**ДИПЛОМЕН ПРОЕКТ**

**ЗА ДЪРЖАВЕН ЗРЕЛОСТЕН ИЗПИТ**

по професия код 481030 „Приложен програмист“

специалност код 4810301 Приложно програмиране“

ТЕМА: „СИСТЕМА ЗА УПРАВЛЕНИЕ НА ЕКЗОТИЧНИ ТУРИСТИЧЕСКИ ДЕСТИНАЦИИ“

Автор:

Жасмина Стоянова Михалева XII Г

Ръководител:

Калоян Тотев

Бургас

СЪДЪРЖАНИЕ

[1 Увод 3](#_Toc194266193)

[2 Цели и обхват на софтуерното приложение 7](#_Toc194266194)

[3 Анализ на решението 10](#_Toc194266195)

[3.1 Потребителски изисквания и работен процес 13](#_Toc194266196)

[3.2 Примерен потребителски интерфейс 21](#_Toc194266197)

[3.3 Диаграми на анализа 27](#_Toc194266198)

[3.4 Модел на съдържанието / данните 27](#_Toc194266199)

[4 Дизайн 31](#_Toc194266200)

[4.1 Реализация на архитектурата на приложението 32](#_Toc194266201)

[4.2 Описание на слоевете, предназначението им, библиотеки и методи включени в съответния слой 33](#_Toc194266202)

[4.3 Организация и код на заявките към база от данни 43](#_Toc194266203)

[4.4 Наличие на потребителски интерфейс (конзолен, графичен, уеб) 44](#_Toc194266204)

[5 Ефективност и бързодействие на решението 45](#_Toc194266205)

[6 Тестване 47](#_Toc194266206)

[7 Заключение и възможно бъдещо развитие 47](#_Toc194266207)

[8 Използвани литературни източници и Уеб сайтове 47](#_Toc194266208)

[9 Приложения 48](#_Toc194266209)

# Увод

В днешния свят на луксозни пътешествия и търсене на уникални преживявания, организацията и управлението на екзотични дестинации се превръща в сериозно предизвикателство. Все повече туристи търсят необичайни локации, персонализирани пакети и изключително обслужване, а туристическите компании и хотелите са изправени пред необходимостта от по-гъвкави, автоматизирани и ефективни решения за управление на техните предложения.

Туристическите агенции, хотелите, луксозните курорти и частните собственици на вили често срещат трудности в координацията между различните услуги, управлението на наличностите и комуникацията с клиентите. Разпокъсаните системи, ръчната обработка на резервации и липсата на интелигентна интеграция често водят до забавяния, грешки и пропуснати бизнес възможности. В същото време, клиентите изпитват неудобства при планирането на своите пътувания, включително ограничена информация за наличните оферти, липса на персонализирани препоръки и сложни процеси на резервация.

За да отговори на тези предизвикателства и нужди на съвременния туристически пазар, Exotico предлага интелигентно и иновативно решение, което оптимизира управлението на резервации, подобрява комуникацията с клиентите, осигурява прозрачност и гарантира безпроблемно потребителско изживяване.

Exotico е иновативна платформа, създадена специално за туристически агенции, собственици и управители на луксозни курорти, хотели и частни вили, както и за индивидуални пътешественици, които търсят по-ефективен и модерен начин за управление на екзотични туристически дестинации. Предоставя интуитивен интерфейс и ефективно управление на резервациите, което прави планирането на екзотични пътувания по-бързо, лесно и сигурно.

Платформата съчетава технологична мощ и графична елегантност, за да осигури по-добро, по-ефективно и персонализирано туристическо преживяване. Използвайки съвременни технологии като Java, Spring Boot и релационна база данни и автоматизирани процеси, Exotico предлага сигурна, мащабируема и високоефективна бекенд инфраструктура, която гарантира безпроблемно управление на дестинации, резервации и потребителски предпочитания.

Exotico е идеална за различни видове туристически услуги, включително:

* Стандартни туристически пакети, които комбинират самолетни билети, хотелско настаняване, организирани дейности и изхранване на база All Inclusive/Полупансион. Платформата автоматизира управлението на резервациите и предоставя прозрачност при избора на дестинация
* Промоционални пакети, които включват специални оферти, сезонни намаления и ексклузивни предложения за екзотични дестинации. Exotico улеснява достъпа до най-добрите оферти и гарантира удобен процес на резервация
* Летни кампании, които предлагат отстъпки за ранни записвания, групови пътувания и специално подбрани туристически програми. Чрез интелигентни алгоритми системата препоръчва най-добрите оферти според потребителските предпочитания

Системата е проектирана така, че да улеснява управлението на резервациите и да подобрява потребителското изживяване, като същевременно предоставя пълна прозрачност на процеса.

Exotico предлага редица значителни предимства, които правят управлението на екзотични туристически дестинации ефективно и лесно. Платформата автоматизира процеса на резервации, намалявайки риска от двойни резервации и човешки грешки, като осигурява безпроблемно изживяване както за туристите, така и за бизнеса. С помощта на интелигентни алгоритми, Exotico анализира потребителските предпочитания и предоставя персонализирани препоръки за дестинации и услуги, което значително подобрява клиентското обслужване.

Една от основните характеристики на Exotico е интеграцията с различни системи, като онлайн платежни платформи, CRM решения и маркетингови инструменти. Това позволява лесно управление на клиентските заявки, плащания и комуникация с потребителите, като спестява време и ресурси на бизнеса. Платформата предлага интуитивен и лесен за използване интерфейс, който прави процеса на резервиране и управление на туристическите пакети бърз и удобен, което е изключително важно за оптимизация на бизнес операциите.

Exotico е проектирана с гъвкавост и мащабируемост в съзнанието, което я прави идеална както за малки туристически агенции, така и за големи международни курорти. Платформата предлага разнообразие от конфигурационни опции, които лесно могат да бъдат адаптирани към различните нужди на бизнеса, гарантирайки, че тя ще расте и ще се променя заедно с бизнеса, който я използва.

Изборът на име за платформата също е важна част от нейната идентичност. Името „Exotico“ съчетава идеята за екзотичност и приключение с технологична иновация. Името подчертава вдъхновението от красивите и уникални дестинации, като същевременно акцентира върху модерния и интелигентен начин за тяхното управление.

* "Exotic" подчертава фокуса върху луксозни, далечни и уникални дестинации.
* "O" символизира завършеност, глобалност и динамика, отразявайки интелигентното управление на туристически услуги

Обединявайки тези елементи, Exotico съчетава иновативност, управление и устойчивост в едно умно и запомнящо се име, което комуникира амбициите на платформата – да направи екзотичните пътувания по-достъпни, ефективни и персонализирани.

