит из трех частей: сущностей, атрибутов и взаимосвязей. диаграмма позволяет рассмотреть систему

целиком и выяснить требования, необходимые для ее разработки, касающиеся хранения информации. ER-диаграмма графически представляет структуру данных проектируемой информационной системы. Сущности отображаются при помощи прямоугольников, содержащих имя. Имена принято выражать существительными в единственном числе, взаимосвязи - при помощи линий, соединяющих отдельные сущности. Взаимосвязь показывает, что данные одной сущности ссылаются или связаны с данными другой.

При создании концептуальной модели были определены сущности и атрибуты сущностей. Из этих данных составлена логическая модель базы данных, которая представлена на рисунке 4.

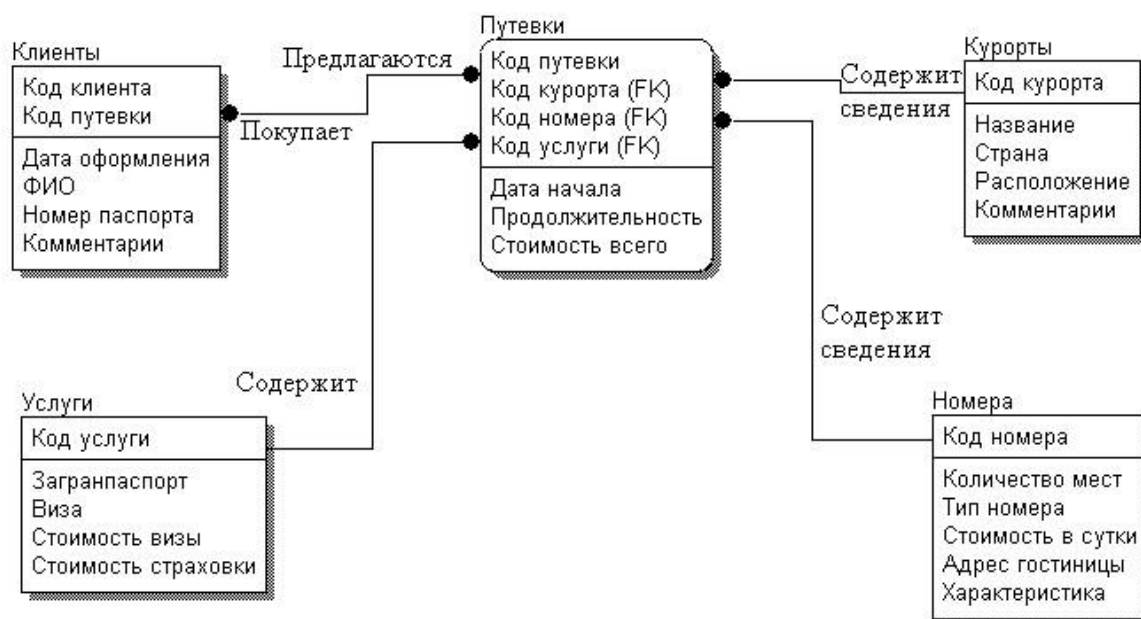


Рисунок 4 – Логическая модель базы данных турфирмы

3.2 Архитектура MVC

В качестве архитектуры для разработки данного приложения используется архитектура MVC (Model-View-Controller, «МодельПредставление-Контроллер»). MVC разделяет данные приложения, пользовательский интерфейс и логику управления на три независимых компонента: модель, представление, контроллер.

Модель – предоставляет данные и представляет бизнес-логику. Объекты модели получают и хранят состояние модели базы данных, также реагирует на команды контроллера, изменяя своё состояние.

Представление - является пользовательским интерфейсом. Отвечает за отображение данных модели пользователю, реагируя на изменения модели.

Контроллер - обрабатывает запросы пользователей. Обычно

пользователь взаимодействует с интерфейсом, и, в свою очередь, происходит запрос URL. Запрос обрабатывается контроллером. Контроллер отображает данные модели и отображает view в качестве ответа.

MVC позволяет упростить большой код, путем разделения его на части. Написание длинного кода сбивает с толку и его трудно изменить без ошибок.

MVC не зависит от определенного языка программирования и не требует объектно-ориентированного метода или других примеров.

Разделение здесь не означает, что нужны три файла (или три папки которые содержат файлы) с названиями model, view и controller. На практике модель занимает основной объем приложения, и представлена в виде большого числа разнотипных классов - сущностей, сервисов, классов работы с БД, и для каждого вида классов делают отдельные папки.

MVC может применяться ко многим различным типам приложений, включая серверные веб-приложения и настольные (клиентские) приложения. Разница между ними заключается в том, что программа веб-приложения обрабатывается в ответ на запрос пользователя, отображает результат (как правило, веб-страницу) и завершает работу. Когда поступают другие запросы, выполняется новая независимая копия программы для обработки этого. В отличие от веб-приложений, персональных компьютеров, мобильных приложений (и приложений, использующих страницы браузера, созданные на JavaScript), они имеют продолжительный срок службы. Они обрабатывают различные запросы пользователя и обновляют информацию на экране без выхода.

