Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Дисциплина «Проектирование программного обеспечения»

Техническое задание

Интерфейс бронирования номеров

Студент: Жук С.С.

ФИТ 3 курс 2 группа

Преподаватель: Курилец А.В.

# **Введение**

В современном мире, с развитием технологий и мобильности, создание приложения для бронирования номеров в отеле становится более актуальным и востребованным. Данный проект предоставляет возможность пользователям легко находить и бронировать жилье по всему миру.

Целью данного проекта является создание удобной и интуитивно понятной платформы для бронирования номеров в отелях. Для достижения этой цели необходимо решить следующие задачи: разработать удобный для пользователей интерфейс, обеспечивающий простую навигацию и простоту использования, предоставить пользователям функционал для бронирования отелей, а администратору – функционал для управления пользователями и номерами.

Актуальность данного проекта обусловлена ростом туристической индустрии и увеличением числа людей, предпочитающих бронировать жилье в онлайн-режиме. Это также содействует сокращению времени и усилий, необходимых для поиска и бронирования жилья, что делает проект неотъемлемой частью современного рынка гостиничных услуг.

Целевой аудиторией приложения являются путешественники всех возрастов, ищущие удобный способ бронирования жилья в разных уголках мира. Также это приложение может быть полезно бизнес-путешественникам и тем, кто планирует длительные командировки.

# **Основания для разработки**

Разработка интерфейса бронирования номеров осуществляется на основании технического задания на создание системы бронирования номеров гостиниц, утвержденного организацией ООО «SearchHotel» 1 сентября 2024 года. В соответствии с этим документом, тема разработки обозначена как «Интерфейс для бронирования номеров» с условным наименованием проекта "HotelBooking". Техническое задание определяет требования к функционалу интерфейса, который должен обеспечить пользователям удобный доступ к системе бронирования номеров с возможностью выбора даты, типа номера, количества гостей.

# **Назначение разработки**

Назначение разработки программы заключается в предоставлении удобного и надежного инструмента для поиска и бронирования отелей, что обеспечивает комфортное и безопасное путешествие для клиентов.

## **3.1. Функциональное назначение**

Назначение программы "Интерфейс бронирования номеров" заключается в предоставлении пользователям удобного и эффективного инструмента для выбора и бронирования номеров в различных отелях. Основной функциональный контекст программы включает:

Поиск отелей и номеров по различным параметрам (местоположение, цена, категория и др.).

Отображение подробной информации о каждом отеле, включая фотографии и описание.

Возможность выбора даты заезда и выезда, а также числа гостей.

Бронирование номеров с сохранением данных о бронировании.

## **3.2. Эксплуатационное назначение**

Программа "Интерфейс бронирования номеров" предназначена для использования потребителями услуг гостеприимства, которые желают бронировать номера в отелях и гостиницах на различные даты и в разных местоположениях. Эксплуатационное назначение программы включает:

Обеспечение пользователей возможностью просмотра, выбора и бронирования отелей и номеров.

Обеспечение удобного интерфейса для навигации и выполнения бронирования.

Гарантирование безопасности и конфиденциальности данных пользователей.

Предоставление пользователю подробной информации о выбранном отеле и номере.

Автоматизацию процесса бронирования и уведомление пользователей о статусе брони.

# **Требования к программному изделию**

# **Требования к функциональным характеристикам**

В интерфейсе бронирования номеров реализован следующий ряд ролей, которые отличаются своими особенностями и уровнями доступа:

* Guest (Гость)
* User (Пользователь)
* Admin (Администратор)

Любой пользователь интернета может подключиться к сервису, набрав правильный URL в адресной строке браузера. По умолчанию он соединяется с сервисом как пользователь, имеющий роль Guest (гость/пользователь по умолчанию).

В режиме Guest пользователю доступна возможность регистрации и авторизации, просмотр списка отелей и номеров.

При попадании на сервис пользователь видит шапку сайта, в которой располагаются такие разделы как «Главная», «Отели», «О нас», «Регистрация/Личный кабинет». Под шапкой на главной находится система быстрой фильтрации по дате, количеству человек, цене, рейтингу. Основной объем главной страницы занимают горячие предложения с самыми выгодными условиями резервирования. Внизу страницы размещены контакты компаний, адрес физического офиса, почта сервиса и дополнительная информация.