С Exotico туристическата индустрия прави крачка към дигиталното бъдеще, където технологиите и изживяванията вървят ръка за ръка. Платформата осигурява по-добро планиране, по-голям контрол и по-висока удовлетвореност както за бизнесите в туристическия сектор, така и за крайните потребители. Екзотичните пътувания стават по-достъпни, по-ефективни и по-персонализирани от всякога.

A diagram of a diagram

AI-generated content may be incorrect.

# Цели и обхват на софтуерното приложение

Съществуващата нужда за по-ефективно и персонализирано управление на туристическите пътувания, особено що се отнася до екзотични дестинации, е основният двигател за създаването на софтуерно приложение за управление на екзотични туристически дестинации. Тези места стават все по-популярни сред туристите, което поставя нови предизвикателства пред туристическите оператори. В отговор на тези предизвикателства, приложението оптимизира процесите в туристическата индустрия, като същевременно подобрява потребителското изживяване и управление на ресурсите. То интегрира нови технологии като Интернет на нещата (IoT), изкуствен интелект (AI) и обработка на големи данни, което позволява по-добро вземане на решения, автоматизиране на процесите и персонализиране на услугите.

Приложението комбинира иновации в технологиите с актуални социални и индустриални тенденции, които насърчават устойчивия туризъм и развитието на по-достъпни и ефективни туристически услуги. То създава възможности за по-добро управление на екзотичните туристически дестинации, като улеснява взаимодействието между туристите и операторите, осигурява лесен достъп до информация и оптимизира процесите на резервации и управление на наличности.

**Цели на приложението:**

* **Интегрирано управление на туристическите дестинации:** Приложението предоставя единно решение за управление на различни аспекти на екзотичните туристически дестинации. То включва управление на резервации, контрол на наличностите, персонализирани предложения и подробни анализи на потребителските данни. Основният фокус е върху оптимизацията на процесите и осигуряването на по-добро изживяване за туристите чрез интелигентни алгоритми и персонализирани услуги
* **Оптимизация на процесите и ефективност:** Приложението използва автоматизация за намаляване на административните разходи, ускоряване на процесите на резервиране и осигуряване на персонализирани предложения за туристите въз основа на техните предпочитания и предишни пътувания. Така се постига по-ефективно управление на всички ресурси и по-добро използване на наличните услуги
* **Сигурност и защита от данни:** Приложението гарантира високото ниво на сигурност на данните и транзакциите чрез интеграция с надеждни онлайн платежни системи и внедряване на най-добрите практики за защита на данни. Това осигурява безопасността на потребителските данни, минимизиране на рисковете от измами и гарантиране на пълна прозрачност при обработката на финансови транзакции
* **Достъпност на информация и лесен интерфейс:** Приложението предоставя интуитивен и удобен интерфейс както за туристите, така и за операторите на дестинациите. Целта е да се улесни достъпът до важна информация за дестинациите, включително наличности, цени, услуги и специални предложения. По този начин, както туристите, така и оператора, имат лесен достъп до нужната информация за планиране и управление на пътуванията

**Обхват на приложението:**

* **Регистрация и автентикация на потребители:** Приложението използва Spring Security за защита на данни и управление на потребителски права. Чрез този механизъм се осигурява сигурен достъп до платформата, като потребителите могат да се регистрират, да влизат в профилите си и да се удостоверяват по различни начини, в зависимост от техните нужди. Управлението на потребителските права позволява различни нива на достъп (например, администратори, оператори и клиенти), което дава възможност за персонализирано изживяване и сигурност
* **Управление на туристическите дестинации:** Приложението предлага пълноценни функции за управление на туристическите дестинации. Операторите могат да добавят нови дестинации, да редактират съществуващите и да премахват неподдържани обекти. Всеки туристически обект може да включва описание на услуги, цени, снимки и информация за настаняване. Това позволява на туристите да получават актуална информация в реално време, което е основно за тяхното вземане на решение при избора на дестинация. Приложението осигурява динамично обновяване на информацията, което е важно за отразяване на промени в наличностите и цените на туристическите обекти
* **Онлайн резервации и настаняване:** Системата предлага интегрирана функция за управление на резервации в реално време. Туристите могат лесно да правят резервации за различни туристически обекти, с автоматизирано потвърждение и известия за статуса на резервацията. Приложението също така прави проверка на наличността в туристическите обекти, което улеснява процеса на планиране на пътуването
* **Филтриране и сортиране на дестинации:** Приложението предоставя възможности за филтриране и сортиране на туристическите дестинации по различни критерии като цена, популярност, рейтинг и вид услуги. Това подобрява потребителското изживяване, като позволява на туристите да намират най-подходящите за тях дестинации, съобразени с техните предпочитания и бюджет. Чрез тези функции туристите могат бързо и лесно да открият дестинации, които отговарят на техните нужди и желания
* **Управление на наличности и оферти:** Приложението предоставя информация в реално време за наличностите на туристическите обекти, както и за текущи и предстоящи промоционални оферти. Чрез автоматизирани промоции и кампании, операторите могат да привлекат повече туристи, като предоставят актуални и персонализирани оферти, базирани на сезонността, търсенето и други фактори. Това позволява на туроператорите да максимизират броя на резервациите и да предлагат по-ефективни маркетингови кампании
* **Проследяване на резервации и плащания:** Приложението осигурява функционалности за проследяване на резервации и плащания. Туристите могат да следят статусите на своите резервации, както и да извършват плащания през различни интегрирани платежни системи. Това позволява на туроператорите да събират плащания по сигурен и ефективен начин, като същевременно предлагат на потребителите лесен начин за управление на техните финансови транзакции. Възможността за извършване на плащания директно през платформата прави процеса бърз и удобен
* **Отчети и статистики:** Платформата предоставя инструменти за генериране на отчети за завършени резервации и финансови транзакции. Тези отчети позволяват на туроператорите да управляват своите финансови потоци по-ефективно, като предоставят точни данни за приходите, разходите и други ключови показатели. Платформата генерира отчети за всички налични резервации, което помага на туроператорите да проследяват и анализират текущите и бъдещите резервации, да управляват ефективно наличностите и да вземат информирани решения за ресурсното разпределение. Това осигурява по-добра видимост на финансовите потоци и оптимизира процесите на управление на дестинациите

Целите и обхватът на софтуерното приложение за управление на екзотични туристически дестинации позволяват не само ефективно управление на туристическите услуги, но и оптимизация на ресурси, подобряване на потребителското изживяване и осигуряване на безопасност и прозрачност в процесите на резервации и плащания. Приложението отговаря на изискванията на съвременните туристи, които търсят персонализирани и иновативни услуги, като същевременно осигурява конкурентно предимство за туроператорите чрез модерни технологии и аналитични възможности.