Взаимодействие между компонентами MVC реализовано немного по-разному в серверных и десктопных приложениях, веб-приложение обрабатывает один запрос пользователя и завершается, а десктопное приложение обрабатывает много запросов без перезапуска.

Серверные приложения обычно используют «пассивную» модель, а десктопные приложения – «активную» модель. В отличие от активных моделей, пассивные модели позволяют подписываться и получать уведомления об изменениях. Серверные приложения в этом не нуждаются.

В схеме с активной моделью Вид подписывается на изменения в Модели. Затем, когда происходит событие (например, пользователь нажимает кнопку), вызывается контроллер. Он предоставляет команды изменения данных для модели. Модель сообщает своим подписчикам (в том числе Виду), что данные изменились, и Вид обновляет интерфейс программы.

Схемы с активными моделями рассчитаны на продолжительную работу. Напротив, схемы пассивной модели обычно используются в короткоживущих приложениях.

Серверные приложения используют пассивные модельные системы. Предположим, что пользователь заходит на страницу форума. В своем браузере он отправляет HTTP-запрос на поиск страницы, содержащей список сообщений. В это время контроллер запускается, анализирует запрос пользователя и запрашивает список сообщений у Model. Получив его, он вызывает View, отправляет список и отображает его в виде веб-страницы. Чтобы добавить сообщение, пользователь заполняет форму, далее вызывается контроллер, отвечающий за обработку данных этой формы, который примет данные, и задействует Модель, чтобы проверить и вставить в базу данных новое сообщение, и затем вернет HTTP ответ с редиректом на страницу просмотра сообщений.

На рисунке 5 приведена схема, изображающая взаимодействие компонентов.

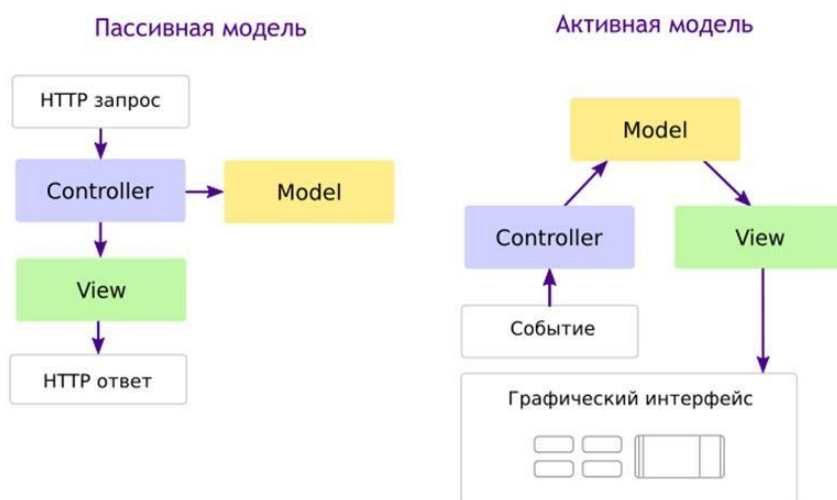


Рисунок 5 – Взаимодействие компонентов MVC

Архитектура Django похожа на «Модель-Представление-Контроллер» (MVC). Контроллер классической модели MVC примерно соответствует уровню, который в Django называется Представление (View), а презентационная логика Представления реализуется в Django уровнем Шаблонов (Templates). Из-за этого уровневую архитектуру Django часто называют «Модель-Шаблон-Представление» (MTV).

Первоначально разработка Django велась для обеспечения более удобной работы с новостными ресурсами, что достаточно сильно отразилось на архитектуре: фреймворк предоставляет ряд средств, которые помогают в быстрой разработке веб-сайтов информационного характера. Например, разработчику не требуется создавать контроллеры и страницы для административной части сайта, в Django есть встроенное приложение для управления содержимым, которое можно включить в любой сайт, сделанный на Django, и которое может управлять сразу несколькими

сайтами на одном сервере. Административное приложение позволяет создавать, изменять и удалять любые объекты наполнения сайта, протоколируя все совершённые действия, и предоставляет интерфейс для управления пользователями и группами (с пообъектным назначением прав).

Веб-фреймворк Django используется в таких крупных и известных сайтах, как Instagram, Disqus, Mozilla, The Washington Times, Pinterest, lamoda и др.

3.3 Системное программное обеспечение

3.3.1 Операционная система Microsoft Windows 7

Windows 7 является операционной системой для продуктов Microsoft. Существует несколько версий Windows. Windows 7 является самой популярной операционной системой в мире с 2017 года.

Эта операционная система совместима с Unicode 5.1. Теги теперь знают больше языков. Операционная система поддерживает несколько сенсорных операций. Все версии операционной системы содержат 50 новых источников. Доступные шрифты оптимизированы для правильного отображения всех символов. Windows 7 - это первая версия Windows с добавленными латинскими и нелатинскими символами. Панель управления источником была улучшена по умолчанию. В системе установлен только источник, показанный на рисунке.

Еще одна особенность Windows 7 - интеграция с производителями панелей. Большинство устройств регистрируются автоматически, но 90% совместимы с драйверами Windows Vista.