К каждому объявлению прикрепляются отзывы клиентов и их оценки. Все комментарии проходят цензуру и автоматически удаляются, если содержат любое нецензурное слово.

Раздел «О нас» предоставляет гостю более подробную информацию о сервисе. В этом разделе гость может ознакомиться с информацией об отелях.

В разделе «Регистрация/Личный кабинет» гость может создать учётную запись(зарегистрироваться) либо войти в уже существующую с нового устройства, пройдя верификацию через почту или номер, к которым привязан аккаунт. При регистрации гостя он открывает веб-сайт сервиса и нажимает кнопку «Регистрация». Пользователь вводит свои данные, такие как фамилия, имя, отчество, номер мобильного телефона, электронная почта, логин и пароль, и подтверждает регистрацию. После регистрации, он может авторизоваться, вводя свой email или номер мобильного телефона и пароль. Система проверяет данные, и, при успешной авторизации, позволяет пользователю получить доступ к своему аккаунту.

После регистрации пользователь из режима Guest автоматически переключается в режим User, соответствующий пользовательской роли User (зарегистрированный пользователь). Зарегистрированному пользователю доступны все возможности пользователя роли Guest и дополнительные функции, позволяющие ему: зарезервировать номер в отеле на подходящую дату, отменить бронь (за 48 часов до начала), оставить отзыв и поставить рейтинг после проживания в отеле, просмотр всей истории бронирования на аккаунте пользователя. Пользователь может просмотреть и изменить свои личные данные в любое время, воспользовавшись соответствующими функциями сервиса, выйти со своего аккаунта или оставаться в системе для дальнейших бронирований.

Также есть роль Admin, которая занимается администрированием системы. Администратор может авторизироваться, следит за активностью пользователей и может их блокировать или лишать роли User в связи с подозрительной активностью на своё усмотрение. Также он имеет доступ ко всем учётным записям пользователей и контролирует размещение предложений на сайте, управляет отправкой уведомлений о бронировании.

# **Требования к клиентской части**

Клиентская часть должна поддерживаться в любом из современных браузеров, в онлайн-режиме. Клиентская часть должна поддерживать корректный режим работы в любом разрешении экрана, в том числе на мобильных устройствах. Клиентская часть должна быть реализована с помощью Node.js.

Интерфейс веб-приложения должен быть логически завершенным и интуитивно понятным.

# **Требования к серверной части**

Серверная часть должна быть реализована с помощью ASP.NET Core на базе языка C#. Серверная часть должна обеспечивать следующий функционал:

* предоставление API для клиентской части приложения;
* защиту и шифрование персональных данных;
* разграничение прав пользователей по ролям;
* валидацию введенных пользователем данных;
* доступ к базе данных с помощью ADO.NET;
* обработку ошибок.

# **Требования к надежности**

Программное средство для выполнения требований надежности должно:

* хранить пароль пользователя в зашифрованном виде;
* ограничить возможность создания более одной учетной записи на одну электронную почту;
* обеспечивать валидацию введенных пользователем данных;
* обеспечивать защиту личных данных пользователей;
* обеспечивать идентификацию, аутентификацию и авторизацию пользователей.

4.2.1. Устойчивость к нагрузкам.

Программа "Интерфейс бронирования номеров" должна быть спроектирована и реализована с учетом высокой степени надежности при работе под нагрузкой. Система должна обеспечивать стабильную работу и отзывчивость даже при одновременном доступе большого количества пользователей.

4.2.2. Аварийное восстановление.

Программа должна иметь механизмы аварийного восстановления, позволяющие восстановить целостность данных и функциональность системы в случае сбоев или неполадок. Это включает в себя регулярное создание резервных копий данных и механизмы восстановления после сбоев.

4.2.3. Защита от вредоносных атак.

Система должна быть защищена от вредоносных атак, таких как взломы, DDoS-атаки и попытки несанкционированного доступа к данным. Должны быть реализованы средства аутентификации, авторизации и мониторинга безопасности.