# Анализ на решението

Системата за управление на екзотични туристически дестинации е уеб базирано приложение, което предоставя лесен достъп до разнообразни екзотични туристически локации, улеснявайки планирането и управлението на пътувания. Чрез автоматизирани функционалности като онлайн резервации, препоръки на дестинации и управление на потребителски профили, платформата осигурява удобство и ефективност в туристическата индустрия.

**Анализ на информацията**

Системата използва структурирани бази данни за съхранение и управление на данните за дестинации, резервации, потребители и наличности. Основните задачи включват:

* **Събиране и обработка на информация** – Дестинациите са категоризирани по различни параметри (рейтинг, цена и т.н.)
* **Управление на резервации** – Поддържане на информация за резервациите в реално време, включително статус на наличности и потвърждения
* **Анализ на потребителските предпочитания** – Използване на машинно обучение или алгоритми за анализ на предишни търсения и резервации за персонализирани препоръки

Приложението също така предлага интелигентни препоръки, като използва статистически анализи. Препоръките могат да бъдат базирани на следния принцип:

* **Оценки и отзиви от потребителите**
* **Актуални промоции и сезонни предложения**

**Доставка на информацията**

За да се улесни разбирането на информацията, приложението разполага с:

* **Интерактивни карти** – Визуализация на туристическите дестинации с интегрирани подробности за всяко място
* **Графики и диаграми** – Статистически данни за потребителските резервации, популярни дестинации и тенденции в туризма
* **Филтри и сортиране** – Възможност за лесно намиране на най-подходящите дестинации
* **Автоматизирани напомняния** – Уведомления за изтичащи резервации

**Контрол и оценка на ефективността**

За да се гарантира качеството и надеждността на платформата, са приети следните методи:

* **Оценка на точността на алгоритмите** – Чрез сравняване на прогнозните резултати с реални данни.
* **Тестване на производителността** – Анализ на времето за реакция на системата при голям брой заявки.
* **Анализ на потребителското поведение** – Събиране на обратна връзка и подобряване на интерфейса според нуждите на потребителите.

**Технологичен стек**

За разработка на приложението са използвани следните технологии:

* **Backend**: **Java** с **Spring Boot** за създаване на мащабируема и сигурна бекенд система
* **Database**: **SQL** за релационна база данни, с **Hibernate** за **ORM (Object-Relational Mapping)**
* **Frontend: HTML, CSS, JavaScript** с възможност за интеграция с **Thymeleaf**

**Алгоритми и оптимизации**

За разработка на приложението са използвани следните алгоритми:

* **Алгоритъм за** **препоръки** – система за персонализирани препоръки, базирана на рейтингова филтрация
* **Алгоритъм за сортиране** – използване на **MergeSort** за ефективно сортиране на дестинациите по цена, популярност и рейтинг
* **Алгоритъм за търсене**  – Оптимизирано търсене по ключови думи
* **Алгоритъм за напомняния** – Изпращане на известия към потребителите за изтичащи резервации

Системата за управление на екзотични туристически дестинации ще предложи модерно и ефективно решение за туристическата индустрия, подобрявайки достъпа до екзотични дестинации и улеснявайки планирането на пътувания. С използването на надеждни технологии и алгоритми, платформата ще предоставя персонализирани предложения, лесно управление на резервации и оптимизирано потребителско изживяване.

## Потребителски изисквания и работен процес

Потребителските изисквания и работният процес за системата за управление на екзотични туристически дестинации са от основно значение за осигуряване на лесен достъп до туристически услуги и ефективно управление на дестинациите. Тази част от анализа се фокусира върху нуждите и очакванията на потребителите, както и върху основните етапи и процеси, които трябва да бъдат взети предвид за оптимизиране на работата на платформата.

**Идентифициране на потребителите**

Първата стъпка в анализа на потребителските изисквания е определянето на целевите групи потребители. Основните категории включват:

* **Туристи** – потребители, които търсят и резервират екзотични туристически дестинации чрез платформата.
* **Туристически агенции** – компании и организации, които публикуват оферти и управляват туристически пакети.
* **Администратори** – лица, които поддържат платформата, осигуряват коректността на информацията и следят за качеството на услугите.

**Определяне на потребностите на потребителите**

За да бъде платформата ефективна, тя трябва да отговаря на следните потребителски нужди:

* **Търсене и резервация на туристически дестинации** – предоставяне на инструмент за лесно търсене на екзотични дестинации въз основа на потребителските предпочитания
* **Филтриране и сортиране на оферти** – възможност за избор на дестинации според цена, популярност, сезонност и рейтинг
* **Онлайн резервации** – система за бърза и сигурна резервация с автоматизирано потвърждения
* **Препоръки за пътувания** – персонализирани предложения за туристически дестинации въз основа на предишни резервации и интереси
* **Управление на наличности** – актуализиране на наличните места и оферти в реално време
* **Проследяване на резервации и плащания** – удобен интерфейс за преглед и управление на направените резервации
* **Сигурност на данните** – защита на личните данни и сигурни онлайн плащания

**Изготвяне на работен процес**

След като са идентифицирани потребителските изисквания, трябва да се дефинира работният процес на платформата. Основните етапи включват:

* **Регистрация и автентикация** – потребителят създава профил и попълва предпочитания за пътувания
* **Търсене и избор на дестинация** – системата предоставя списък с екзотични дестинации с възможност за филтриране и сортиране
* **Преглед на информация и детайли** – потребителят разглежда снимки, описания и отзиви за дестинацията
* **Резервация и плащане** – системата предоставя възможност за онлайн резервация и извършване на плащане
* **Генериране на препоръки** – платформата предлага персонализирани туристически предложения въз основа на рейтингите
* **Проследяване на резервации** – потребителят може да следи и управлява своите резервации чрез профила си
* **Известия и напомняния** – автоматични нотификации за предстоящи пътувания, промоции и актуализации

**Интеграция на системата**

За да осигури по-добро потребителско изживяване, платформата трябва да поддържа интеграция със:

* **Външни API за туристически данни** – интеграции с Booking.com, TripAdvisor и Expedia за допълнителна информация и отзиви
* **Системи за онлайн плащания** – поддръжка на PayPal, Stripe и кредитни карти за сигурни разплащания
* **Системи за нотификации** – имейл и push известия за актуализации и потвърждения на резервации

**Обучение и поддръжка**

За по-добро потребителско изживяване платформата ще включва:

* **Интерактивен наръчник** – ръководство за новите потребители относно използването на платформата
* **Често задавани въпроси (FAQ)** – отговори на най-често срещаните запитвания
* **Форма за обратна връзка** – механизъм за събиране на мнения и предложения за подобрения
* **Техническа поддръжка** – екип за разрешаване на възникнали проблеми и поддръжка на системата

Картина, която съдържа текст, диаграма, План, Техническо чертане

Генерираното от ИИ съдържание може да е неправилно.