Windows 7 может улучшить совместимость со старыми программами, но может быть недоступна в Windows Vista. Windows XP В режиме Windows XP вы можете использовать Windows XP для запуска старых программ, которые поддерживают практически любое существующее приложение на виртуальной машине.

Во-первых, новая версия Embedded DirectX версии 11 добавляет следующие улучшения: Новая подсветка компьютера поддерживает многопоточный рендеринг мозаики, новый алгоритм сжатия текстур.

Сетевые технологии могут использовать выбор файлов путем временного хранения кэшированного трафика. Если пользователю локальной сети необходимо передавать файлы по сети, он не может загружать каналы из локального кэша и использовать каналы с ограниченной пропускной способностью. Эта технология предназначена для использования в больших сетях и больших операционных системах.

Вы можете создать все конфигурации АИК и образы конфигурации

Windows. Поэтому в Windows 7 используются спецификации USB 3.0, Bluetooth 4.0 и DirectX 11.1. Поддерживает .NET Framework 4.5.

Windows 7 также получила поддержку UEFI OEM и образов продуктов UEFI для Windows 7 SP1. При необходимости установщик UEFI для Windows 7 может выполнить запись на USB-устройство. Вы можете добавить полную поддержку UEFI, используя Windows AIK.

В отличие от Windows Vista, Windows 7 предлагает более гибкую конфигурацию управления пользователями (UAC) с двумя аргументами. Если вы попытаетесь сменить компьютер, отобразится только местоположение по умолчанию. Уведомление о смене компьютера (рабочий стол выглядит темным)

BitLocker был добавлен для шифрования съемных носителей с использованием технологии шифрования BitLocker, но TPM не поддерживается, поскольку съемный носитель зашифрован.

Эти события также влияют на брандмауэр Windows. Пользователь вернул предупреждение, чтобы разрешить доступ к сети.

Групповая политика и блокировка приложений могут блокировать выполнение определенных программ.

Лицензионная версия может блокировать удаленные подключения к компьютерам, которые требуют аутентификации пользователя. Кроме того, адрес подключенного хоста заблокирован для связи и управления авторизацией между консолью и компьютером.

С DirectAccess вы можете установить безопасное соединение с сервером в фоновом режиме, а не с VPN, если вам нужно взаимодействовать с пользователем. DirectAccess может применять групповую политику, прежде чем пользователи войдут в систему.

3.4 Прикладное программное обеспечение

3.4.1 Браузер Google Chrome

Google Chrome - это браузер, разработанный Google на основе бесплатного браузера и движка Blink. Первая версия была выпущена 11 декабря 2008 года. По данным StatCounter, Chrome - самый популярный браузер в мире, которым пользуются около 300 миллионов пользователей Интернета.

Secure Chrome регулярно получает обновления (сайты, содержащие вредоносный код) и предупреждает, когда пользователь пытается посетить вредоносную страницу.

Плагины (наиболее часто используемые Adobe Flash Player) не являются частью браузера и не могут быть включены в песочницу. Таким

образом, безопасное выполнение кода страницы отключается плагинами, операционная система которых не имеет дополнительной защиты.

30 марта 2010 года Adobe Flash включается в браузер и не требует от пользователей его загрузки и установки. Flash Player обновляется автоматически.

Adobe Flash Player интегрирован с версии 5.0.

Начиная с версии 4.2, Chrome прекратил обслуживание плагина NPAPI. Этот плагин поддерживает новый формат PPAPI (Pepper API), который специально разработан для запуска плагина в отдельном процессе (песочнице). Таким образом, эта версия уязвимости в самом плагине не может проникнуть вредоносный код в систему.

Браузер поддерживает секретный режим. Страницы, видимые в окне «Инкогнито», не отображаются в истории браузера или в истории поиска и, как файлы cookie, не оставляют других следов на вашем компьютере. Пользователи автоматически удаляются после закрытия этого окна. Однако все загруженные файлы или созданные закладки будут сохранены.

Как предупреждает браузер весной 2015 года, переход в «секретный режим» не влияет на работу других пользователей, серверов или программ и не препятствует следующему:

- интернет-сайты, которые собирают информацию или сообщают ее другим;
- интернет-провайдеры и сотрудники, которые отслеживают доступ к веб-сайтам;
- вредоносное программное обеспечение, которое отслеживает нажатия клавиш.

Speed Chrome использует высокопроизводительный обработчик JavaScript V8. Кроме того, Chrome может использовать функцию предварительной выборки DNS для ускорения загрузки страницы.

Надежность. Chrome использует архитектуру многих процессов, и часто каждая карта или плагин содержит отдельный процесс. Эта процедура называется изоляцией, которая исключает вмешательство в закладки.

Chrome сможет выполнять задачи в соответствии с потребностями диспетчера задач, который может просматривать сайты и плагины, использующие пользовательскую память, процессоры и онлайн-каналы.