4.2.4. Мониторинг и регистрация событий.

Программа должна вести журнал событий, который позволяет отслеживать действия пользователей и системных событий. Это помогает выявлять и анализировать возможные проблемы и инциденты, а также обеспечивать безопасность данных.

4.2.5. Управление ошибками и исключениями.

Система должна корректно обрабатывать ошибки и исключения, чтобы избегать непредвиденных сбоев и потери данных. Должны быть предусмотрены механизмы регистрации и уведомления о возникающих ошибках для последующего анализа и устранения.

4.2.6. Обновление и обслуживание.

Программа должна предоставлять механизмы обновления и обслуживания, чтобы обеспечивать актуальность и работоспособность при изменениях в окружающей среде, а также для внесения улучшений и исправлений.

4.2.7. Отказоустойчивость.

Система должна быть спроектирована с учетом отказоустойчивости, позволяя продолжать работу при отдельных сбоях в компоненте или сервере. Это включает в себя резервное копирование данных и дублирование ключевых компонентов.

4.2.8. Соответствие стандартам безопасности.

Программа должна соответствовать современным стандартам безопасности, включая шифрование данных, защиту от SQL-инъекций и другие меры, чтобы предотвратить утечку и несанкционированный доступ к данным пользователей.

4.2.9. Тестирование надежности.

Проведение тестирования на надежность должно быть обязательной частью разработки, включая тестирование сценариев сбоев, нагрузочное тестирование и тестирование на устойчивость к внешним атакам.

4.2.10. Регулярное обновление и аудит безопасности.

Система должна регулярно обновляться, а также подвергаться аудиту безопасности для выявления и устранения уязвимостей.

Эти требования обеспечивают надежность и безопасность работы программы "Интерфейс бронирования номеров", что является важным фактором для обеспечения удовлетворения потребностей клиентов и защиты их данных.

# **Условия эксплуатации**

Приложение должно быть спроектировано таким образом, чтобы предоставить возможность для комфортного использования всеми членами целевой аудитории – то есть людьми всех полов, возрастов и социальных положений. Для достижения этого, интерфейс приложения должен быть интуитивно понятным и удобным в использовании.

4.3.1. Требования к аппаратному обеспечению.

Программа "Интерфейс бронирования номеров" должна быть доступна для использования на различных аппаратных платформах. Минимальные требования к аппаратному обеспечению для работы программы включают:

Персональный компьютер, ноутбук или мобильное устройство с современным веб-браузером.

Доступ к интернету с минимальной скоростью соединения, обеспечивающей стабильную загрузку веб-страниц и обработку данных.

4.3.2. Условия доступа.

Доступ к программе предоставляется через веб-браузер или мобильное приложение, совместимое с операционными системами, указанными в требованиях к аппаратному обеспечению. Пользователи должны иметь стабильное интернет-соединение для доступа к программе. Для использования определенных функций, таких как онлайн-платежи, могут потребоваться дополнительные условия доступа, такие как наличие банковской карты.

4.3.3. Работа с данными.

Пользователи программы обязаны предоставлять корректные и достоверные данные при бронировании номеров. Для обеспечения безопасности данных пользователей, особенно при онлайн-платежах, рекомендуется использовать защищенное и шифрованное соединение. Программа должна также соблюдать законодательство о защите данных и конфиденциальности.

4.3.4. Обновления и обслуживание.

Программа будет регулярно обновляться для улучшения функциональности, исправления ошибок и обеспечения безопасности. Пользователи должны быть информированы о доступных обновлениях. Обновления и обслуживание могут потребовать краткосрочного простоя в доступе к программе, что будет предварительно уведомлено пользователям.

4.3.5. Соблюдение законодательства.

Пользователи обязаны соблюдать законы и нормативные акты, регулирующие использование программы и бронирование номеров. Программа также должна соблюдать местное и международное законодательство, включая правила обработки и хранения данных.

4.3.6. Инструкции по использованию.

Пользователям должны быть предоставлены подробные инструкции по использованию программы, включая руководства и обучающие материалы, чтобы обеспечить корректное и эффективное использование функционала программы.