Картина, която съдържа текст, Паралелен, екранна снимка, документ

Генерираното от ИИ съдържание може да е неправилно.

Картина, която съдържа текст, екранна снимка, документ, Шрифт

Генерираното от ИИ съдържание може да е неправилно.

Картина, която съдържа текст, Паралелен, документ, номер

Генерираното от ИИ съдържание може да е неправилно.

Картина, която съдържа текст, диаграма, План, Шрифт

Генерираното от ИИ съдържание може да е неправилно.

## A screenshot of a computer AI-generated content may be incorrect.A screenshot of a computer AI-generated content may be incorrect.Примерен потребителски интерфейс

A screenshot of a login form

AI-generated content may be incorrect.

A screenshot of a login form

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect. Приложението предоставя интуитивен и лесен за използване интерфейс за управление на екзотични дестинации. Потребителите могат да разглеждат налични дестинации, да правят резервации и да управляват своите профили. Те имат възможност да преглеждат и анулират активни резервации, както и да оставят рейтинги за посещени дестинации.

Мениджърите имат разширени функции за управление на дестинации, включително добавяне, редактиране и изтриване на дестинации, но всяко тяхно действие трябва да бъде одобрено от администратор. Администраторите имат пълен контрол върху системата – те могат да добавят, редактират и изтриват дестинации без одобрение, както и да управляват потребителски профили, резервации и рейтинги. Те също така могат да генерират отчети за активните резервации и да преглеждат всички действия на потребителите.

Допълнително, страниците за управление на администратори и служители позволяват на администраторите да добавят нови потребители или да изтриват съществуващи, което гарантира пълен контрол върху достъпа до системата. Всички нови записи се съхраняват в базата данни и могат да бъдат редактирани или изтривани по всяко време.

Интерфейсът на системата е проектиран така, че да предоставя лесен и бърз начин за въвеждане на информация, като гарантира ефективност, точност на данните и удобство при използване. Това прави Exotico идеалното решение за управление на екзотични дестинации, като осигурява сигурност и гъвкавост за потребители, мениджъри и администратори.

## Диаграми на анализа

A diagram of a computer

AI-generated content may be incorrect.

## Модел на съдържанието / данните

Моделът на съдържанието/данните е предназначен да съхранява и управлява цялата важна информация, свързана с туристическите резервации, дестинации и потребителите. Той включва различни обекти, техните атрибути и взаимоотношения.

**Обекти:**

1. **Дестинации (Destinations)**

* Таблицата съхранява информация за туристическите дестинации
* Включва данни като среден рейтинг, дати на отпътуване и връщане, описание, цена, статус, брой останали места и популярност
* Връзката с таблицата "Резервации" е чрез външен ключ DestinationId
* Връзката с таблицата "Оценки" е чрез външен ключ DestinationId
* Връзката с таблиците "Кампании" и "Промоции" също е чрез външен ключ DestinationId

1. **Резервации (Reservations)**

* Таблицата съдържа информация за направените резервации
* Включва атрибути като коментар, рейтинг на дестинацията, статус на резервацията, брой хора и обща цена
* Връзката с таблицата "Дестинации" е чрез външен ключ DestinationId
* Връзката с таблицата "Потребители" е чрез външен ключ UserId

1. **Потребители (Users)**

* Таблицата съдържа информация за регистрираните потребители
* Включва данни като имейл, име, фамилия, парола, роля и потребителско име
* Връзката с таблицата "Резервации" е чрез външен ключ UserId
* Връзката с таблицата "Оценки" е чрез външен ключ UserId

1. **Оценки (Ratings)**

* Таблицата съхранява потребителските оценки за дестинациите
* Включва коментари, звезден рейтинг и времеви печат на оценката
* Връзката с таблицата "Дестинации" е чрез външен ключ DestinationId
* Връзката с таблицата "Потребители" е чрез външен ключ UserId

1. **Кампании (Campaigns)**

* Таблицата съхранява информация за маркетингови кампании
* Включва атрибути като процент на отстъпка и стара цена
* Връзката с таблицата "Дестинации" е чрез външен ключ DestinationId

1. **Промоции (Promotions)**

* Таблицата съдържа информация за промоционални оферти
* Включва процент на отстъпка и предишна цена
* Връзката с таблицата "Дестинации" е чрез външен ключ DestinationId

**Взаимоотношения:**

1. **Дестинации (Destinations) ↔ Резервации (Reservations)**

* Една дестинация може да има множество резервации
* Всяка резервация е свързана с една конкретна дестинация

1. **Потребители (Users) ↔ Резервации (Reservations)**

* Един потребител може да направи множество резервации
* Всяка резервация принадлежи на един потребител

1. **Потребители (Users) ↔ Оценки (Ratings)**

* Един потребител може да оцени множество дестинации
* Всяка оценка принадлежи на един потребител

1. **Дестинации (Destinations) ↔ Оценки (Ratings)**

* Една дестинация може да има множество оценки
* Всяка оценка принадлежи на една дестинация

1. **Дестинации (Destinations) ↔ Кампании (Campaigns) и Промоции (Promotions)**

* Всяка дестинация може да има промоции и да бъде част от маркетингова кампания

Тези взаимоотношения и връзки позволяват ефективно управление на данните и гарантират цялостността на информацията в системата за туристически резервации.

**Потребителски роли (3 нива на достъп):**

1. **Администратор:**
   * **Управление на акаунт:** Администраторът има пълен достъп до своя акаунт, включително възможността да изтрива акаунта си, ако е необходимо
   * **Управление на дестинации:** Администраторът има пълен контрол върху добавянето, редактирането и изтриването на дестинации без необходимост от одобрение. Може да изтрива дестинации или да одобрява/отхвърля заявки от мениджърите
   * **Управление на рейтинги:** Администраторът може да преглежда всички дестинации и рейтингите, които са им дадени, и да изтрива неподходящи или некоректни оценки
   * **Управление на резервации:** Администраторът има достъп до всички активни и отменени резервации. Той може да отменя резервации по нужда и да генерира PDF отчети с активните резервации
   * **Управление на потребители:** Администраторът има правото да създава нови акаунти за мениджъри и други администратори, както и да изтрива съществуващи акаунти
2. **Мениджър:**
   * **Управление на дестинации:** Мениджърът може да добавя, редактира и изтрива дестинации, но всички действия изискват одобрение от администратора. Това включва подаване на заявки за добавяне, изриване или промяна на дестинации, които ще бъдат разгледани от администратора
   * **Управление на рейтинги:** Мениджърът има достъп до всички дестинации и рейтингите за тях. Той може да премахва рейтинги, ако съдържат неподходящо съдържание
   * **Управление на резервации:** Мениджърът има достъп до всички активни и отменени резервации и може да ги отменя в случай на нужда. Може също да генерира отчети за активните резервациите в системата
   * **Ограничен достъп до профили:** Мениджърът може да вижда своите данни за профила, но не може да добавя или премахва акаунти на други потребители или служители
3. **Потребител:**
   * **Управление на лични данни:** Потребителят има достъп само до своя акаунт, където може да види своите лични данни като име, имейл и роля
   * **Управление на резервации:** Потребителят може да разглежда своите резервации, да анулира активни резервации и да преглежда историята на миналите си пътувания
   * **Оставяне на рейтинг:** Потребителят може да оставя рейтинг на дестинации, които вече е посетил
   * **Ограничен достъп:** Потребителят има ограничени възможности в системата и не може да управлява дестинации, резервации или профили на други потребители