Адресная строка омнибокса вверху каждой карточки. Это комбинация адресной строки и строки поиска. Если адрес омнибокса не соответствует написанию URL-адреса, он перенаправляет поисковый запрос в поисковую систему, например, если в начале пропущено слово в адресе адреса нет места. Автозаполнение также помогает искать закладки и историю ранее посещенных страниц. Браузер может вызывать разные поисковые системы прямо из адресной строки.

Приложение запускается через один из браузеров (Internet Explorer, Opera, Firefox, Google Chrome) по адресу <http://localhost/8080>.

3.5 Инструментальное программное обеспечение

3.5.1 Среда разработки PyCharm

PyCharm — интегрированная среда разработки для языка программирования Python. Предоставляет средства для анализа кода, графический отладчик, инструмент для запуска юнит-тестов и поддерживает веб-разработку на Django. PyCharm разработана компанией JetBrains на основе IntelliJ IDEA. PyCharm работает под операционными системами Windows, Mac OS X и Linux. Основные возможности:

- статический анализ кода, подсветка синтаксиса и ошибок.
- навигация по проекту и исходному коду: отображение файловой структуры проекта, быстрый переход между файлами, классами, методами и использованиями методов.
- рефакторинг: переименование, извлечение метода, введение переменной, введение константы, подъем и спуск метода и т. д.
- инструменты для веб-разработки с использованием фреймворка Django
- встроенный отладчик для Python
- встроенные инструменты для юнит-тестирования
- разработка с использованием Google App Engine
- поддержка систем контроля версий: общий пользовательский интерфейс для Mercurial, Git, Subversion, Perforce и CVS с поддержкой списков изменений и слияния

3.5.2 Используемые технические средства

При разработке ПО использовался персональный компьютер со следующими характеристиками:

- процессор Intel® Core™ i3-5005U CPU @ 2.00 GHz;
- оперативная память 4,00 ГБ DDR3L 1600 МГц;
- жесткий диск HDD 500 ГБ;
- оптический привод встроенный DVD-RW; поддержка Double Layer;
- видеокарта NVIDIA GeForce 920m.

ЗАКЛЮЧЕНИЕ

Прогресс, достигнутый за последние несколько лет во всех аспектах вычислительной техники, включая теорию, технологию и приложения, привели к значительному расширению области применения компьютеров и росту числа их пользователей.

Увеличение использования интернета среди компаний и привело к широкому распространению веб-приложений. Веб-приложения дают возможность оптимизировать свои операции, повысить эффективность и сократить расходы. Также у них есть дополнительное преимущество: они работают на разных платформах, имеют более широкий охват и легко доступны из любой точки мира. Успешность туристической компании, в современном мире, зависит от грамотно реализованной информационной системы.

В ходе выполнения дипломного проекта в соответствии с заданием была разработана информационная система туристической фирмы. Была разработана структура базы данных в СУБД SQLite.

Работа была выполнена с использованием новых и актуальных технологий разработки веб-приложений, как Django, Atom. Разработанная программа устойчиво выполняет все свои функции.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- 1 Абдулаев, В.И. Прикладная информатика в экономике. Подготовка выпускной квалификационной работы : учебное пособие / В.И. Абдулаев, О.Е. Иванов, Е.А. Смиренский. - 2010. - 101 с.
- 2 Вендров, А.М. Практикум по проектированию программного обеспечения информационных систем : учебное пособие по специальностям "Прикладная информатика " / А.М. Вендров. - М.: 2006. - 191 с.
- 3 Годин, В.В. Информационное обеспечение управленческой деятельности : учебник / В.В. Годин, И.К. Корнеев. - М.: Мастерство, Высшая школа, 2011. - 240 с.
- 4 Маклаков, С.В. Моделирование бизнес-процессов : учебник / С.В. Маклаков. - М.: Диалог-МИФИ, 2012. - 224 с.
- 5 Прохоренок Н.А. Python 3 и PyQt. Разработка приложений. – СПб.: БХВ-Петербург, 2012. – 704 с.
- 6 Титоренко, Г.А. Автоматизированные информационные технологии в экоИТИ. 2010. - 159 с.
- 7 Агафонов, В.Н. Логическое программирование / В.Н. Агафонов. - М.: [не указано], 1998. - **427** с.
- 8 Арсак, Ж. Программирование игр и головоломок / Ж. Арсак. - М.: [не указано], 2014. - **501** с.
- 9 Кнут, Д.Э. Искусство программирования (Том 1. Основные алгоритмы): моногр. / Д.Э. Кнут. - М.: [не указано], 2000. - **700** с.
- 10 Кнут, Д.Э. Искусство программирования (Том 2. Получисленные алгоритмы) / Д.Э. Кнут. - М.: [не указано], 2000. - **225** с.
- 11 Кнут, Д.Э. Искусство программирования (том 3) / Д.Э. Кнут. - М.: [не указано], 1978
- 12 Роберт Мартин Чистый код. Создание, анализ и рефакторинг. – СПб.: Питер, 2015– 464 с.
- 13 Дейт, К. Дж. Д27 Введение в системы баз данных, 8-е издание.: Пер. с англ. — М.: Издательский дом "Вильямс", 2015. — 1328 с.: ил. — Парал. тит. англ.
- 14 Пьюривал С. П96 Основы разработки веб-приложений. – СПб.: Питер, 2015. – 272 с.: ил.
- 15 Кузнецов М. В., Симдянов И. В. К89 Самоучитель MySQL 5. — СПб.: БХВ-Петербург, 2006. — 560 с.: ил. ISBN 978-5-94157-754-5.