Эти условия эксплуатации обеспечивают корректное и безопасное использование программы "Интерфейс бронирования номеров" при соблюдении стандартных требований к аппаратному обеспечению, доступу, безопасности и обслуживанию.

# **Требования к составу и параметрам технических средств**

Минимальные аппаратные требования к пользовательской машине:

* минимальное количество ядер процессора – 2;
* минимальный объем оперативной памяти – 4 Гб;
* минимальный доступный объем дискового пространства – 20 Гб.

Рекомендуемые аппаратные требования к пользовательской машине:

* минимальное количество ядер процессора – 4;
* минимальный объем оперативной памяти – 8 Гб;
* минимальный доступный объем дискового пространства – 40 Гб.

# **Требования к информационной и программной совместимости**

Язык программирования, используемый серверной частью – C#.

Язык программирования, используемый клиентской частью – JavaScript.

Исходный язык, используемый для написания структурированных запросов к базе данных – T-SQL.

Информационные структуры на входе и выходе для передачи данных между клиентом и сервером – JSON. Передача данных осуществляется по протоколу HTTP. Связь между сервером и СУБД – локальная.

# **Требования к программной документации**

## **5.1. Техническая документация.**

### **5.1.1. Руководство по установке и настройке.**

Должно содержать пошаговую инструкцию по установке и настройке программы "Интерфейс бронирования номеров" на серверной инфраструктуре. Инструкция должна включать требования к аппаратному и программному обеспечению, а также настройки базы данных.

5.1.2. Руководство пользователя.

Должно предоставить пользователям подробное описание функционала программы и инструкции по ее использованию. Руководство должно включать в себя снимки экрана и примеры использования.

5.1.3. Документация по безопасности.

Должна включать в себя описание мер безопасности, рекомендации по обеспечению защиты данных и инструкции по обнаружению и предотвращению угроз безопасности.

5.2. Техническая документация для разработчиков.

5.2.1. Описание архитектуры.

Должно содержать высокоуровневое описание архитектуры программы, включая клиент-серверную модель, основные компоненты и их взаимодействие.

5.2.2. Документация кода.

Должна включать в себя комментарии к коду, описывающие структуру, функции и особенности реализации. Комментарии должны быть актуальными и соответствовать стандартам документирования кода.

5.2.3. Документация по API для разработчиков.

Если предусмотрено внешнее API, то разработчикам должна быть предоставлена документация, описывающая доступные методы, параметры запросов и формат ответов.

5.2.4. Схемы базы данных.

Должны быть предоставлены схемы базы данных, описывающие структуру таблиц, связи между ними и типы данных.

5.3. Документация по обслуживанию и администрированию.

5.3.1. Инструкции по обслуживанию.

Должны включать в себя инструкции по регулярному обновлению программы, резервному копированию данных и восстановлению в случае сбоев.

5.3.2. Инструкции по мониторингу.

Должны описывать мероприятия по мониторингу работы системы, включая средства мониторинга и действия в случае выявления проблем.

5.3.3. Инструкции по администрированию.

Должны содержать инструкции для администраторов системы, включая управление пользователями, настройкой доступа и обслуживание серверной инфраструктуры.

5.4. Справочная документация.

5.4.1. Терминология.

Должна включать в себя объяснение основных терминов и понятий, используемых в программе.

5.4.2. Список часто задаваемых вопросов (FAQ).

Должен содержать ответы на часто задаваемые вопросы пользователей для быстрого разрешения типичных проблем.

5.5. Обновление документации.

Документация должна регулярно обновляться и поддерживаться актуальной в соответствии с изменениями в программе и требованиями пользователям.

# **Стадии и этапы разработки**

Разработка должна быть проведена в 3 этапа:

* техническое задание;
* технический проект;
* внедрение.

На стадии «Техническое задание» должен быть выполнен этап разработки, согласования и утверждения настоящего технического задания.

На стадии «Технический (и рабочий) проект» должны быть выполнены перечисленные ниже этапы работ:

* разработка программы;
* разработка программной документации;
* испытание программы.

На этапе разработки программы должна быть выполнена работа по программированию и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями ГОСТ 19.201-78.