# Дизайн

Дизайнът на системата Exotico е създаден с внимателно подбрана цветова палитра, която съчетава естетика и удобство, за да осигури приятно и ангажиращо потребителско изживяване. Цветовете са подбрани така, че да създават баланс между модерна визия и лесна навигация, като същевременно гарантират добра четливост и контраст.

Интерфейсът на системата е интуитивен и добре структуриран, като улеснява достъпа до основните функции и навигацията между различните секции. Визуалните елементи са организирани по начин, който минимизира сложността и прави работата със системата удобна и ефективна.

Освен това, Exotico предлага възможност за персонализиране на цветовата тема, което позволява на потребителите да адаптират визията на платформата според личните си предпочитания. Тази гъвкавост осигурява комфорт и по-приятно изживяване при дългосрочна употреба.

Със своя изчистен дизайн и добре подредени елементи, системата гарантира ефективност и удобство, като позволява на потребителите да се фокусират върху управлението на резервации, дестинации и рейтинги без излишни затруднения.

## Реализация на архитектурата на приложението

Архитектурата на приложението определя начина, по който различните компоненти на приложението си взаимодействат, за да осигурят ефективно управление на дестинации, резервации и потребителски профили. Една от възможните архитектури е MVC (Model-View-Controller), която разделя логиката на приложението на три основни слоя:

* **Модел (Model)** – отговаря за управлението на данните и бизнес логиката на приложението. Той съхранява информация за потребителите, администраторите, мениджърите, резервациите и дестинациите. Моделът предоставя методи за достъп, актуализация и изтриване на тези данни, като осигурява структурирана и сигурна работа с тях.
* **Контролер (Controller)** – обработва заявките на потребителите, като действа като посредник между модела и изгледа. Контролерът получава входните данни, валидира ги и комуникира с модела, за да изпълни необходимите операции. След това предоставя обработената информация на изгледа, който я визуализира за потребителя.
* **Изглед (View)** – представлява интерфейса на приложението, с който потребителите взаимодействат. Той е отговорен за визуализирането на информацията и осигуряването на удобни форми за въвеждане на данни. Изгледът получава обработените данни от контролера и ги представя по ясен и интуитивен начин, като гарантира лесна навигация в платформата.

Използването на MVC архитектура в приложението осигурява ясна организация на кода, което улеснява неговото разширяване и поддръжка. Това позволява на разработчиците лесно да добавят нови функционалности, без да нарушават вече съществуващата логика на приложението. Разделението между модел, изглед и контролер допринася за по-добра мащабируемост, сигурност и ефективност при работа със системата.

## Описание на слоевете, предназначението им, библиотеки и методи включени в съответния слой

Приложението е с MVC архитектура, която осигурява ясна структура и висока ефективност при управлението на дестинации, резервации и потребителски профили. Основните слоеве включват слой на контролерите (Controllers Layer), слой на услугите (Services Layer), слой на репозиторитата (Repositories Layer) и слой на моделите (Models Layer).

**Слой на контролерите (Controllers Layer)**

Контролерите са отговорни за обработката на потребителските заявки и връщането на резултати към тях. Те комуникират със слоя на услугите, за да извлекат необходимите данни, и ги предават към изгледа.

1. **AdminProfilesController**
   * Този контролер управлява потребителските профили, като позволява преглед, създаване и изтриване на потребители с роли админ и мениджър
   * Използвани библиотеки:

* Spring MVC (@Controller, @GetMapping, @PostMapping) – обработка на HTTP заявки
* Spring Dependency Injection (@Autowired) – инжектиране на UserService и UserRepository
* Spring Data JPA – работа с базата данни
* Java Streams – филтриране на потребители по роли
  + Методи:
* profilesManagement() – GET заявка
* createUser() – POST заявка
* deleteUser() – POST заявка

1. **AuthorizationController**

* Контролерът управлява потребителската автентикация, като обработва вход, изход, регистрация и достъп до профилната страница
* Използвани библиотеки:
* Spring MVC (@Controller, @GetMapping, @PostMapping) – обработка на HTTP заявки
* Spring Security (SecurityContextHolder, @AuthenticationPrincipal) – управление на потребителската сесия
* Spring Dependency Injection (@Autowired) – инжектиране на SecurityConfig и UserService
* Методи:
* register() – POST заявка за регистрация на нов потребител
* index() – GET заявка за началната страница
* register(Model model) – GET заявка за страницата за регистрация
* logout() – GET заявка за изход от профила
* login() – GET заявка за страницата за вход
* userProfile() – GET заявка за профилната страница на потребителя

1. **DestinationManagementController**

* Контролерът управлява дестинациите, като осигурява преглед, добавяне, редактиране, изтриване и заявки за промени в дестинациите
* Използвани библиотеки:
* Spring MVC (@Controller, @GetMapping, @PostMapping, @RequestParam, @PathVariable, @ModelAttribute) – обработка на HTTP заявки
* Spring Security (SecurityContextHolder, Authentication, @AuthenticationPrincipal) – управление на потребителската сесия и права
* Spring Dependency Injection (@Autowired) – инжектиране на DestinationService и UserService
* Spring Core (ByteArrayResource, Resource) – обработка на изображения
* Java I/O (NIO) (Files, Paths, Path) – управление на файлове

Java Date API (LocalDate, @DateTimeFormat) – работа с дати

* Методи:
* destinationManagementList() – GET заявка за списък с дестинации
* addDestinationForm() – GET заявка за страницата за добавяне на дестинация
* getImage() – GET заявка за зареждане на изображение
* saveDestination() – POST заявка за създаване на дестинация с качване на изображение
* editDestinationForm() – GET заявка за формуляр за редактиране на дестинация
* updateDestination() – POST заявка за обновяване на дестинация
* deleteDestination() – POST заявка за изтриване на дестинация
* requestEditDestination() – POST заявка за заявка за редактиране
* requestDeleteDestination() – POST заявка за заявка за изтриване
* approveRequest() – POST заявка за одобрение на заявка за редактиране/изтриване
* declineRequest() – POST заявка за отказ на заявка за редактиране/изтриване
* viewAcceptedDestinations() – GET заявка за филтриране и сортиране на одобрените дестинации