- 16 Адам Фримен Ш74 Pro ASP.NET MVC 5. — СПб.: Питер, 2017. – 537 с.:
- 17 Хеник Б. ПЗ8 HTML и CSS: путь к совершенству. - СПб.: Питер, 2011. – 336 с.: ил.

Приложение А

Листинг программы

```
{% load static %}
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <meta http-equiv="Expires" content="30">

  <!--Favicon-->
  <link rel="shortcut icon" href="static/images/favicon.ico" type="image/x-
  icon"> <link rel="icon" href="static/images/favicon.ico" type="image/x-icon">

  <!--Resetting the document--> <link rel="stylesheet"
  href="https://cdnjs.cloudflare.com/ajax/libs/normalize/7.0.0/normalize.min.css">

  <!--Fonts from Google-->
  <link
  href="https://fonts.googleapis.com/css?family=Alegreya+Sans:400,500,700"
  rel="stylesheet">

  <!--Custom Stylesheet-->
  <link rel="stylesheet" href="static/css/main.min.css">

  <!--Responsive Stylesheet-->
  <link rel="stylesheet" href="static/css/responsive.min.css">

  <!--Owl Carousel Stylesheet-->
  <link rel="stylesheet"
  href="https://cdnjs.cloudflare.com/ajax/libs/OwlCarousel2/2.2.1/assets/owl.carousel.min.css">
  <link rel="stylesheet"
  href="https://cdnjs.cloudflare.com/ajax/libs/OwlCarousel2/2.2.1/assets/owl.theme
  e.
```



```
default.min.css">
```

```
<!--Slick Carousel CSS CDN--> <link rel="stylesheet" type="text/css"
href="https://cdnjs.cloudflare.com/ajax/libs/slick-carousel/1.8.1/slick.min.css"/>
<link rel="stylesheet" type="text/css"
href="https://cdnjs.cloudflare.com/ajax/libs/slick-
carousel/1.8.1/slicktheme.min.css"/>
```

```
<title>Туризм</title>
</head>
<body>
```

```
<!--Youtube Video-->
<div class="youtube_video">
  <iframe id="youtube_player" width="840" height="473"
src="https://www.youtube.com/watch?v=2D-ssLZXMMY" frameborder="0"
allowfullscreen></iframe>
</div>
```

```
<!--Navigation Bar-->
<header id="top_header" class="clearfix">
  <div class="wrapper">
    
    <nav id="main_nav">
      <a href="#home_container" class="nav_active" >Домашняя страница</a>
      <a href="#about_us_container">О нас</a>
      <a href="#destinations_container">Направления</a>
      <a href="#contact_us_container" id="contact_us">Контакты</a>
      <a href="https://www.facebook.com/" target="_blank" id="fb"></a>
      <a href="https://www.instagram.com/" target="_blank" id="insta"></a>
      <a href="https://www.linkedin.com/" target="_blank" id="linkedin"></a>
    </nav>
    <button id="mobile_menu">Меню</button>
  </div>
</header>
```

```
<!--Home Section-->
<div class="home_container" id="home_container">
  <div class="wrapper">
    <div class="left_section">
```

```
<div class="outer">
  <div class="inner">
    <h1>Тур по Казахстану</h1>
    <p>
```

Наши турпакеты позаботятся обо всех деталях, поэтому вы сможете сосредоточиться на том, чтобы окунуться в невероятный мир вокруг вас.</p>

```
    <button class="hero_button" href="#destinations_container">
      Возьми меня к месту назначения
    </button>
    <button class="gallery_button" href="#">
      Посмотреть Галерею
    </button>
  </div>
</div>
</div>
<div class="right_section">
  <div class="outer">
    <div class="inner">
      
      <button href="#" class="play_video_button"></button>
    </div>
  </div>
</div>
<div class="clearfix"></div>
<div class="scroll_icon"></div>
</div>
</div>
```

```
<!--About us Section-->
<div class="about_us_container" id="about_us_container">
  <div class="wrapper">
    <div class="left_section">
      <div class="outer">
        <div class="inner">
          
        </div>
      </div>
    </div>
  </div>
```

```
<div class="right_section">
  <div class="outer">
    <div class="inner">
      <h1>О нас</h1>
      <p>Туры в Казахстан - самую большую страну Центральной Азии.
Современный Казахстан манит самыми разнообразными экскурсиями и
программами: знаменитый высокогорный каток Медео, первый в мире
космодром Байконур, грандиозный каньон Чарын - собрат американского
Гранд каньона, уникальные суфийские памятники, древние города Тараз,
Шымкент и Туркестан. Войлочные юрты, целебный кумыс, бескрайние
степи
- Казахстан запомнится вам яркой и своеобразной культурой
кочевников.</p>      <p>Путешествие потрясающее. Он объединяет
людей из разных уголков мира, расширяет кругозор, строит новые
отношения и питает уже существующие. Путешествие открывает глаза на
культуру, причуды и традиции региона, предлагая новые перспективы и
позволяя оценить некоторые вещи дома, которые они могут принять как
должное.</p>      <p>Гарантированные туры в Казахстан с
фиксированными датами разрабатываются с учетом рекомендаций и
предложений туристов, побывавших в туре. Таким образом постоянно
совершенствуются не только программа тура, но и качество его
организации. Путешествие по Казахстану - доступно и удобно, даже если в
вашей стране нет организованных туров в Казахстан. Вы можете заказать
тур, отправив нам заявку по Интернету, либо другим способом,
воспользовавшись нашей контактной информацией на сайте</p>
    </div>
  </div>
</div>
</div>
```

```
<!--Destination Section-->
<div class="destinations_container" id="destinations_container">
  <div class="wrapper">
    <h1>Места</h1>
  </div>
  <div class="owl-carousel">
    <div class="item">
      <center>
        
        <h2>Природа</h2>
        <p>
```

Пусть откроется чудо природы, от горных вершин, пронизывающих облака, до крошечных радужных рыб, плетущихся между коралловыми рифами</p>

</center>

</div>

<div class="item">

<center>

<h2>Духовные места</h2>

<p>Тур по духовным местам Казахстана</p>

</center>

</div>

<div class="item">

<center>

<h2>Экскурсия в каньон Чарын</h2>

<p>Экскурсия в каньон Чарын - один из однодневных туров в Казахстане. Оказавшись в Алматы проездом или на выходных, вы можете заказать экскурсию к каньону Чарын - младшему брату Гранд Каньона в США.</p>

</center>

</div>

<div class="item">

<center>

<h2>Экскурсия по городу Алматы и урочищу Медео</h2>

<p>Кафедральный Собор Зенкова, Мемориальный парк 28 героевпанфиловцев, гора Кок-Тюбе, главная мечеть, Площадь Независимости, ущелье Медео</p>

</center>

</div>

</div>

</div>

<!--Why Choose Us Section -->

<div class="raanvara_different_container">

<div class="wrapper">

<h1>Внутренний туризм отличается</h1>

<div class="different_section">


```

    <h2>100% индивидуальные пакеты</h2>
    <p>Никаких предварительно упакованных туров! Мы разрабатываем
наши турпродукты исключительно для ВАС!</p>
  </div>
  <div class="different_section">
    
  <h2>
Непревзойденное обслуживание клиентов</h2>
    <p>
Чувствуйте себя комфортно и безопасно, зная, что о тебе будут тщательно
заботиться</p>
  </div>
  <div class="different_section">
    
    <h2>Эксклюзивные компоненты тура</h2>
    <p>Уникальные туристические туристические и автобусные
туристические рынки, не похожие ни на один другой туристический
оператор. </p>
  </div>
  <div class="different_section">
    
    <h2>Местные гиды</h2>
    <p>Искренне заботливые неординарные личности, наши знающие гиды
</p>
  </div>
</div>
</div>
</div>

<!--Testimonials Section-->
<div class="testimonials_container">
  <h1>
Наши клиенты любят нас</h1>
  <div class="testimonial_carousel">
    <div class="testimonial">
      <p>Это был наш первый опыт и путешествие по Казахстану и это
было потрясающе. Мы наслаждались много. Сервис был превосходным.
Они помогли нам всеми возможными способами. Еще раз спасибо за то,
что сделали наш тур потрясающим и запоминающимся.</p>
      <h4>Сымбат студентка </h4>
    </div>
    <div class="testimonial">
      <p>

```

Прежде всего, большое спасибо за безупречное планирование, организацию и выполнение всей поездки. Группа ценит энтузиазм и гостеприимство</p>

<h4>Мадияр студент</h4>

</div>

<div class="testimonial">

<p></p>

<h4></h4>

</div>

</div>

</div>

<!--Contact Us section-->

<div class="contact_us_container" id="контакты">

<div class="wrapper">

<div id="google_map"></div>

<form id="contact_form" action="mail.php" method="POST">

<h1>Контакты</h1>

<input type="text" name="ФИО" placeholder="фио" id="ФИО">

<input type="text" name="lastname" placeholder="ЛОГИН" id="Логин">

<input type="email" name="почта" placeholder="ПОЧТА" id="почта">

<input type="tel" name="номер телефона" placeholder="НОМЕР
ТЕЛЕФОНА" id="Телефон">

<input type="number" name="ин" placeholder="НОМЕР
ДОКУМЕНТА" min="10" max="60" id="Номер">