1. **PromotionAndCampaignController**

* Контролерът управлява промоциите и кампаниите, като осигурява преглед на всички промоции и кампании
* Използвани библиотеки:
* Spring MVC (@Controller, @GetMapping) – обработка на HTTP заявки
* Spring Security (@AuthenticationPrincipal) – идентификация на текущия потребител
* Spring Dependency Injection (@Autowired) – инжектиране на PromotionService и CampaignService
* Методи:
* listPromotionsAndCampaigns() – GET заявка за извличане и показване на всички промоции и кампании

1. **RatingController**

* Контролерът управлява оценките на дестинациите, като осигурява подаване, преглед и изтриване на рейтинги
* Използвани библиотеки:
* Spring MVC (@Controller, @GetMapping, @PostMapping) – обработка на HTTP заявки
* Spring Security (@AuthenticationPrincipal) – идентификация на текущия потребител
* Spring Data JPA – работа с базата данни чрез DestinationRepository, RatingRepository, ReservationRepository
* Java Collections API – използване на List, Map, HashMap за управление на рейтинги
* Методи:
* showRatingPage() – GET заявка за показване на страница за оценяване на дестинации
* rateDestination() – POST заявка за добавяне на рейтинг и коментар към дестинация
* showRatingManagementPage() – GET заявка за администриране на всички оценки
* deleteRating() – POST заявка за изтриване на рейтинг и обновяване на средния рейтинг на дестинацията

1. **ReservationController**

* Контролерът управлява резервациите на потребителите, като осигурява създаване, анулиране и преглед на резервации, както и изпращане на имейл потвърждения
* Използвани библиотеки:
* Spring MVC (@Controller, @GetMapping, @PostMapping) – обработка на HTTP заявки
* Spring Security (SecurityContextHolder, @PathVariable, @RequestParam) – идентификация на текущия потребител
* Spring Data JPA – работа с базата данни чрез ReservationRepository и DestinationRepository
* Jakarta Mail API – изпращане на имейли с потвърждения и анулации
* Java Collections API – List за управление на резервации
* Методи:
* showReservationForm() – GET заявка за показване на формуляр за резервация
* submitReservation() – POST заявка за създаване на резервация и изпращане на потвърждение по имейл
* cancelReservation() – POST заявка за анулиране на резервация и актуализиране на местата в дестинацията
* viewUserReservations() – GET заявка за показване на активни, анулирани и минали резервации на потребителя

1. **ReservationManagementController**

* Контролерът управлява всички резервации в системата, като позволява анулиране, преглед и изтегляне на отчет в PDF формат
* Използвани библиотеки:
* Spring MVC (@Controller, @GetMapping, @PostMapping) – обработка на HTTP заявки
* Spring Security (@AuthenticationPrincipal) – извличане на текущия потребител
* Spring Data JPA – работа с базата чрез ReservationRepository и DestinationRepository
* Jakarta Mail API – изпращане на имейли при анулиране на резервация
* Java Stream API – филтриране на активни, анулирани и минали резервации
* Spring Boot File Handling (ResponseEntity, InputStreamResource) – генериране и изтегляне на PDF
* Методи:
* viewAllReservations() – GET заявка за показване на всички резервации (активни, анулирани и минали)
* cancelReservationManagement() – POST заявка за анулиране на резервация, обновяване на дестинацията и изпращане на имейл
* downloadDestinationsReservationsPdf() – GET заявка за генериране и изтегляне на PDF отчет с активни резервации

1. **StatisticsController**

* Контролерът обработва заявки за статистики и визуализира данни за рейтинги и популярност на дестинациите
* Използвани библиотеки:
* Spring MVC (@Controller, @GetMapping) – обработка на HTTP заявки
* Spring Security (@AuthenticationPrincipal) – извличане на текущия потребител
* Методи:
* showStatistics() – GET заявка за зареждане на страницата със статистики

**Слой на услугите (Services Layer)**

Този слой съдържа бизнес логиката на приложението и управлява комуникацията между контролерите и репозиториите. Той обработва заявките, валидира данните и изпраща резултатите обратно към контролерите.

1. **CampaignService**

* Този сървис обработва кампании с отстъпки за дестинации и комуникира с репозиториите (CampaignRepository и DestinationRepository)
* Използвани библиотеки:
  + Spring Core (@Service, @Autowired) – управление на зависимостите и бизнес логиката
  + Spring Data JPA (@Transactional) – управление на базови операции с транзакции
* Методи:
  + createCampaign(Long destinationId, double discountPercentage) - Създава нова кампания с отстъпка за дадена дестинация
  + getAllCampaigns() - Връща всички активни кампании от базата данни

1. **DestinationService**

* Този сървис управлява дестинациите – съхранение, актуализиране, изтриване и филтриране
* Използвани библиотеки:
  + Spring Data JPA – манипулиране на базата данни (@Transactional, Specification<>)
  + Spring Core – управление на зависимостите (@Service, @Autowired)
* Методи:
  + saveDestination(Destination destination) - запазва нова дестинация
  + getDestination(long id) - връща дестинация по ID
  + updateDestination(Destination destination) - обновява дестинация
  + deleteDestination(Long id) - изтрива дестинация и всички свързани резервации с нея
  + getFilteredAndSortedDestinations(...) - филтрира дестинациите по ключова дума, цена, рейтинг и дати и позволява сортиране по популярност, рейтинг или цена
  + mergeSort(List<Destination> list, String sortBy) - рекурсивно разделя списъка с дестинации
  + merge(List<Destination> left, List<Destination> right, String sortBy) - слива списъците, сортирайки по цена, популярност или рейтинг

1. **MailAndPdfService**

* Този сървис отговаря за изпращане на имейли и генериране на PDF документи с резервации
* Използвани библиотеки:
* Jakarta Mail – изпращане на имейли
* iText PDF – създаване на PDF документи
* Spring Core – управление на зависимостите (@Service)
* Методи:
* sendConfirmationMail(Reservation reservation, User user) – изпраща имейл за потвърждение на резервация
* sendCancelationMail(Reservation reservation, User user) - изпраща имейл за отказана резервация
* generateAllActiveReservationsPdf(List<Reservation> activeReservations) - създава PDF с всички активни резервации, групирани по дестинация
* getMailProperties() - конфигурира **имейл сървъра** за изпращане на съобщения