<input type="text" name="тип" placeholder="ТУР" id="dтип">

<input type="date" name="когда хотите начать" id="дата старта">

<input type="date" name="резерв" id="резерв">

<textarea rows="8" cols="50" placeholder="сООБЩЕНИЕ
id="сообщение"></textarea>

<input type="submit" value="ОТПРАВИТЬ" id="отправить">

<p id="статус"></p>

```

    </form>
  </div>
</div>

<!--jQuery CDN-->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

<!--Owl Carousel Script File-->
<script
src="https://cdnjs.cloudflare.com/ajax/libs/OwlCarousel2/2.2.1/owl.carousel.min
.js "></script>

<!--Slick Carousel JS CDN-->
<script                                     type="text/javascript"
src="https://cdnjs.cloudflare.com/ajax/libs/slickcarousel/1.8.1/slick.min.js"></sc
ript>

<!--Master Script File-->
<script src="static/js/master.min.js"></script>

<!--Google Maps--> <script async defer
src="https://goo.gl/maps/iUHnW3wHs4kE6YyA8">
</script>

<script>
function initMap() {
  var chembur = {lat: 19.061908, lng: 72.9032800};      var markerCenter =
{lat: 19.0608151, lng: 72.9033282};
  var map = new google.maps.Map(document.getElementById("google_map"),
{
  zoom:16,      center: chembur,
  mapTypeId: 'satellite'
});
  var marker = new google.maps.Marker({      position: markerCenter,
  icon: "static/images/map_marker.png",
});

  marker.setMap(map);

  google.maps.event.addListener(marker, 'click', function(){      var
infoWindow = new google.maps.InfoWindow({

```

```

        content: "город Алматы"
    });
    infoWindow.open(map, marker);
    });
}
</script>
{% block content %}
{% endblock content %}
'9'
</body> </html>
from django.contrib import admin
from django.apps import AppConfig

```

```

class MoiSaitConfig(AppConfig):
    name = 'moi_sait'
    from django.db import models
    from django.urls import reverse

```

```

class Turizm(models.Model):
    title = models.CharField(max_length=200)
    author = models.ForeignKey(
        'auth.User',
        on_delete=models.CASCADE,
    )
    body = models.TextField()

```

Продолжение приложения А

```

    def __str__(self):
        return self.title
from django.test import TestCase
from django.urls import path
from .views import BlogListView

urlpatterns = [
    path("", BlogListView.as_view(), name='index'),
]
from django.views.generic import ListView

from django.urls import reverse_lazy

from .models import Turizm

class BlogListView(ListView):
    model = Turizm
    template_name = 'index.html'
    import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))

```



```

# Quick-start development settings - unsuitable for production
# See https://docs.djangoproject.com/en/2.2/howto/deployment/checklist/

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'c6b9#xw9174pn^gr3#rz5@##5m!-h-g9_i46hchnv(&lo+o%ds'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    Продолжение приложения A
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'moi_sait.apps.MoiSaitConfig',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'sait.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',

```

```

'DIRS': [os.path.join(BASE_DIR, 'templates')],
'APP_DIRS': True,
'OPTIONS': {
    'context_processors': [
        'django.template.context_processors.debug',
        'django.template.context_processors.request',
        'django.contrib.auth.context_processors.auth',
        'django.contrib.messages.context_processors.messages',
    ],
},
},
]

```

```
WSGI_APPLICATION = 'sait.wsgi.application'
```

```
# Database
```

```
# https://docs.djangoproject.com/en/2.2/ref/settings/#databases
```

```
DATABASES = {
```

```

    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME':
os.path.join(BASE_DIR, 'db.sqlite3'),
    }
}

```

Продолжение приложения

```
# Password validation
```

```
# https://docs.djangoproject.com/en/2.2/ref/settings/#auth-password-validators
```

```
AUTH_PASSWORD_VALIDATORS = [
```

```

    {
        'NAME':
'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME':

```

```
'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
    {
        'NAME':
'django.contrib.auth.password_validation.NumericPasswordValidator',
    },
]
```

```
# Internationalization
```

```
# https://docs.djangoproject.com/en/2.2/topics/i18n/
```

```
LANGUAGE_CODE = 'en-us'
```

```
TIME_ZONE = 'UTC'
```

```
USE_I18N = True
```

```
USE_L10N = True
```

```
USE_TZ = True
```

```
# Static files (CSS, JavaScript, Images)
```

```
# https://docs.djangoproject.com/en/2.2/howto/static-files/
```

```
Продолжение приложения A STATIC_URL =
```

```
'/static/'
```

```
STATICFILES_DIRS = [os.path.join(BASE_DIR, 'static')] from django.contrib
```

```
import admin
```

```
from django.urls import path, include
```

```
urlpatterns = [ path('admin/', admin.site.urls), path("", include('moi_sait.urls')),
```

```
]
```

```
import os
```

```
from django.core.wsgi import get_wsgi_application
```

```
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'sait.settings')
```

```
application = get_wsgi_application() import os
```

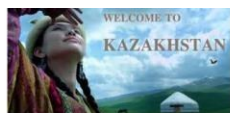
```
import sys

def main():
    os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'sait.settings')
    try:
        from django.core.management import execute_from_command_line
    except ImportError as exc:
        raise ImportError(
            "Couldn't import Django. Are you sure it's installed and "
            "available on your PYTHONPATH environment variable? Did you "
            "forget to activate a virtual environment?"
        ) from exc
    execute_from_command_line(sys.argv)

if __name__ == '__main__':
    main()
```

Приложение Б

Обзор сайта



Тур по Казахстану


Наши турагенты позаботятся обо всех деталях, поэтому вы сможете сосредоточиться на том, чтобы окунуться в невероятный мир вокруг вас.

ВЗЬМИ МЕНЯ К МЕСТУ НАЗНАЧЕНИЯ

ПОСМОТРЕТЬ ГАЛЛЕРЕЮ



Рисунок Б.1 – Главная страница сайта



О нас

Туры в Казахстан - самую большую страну Центральной Азии. Современный Казахстан манит самыми разнообразными экскурсиями и программами: знаменитый высокогорный коток Медео, первый в мире космодром Байконур, грандиозный каньон Чарын - собрат американского Гранд каньона, уникальные суфийские памятники, древние города Тараз, Шымкент и Туркестан. Войлочные юрты, целебный кумыс, бескрайние степи - Казахстан запомнится вам яркой и своеобразной культурой кочевников.

Путешествие потрясающее. Он объединяет людей из разных уголков мира, расширяет кругозор, строит новые отношения и питает уже существующие. Путешествие открывает глаза на культуру, причуды и традиции региона, предлагая новые перспективы и позволяя оценить некоторые вещи дома, которые они могут принять как должное.

Гарантированные туры в Казахстан с фиксированными датами разрабатываются с учетом рекомендаций и предложений туристов, побывавших в туре. Таким образом постоянно совершенствуются не только программа тура, но и качество его организации. Путешествие по Казахстану - доступно и удобно, даже если в вашей стране нет организованных туров в Казахстан. Вы можете заказать тур, отправив нам заявку по Интернету, либо другим способом, воспользовавшись нашей контактной информацией на сайте

Рисунок Б.2 – Информация по турам Казахстана





| | | | |
|--|---|--|--|
|  |  |  |  |
| Природа Пусть откроется чудо природы, от горных вершин, пронизывающих облака, до крошечных радужных рыб, плывущих между коралловыми рифами | Духовные места Тур по духовным местам Казахстана | Экскурсия в каньон Чарын Экскурсия в каньон Чарын - один из однодневных туров в Казахстане. Оказавшись в Алматы проездом или на выходных, вы можете заказать экскурсию к каньону Чарын - младшему брату Гранд Каньона в США. | Экскурсия по городу Алматы и урочищу Медео Кафедральный Собор Зенкова, Мемориальный парк 28 героев-панфиловцев, гора Кок-Тобе, главная мечеть, Площадь Независимости, ущелье Медео |

Рисунок Б.3 – Виды туров

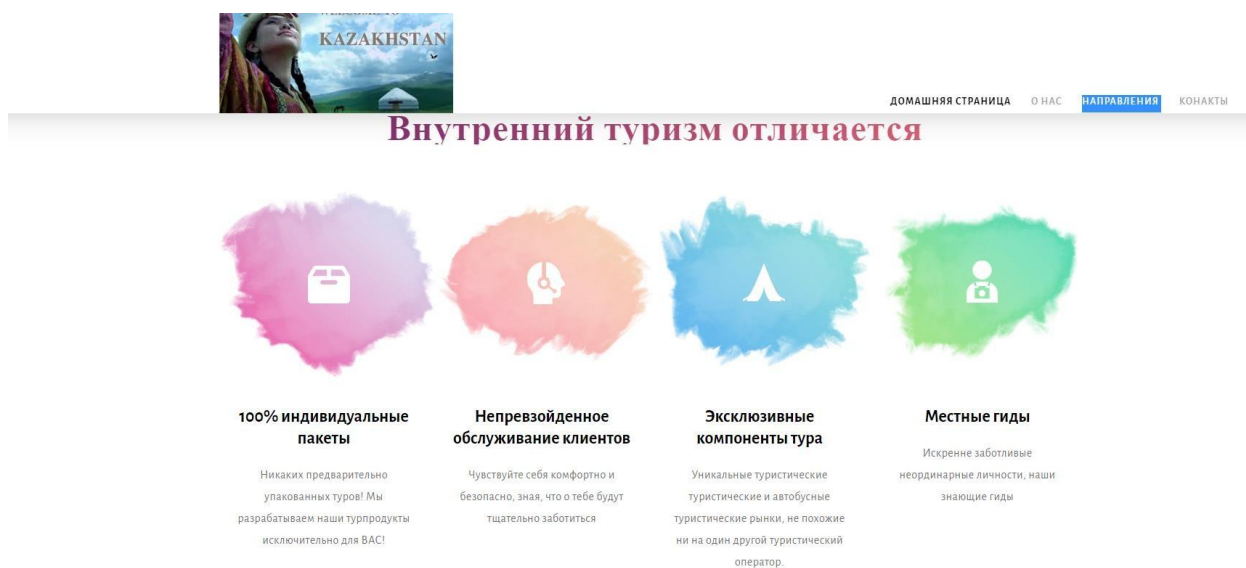


Рисунок Б.4 – Особенности услуг турфирмы