1. **PromotionService**

* Този сървис управлява промоции за дестинации, като създава и поддържа отстъпки
* Използвани библиотеки:
* Spring Core – @Service, @Autowired за инжектиране на зависимости.
* Spring Data JPA – работа с базата данни чрез PromotionRepository и DestinationRepository.
* Jakarta Transactions – @Transactional за управление на базовите операции.
* Методи:
* createPromotion(Long destinationId, double discountPercentage) – създава промоция към дадена дестинация
* getAllPromotions() - извлича всички активни промоции

1. **RatingService**

* Този сървис е отговорен за управление на рейтинги за дестинации. Той създава нови рейтинги и актуализира средния рейтинг на съответната дестинация
* Използвани библиотеки:
* Spring Core – @Service, за деклариране на сървиса
* Spring Data JPA – работа с репозитории (RatingRepository и DestinationRepository)
* Методи:
* saveRating(Rating rating) – записва нов рейтинг

1. **ReservationNotificationService**

* Този сървис е отговорен за обработка на уведомления за налични изтичащи резервации на потребителите. Изтичаща резервация е такава, чиято дата на заминаване е в рамките на следващите 3 дни
* Използвани библиотеки:
* Spring Core – @Service, за деклариране на сървиса
* Spring Data JPA – работа с репозитория за резервации (ReservationRepository)
* Java Time – работа с дати (LocalDate) за филтриране на резервациите
* Методи:
* getUrgentReservations(User user) - Връща списък с изтичащите резервации на потребителя

1. **StatisticService**

* Този сървис предоставя статистически данни за дестинациите, като информация за средната им оценка и популярността им
* Използвани библиотеки:
* Spring Core – @Service, за деклариране на сървиса
* Spring Data JPA – работа с репозиториите за дестинации (DestinationRepository) и оценки (RatingRepository)
* Java Streams – за обработка на списъци и създаване на DTO(Data Transfer Object) обекти
* Методи:
* getDestinationRatings() - Връща списък от дестинации с техните среден рейтинг
* getDestinationPopularity() - Връща списък от дестинации с брой хора, резервирали всяка от дестинациите

1. **UserService**

* Този клас предоставя методи за управление на потребители, включително създаване, актуализиране, изтриване и зареждане на потребители по различни критерии (например по имейл или потребителско име). Също така е имплементиран за използване със Spring Security, за да се справи с автентикацията на потребителите
* Използвани библиотеки:
* Spring Security - Имплементация на интерфейса UserDetailsService за управление на автентикацията
* Spring Data JPA - Използва се за взаимодействие с базата данни чрез UserRepository
* Spring Security PasswordEncoder - Използва се за криптиране на паролите на потребителите при записване
* Методи
* createUser(User user) – създава нов потребител
* getUserById(int id) – връща потребител по ID
* getUserByUsername(String username) - връща потребител по потребителско име
* updateUser(User user) – актуализира информацията за потребител
* deleteUser(int id) – изтрива потребител
* loadUserByUsername(String username) – зарежда потребител по потребителско име
* getUserByEmail(String email) – връща потребител по имейл

**Слой на репозиторитата (Repositories Layer)**

Този слой съдържа бизнес логиката на приложението и е отговорен за връзката между контролерите и репозиториите. В този слой са използвани библиотеки като Spring Core и Spring JPA, а репозиторитата, включени в него са: CampaignRepository, DestinationRepository, PromotionRepository, RatingRepository, ReservationRepository и UserRepository

**Слой на моделите (Models Layer)**

Този слой съдържа моделите на приложението, които представляват таблиците в базата данни и са отговорни за съхранението и обработката на данни. В този слой не се използват конкретни библиотеки, но се използват анотации като @Entity, @Table, @Column, @Id, @ManyToOne, @OneToOne, @JoinColumn, както и класове като Campaign, Destination, Promotion, Rating, Reservation и User.

Тези слоеве работят заедно, за да осигурят пълната функционалност на системата за управление на сгради.

## Организация и код на заявките към база от данни

**Интерфейс на репозитори на Потребител**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**A screenshot of a computer program

AI-generated content may be incorrect.Клас на модел на Потребител**

Организацията на заявките към базата от данни в тази точка на системата за екзотични дестинации е реализирана с използването на Spring Data JPA. Това позволява да се направят заявки към базата от данни чрез дефиниране на репозиторита за съответните модели на данните.

В предоставения код е видно използването на интерфейс JpaRepository<User,Integer> за репозиторито на модела User. Този интерфейс предоставя базови методи за изпълнение на заявки към базата от данни, като findByUsername(), findByEmail() методите са генерирани автоматично от Spring Data JPA въз основа на името на метода.

В този слой от системата за сгради не се използват конкретни библиотеки за работа с базата данни, тъй като това е реализирано от Spring Data JPA, който автоматично генерира и изпълнява заявки към базата данни

## Наличие на потребителски интерфейс (конзолен, графичен, уеб)

Проектът представлява софтуерна система за управление на екзотични дестинации. На началната страница на уеб сайта се предоставя информация за популярни дестинации, както и бутони за регистрация и вход.

Системата предлага следните функционалности:

1. **Главна страница**: При първоначален достъп до системата потребителят вижда информация за популярни дестинации, както и възможност за регистрация и вход
2. **Основна страница с дестинации**: След успешен вход, потребителят попада на основната страница, където се визуализират всички налични дестинации. Дестинациите, които са налични за резервация, разполагат с бутон за резервиране, докато тези, които не са налични, са маркирани като затворени за резервация
3. **Навигационна лента**: Системата включва навигационна лента, която позволява на потребителите да достъпват различни секции:

* **Промоции и кампании** – съдържа специални оферти и летни кампании
* **Статистики** – показва средния рейтинг и популярността на всяка дестинация
* **Моят акаунт** – предоставя достъп до личния профил и управление на резервации според ролята на потребителя

1. **Моят акаунт**: Секцията "Моят акаунт" съдържа различни подменюта в зависимост от ролята на потребителя (потребител, мениджър или администратор):

* **Профил** – позволява управление на лична информация като име, имейл адрес, телефон и парола
* **Моите резервации** – показва всички резервации на потребителя, разделени на активни, отменени и минали
* **Добави рейтинг** – позволява на потребителя да добави рейтинг към дестинация, която е посетил
* **Управлявай дестинации** – мениджърите могат да добавят, редактират и изтриват дестинации, като всяко действие изисква одобрение от администратор. Администраторите имат пълни права
* **Управлявай рейтинги** – мениджърите и администраторите могат да виждат и изтриват неподходящи рейтинги
* **Управлявай резервации** – виждане на всички активни и отменени резервации, както и възможност за отмяна на активни резервации и генериране на PDF отчет
* **Управлявай профили** – само администраторите могат да добавят или изтриват профили на други администратори и мениджъри

# Ефективност и бързодействие на решението

Ефективността и бързодействието на системата за управление на екзотични дестинации зависят от множество фактори, включително архитектурата на приложението, хардуерните характеристики на сървърите и мрежата, използваните бази данни и техните конфигурации, както и оптимизацията на кода на системата.

За да се гарантира висока ефективност и бързодействие на системата, могат да се приложат следните методи:

* **Кеширане на данни** – Кеширането на данни може да намали времето за заявки към базата данни, като запазва копия на често използваните данни в паметта на сървъра или клиентската машина. Това е особено полезно за често посещавани дестинации или за статистики, които не се променят често.
* **Използване на оптимизирани заявки към базата данни** – Избягването на неефективни заявки, използването на индекси и оптимизация на конфигурациите на базите данни ще подобрят бързодействието на системата. Също така, внимателното структуриране на таблиците и въпросите ще помогне да се минимизират времето за отговор и натоварването на сървъра.
* **Паралелна обработка на заявки** – Разделянето на заявките на независими части, които могат да бъдат обработвани паралелно, ще намали времето за изпълнение на заявките и ще увеличи производителността на системата. Това може да се прилага за обработки на резервации, рейтинги и статистики.
* **Използване на кеширащи системи за изображения и медийни файлове** – За изображенията на дестинациите и други медийни файлове, използването на специализирани кеширащи системи като Cloudinary или подобни облачни решения ще подобри бързодействието на системата и ще спести място на сървърите.
* **Оптимизация на кода** – Оптимизацията на кода на приложението е от съществено значение за намаляване на времето за зареждане на страниците и за намаляване на времето за отговор на заявките на потребителите. Това включва минимизиране на излишния код, използването на асинхронни заявки и опростяване на логиката на интерфейса.

Тези методи ще допринесат за оптимизацията на системата за управление на екзотични дестинации, като гарантират по-бързо зареждане на страниците, по-добра реакция при заявки и по-висока производителност на приложението.

# A screen shot of a computer program AI-generated content may be incorrect.Тестване

A close-up of a computer screen

AI-generated content may be incorrect.A screenshot of a computer screen

AI-generated content may be incorrect.A screenshot of a computer code

AI-generated content may be incorrect.

**Описание на Selenium тестовете:**

1. setUp() - този метод се изпълнява преди всеки тест. Настройва тестовата среда, като автоматично изтегля и стартира правилната версия на ChromeDriver с помощта на WebDriverManager. Браузърът се стартира в headless режим (без визуално отваряне на браузъра), като се задават и допълнителни опции (като големина на прозореца). Инициализира се и WebDriverWait с 10 секунди за изчакване на елементи на страницата
2. testPageTitle() - изпълнява тест, който зарежда началната страница на сайта и проверява дали заглавието на страницата съдържа текста "Exotico - Открий света". Ако заглавието не съвпада, тестът няма да мине
3. testNavigationLinks() - зарежда началната страница и изчаква линковете за "Влез" и "Регистрирай се" да се покажат. След това проверява дали тези линкове са видими на страницата. Ако някой от тях не е видим, тестът няма да мине
4. testDestinationsSection() - този тест зарежда началната страница и изчаква картите за дестинациите Бали, Малдиви и Сейшелски острови да се появят. След това проверява дали тези елементи са видими. Ако някоя карта липсва, тестът няма да мине

A screen shot of a computer

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

**Описание на JUnit и Mockito тестовете:**

1. setUp() - подготвя тестовата среда, като инициализира нов тестов потребител и конфигурира необходимите мокове за репозиторито и енкодер на паролата
2. createUser\_ShouldEncodePasswordAndSaveUser() - тества дали при създаване на потребител, паролата му се криптира правилно и дали потребителят се записва в базата данни чрез метода save() на репозиторито
3. getUserById\_ShouldReturnUser() - тества дали методът getUserById() връща правилния потребител, когато той съществува в базата данни, и дали се прави правилна заявка към репозиторито
4. getUserById\_ShouldThrowExceptionIfUserNotFound() - тества дали методът getUserById() хвърля изключение, когато се търси потребител, който не съществува в базата данни

# Заключение и възможно бъдещо развитие

В заключение, проектът за система за управление на екзотични дестинации Exotico предлага иновативно и ефективно решение за управление на пътувания и резервации, което подобрява както удобството на потребителите, така и процесите на администриране и управление на дестинации. Чрез интеграцията на съвременни технологии и интуитивен потребителски интерфейс, системата гарантира лесен достъп до информация за дестинации, специални оферти и възможности за резервация.

Проектът осигурява персонализирано обслужване, като позволява на потребителите да управляват своите резервации и профили, както и да оставят отзиви за посетените дестинации. За мениджърите и администраторите са осигурени мощни инструменти за управление на дестинации, рейтинги и резервации, които оптимизират процесите и гарантират висока ефективност.

Системата е проектирана да бъде мащабируема и адаптивна, с възможности за бъдещо разширение на функционалностите, като например интеграция на нови дестинации, по-добри алгоритми за препоръки и персонализирани промоции. Използването на съвременни технологии и добра архитектура на системата също така гарантира висока производителност и сигурност на данните, което е от ключово значение за успешното функциониране на системата.

Бъдещото развитие на проекта включва нови възможности за по-голяма автоматизация на процесите, както и интеграция с допълнителни платформи и услуги, което ще подобри потребителското изживяване и ще направи системата още по-интелигентна и ефективна. С тези предимства, системата има потенциала да трансформира начина, по който се организират и управляват пътуванията, предлагайки на потребителите по-богати и удобни възможности за избор и резервиране на екзотични дестинации.

# Използвани литературни източници и Уеб сайтове

CSS, https://developer.mozilla.org/en-US/docs/Web/CSS

Git, https://git-scm.com/doc

HTML, https://devdocs.io/html/

IntelliJ IDEA Class diagrams, https://www.jetbrains.com/help/idea/class-diagram.html

IntelliJ IDEA Database Diagrams, https://www.jetbrains.com/help/idea/creating-diagrams.html

JavaScript, https://developer.mozilla.org/en-US/docs/Web/JavaScript

Spring Security, https://spring.io/projects/spring-security

SQL Server, https://learn.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver16

Thymeleaf, https://www.thymeleaf.org/

# Приложения

Използваните приложения са:

Microsoft Word – за структуриране и написване на документацията на проекта

Microsoft PowerPoint – за изграждане на презентацията на проекта

IntelliJ IDEA 2024.2.0.2 – за изписване на кода, уеб страниците и функционалността на проекта

LucidChart – за създаването на диаграмите

GitHub - за прилежност и подреденост на целия проект, следене на срокове и създавани проблеми