

Факултет по Компютърни системи и технологии

Катедра „Информационни технологии в индустрията"

Дипломна работа

на тема:

Разработване на уеб сайт за фризьорски салон

Дипломант: Васко Колев

Фак. №: 501219038

Специалност: ИТИ

Научен ръководител: доц. д-р Галя Павлова

**Съдържание:**

1. Увод………………………………………………………………………………………...

1. Индустрията на Фризьорството…….………………………………………………….

1.1. История на Фризьорството……………………………………………………….

2. Необходимостта за модернизация на фризьорските салони………………………...

1. Обзор на съществуващи приложения……………………………………………………

1. Общ преглед на проекта………………………………………………………………..

1.1. Предназначение и цели на уеб-сайта…...…………………………………..........

1.2. Плюсове и минуси на уеб-сайт за фризьорски салон..………………………….

1.3. Използвани технологии………..………………………………………………….

2. Сравнение с конкуренти на пазара……………………………………………….........

1. Проектиране на системата...…………………..……………………………………….....

1. Архитектурен дизайн ………………..…………………………………………..........

1.1. Описание на използваните технологии………………………………………….

1.2. Описание структурата на сайта ……….…………………………………………

2. Функционалности на уеб-сайта…….………………………………………………...

3. Дизайн на потребителския интерфейс..… …………………………………..............

1. Програмна реализация.……………………………………………………………………

1. „Бек-енд“ разработка…...……………………………………………………………...

2. „Фронт-енд“ разработка……..…………………………………………………….......

3. Разработка на База данни……………………………………………………………...

4. Тестване и валидация………………………………………………………………….

5. Отстраняване на грешки……………………………………………………………….

6. Деплоймънт и инсталиране……………………………………………………………

7. Поддръжка и оптимизация…………………………………………………………….

1. Ръководство за работа……………………………………………………………………. 1. Инсталация и настройки…………………………………………………....................

2. Ръководство за клиента………………………………………………………………..

3. Ръководство за служителя..…………………………………………………….............

1. Заключение………………………………………………………………………………...
2. Използвана литература…...……………………………………………………………….
3. Приложения…………….…………………………………………………………………..

# **Ⅰ. Увод**

1. **Индустрията на фризьорството**

Индустрията на фризьорството представлява важна и динамична сфера в областта на козметиката и красотата. С течение на времето, фризьорството е израснало от простата услуга по рязане на коса до сложна индустрия, която включва различни стилове, техники и иновации както и грижа за клиента. Тази индустрия не само променя визията на хората, но и играе ключова роля за тяхното самочувствие и увереност. Фризьорите са творци на стил, които не просто режат и оформят косата, а създават изкуство, което отразява личността и индивидуалността на всеки клиент. Тенденциите в сферата постоянно се развиват, отразявайки модерните вкусове и предпочитания на обществото. От класическите стилове до смели и иновативни техники, фризьорството е израз на творчество и еволюция. Със завишаването на стандартите за качество и клиентско обслужване, индустрията се стреми към постоянно усъвършенстване и иновации.

* 1. **История на фризьорството**

Историята на фризьорството е дълбока и богата, простираща се хилядолетия назад. То е тясно свързано с културните и социалните тенденции на различни цивилизации през вековете.

В древността фризьорството е имало огромно значение. В Египет например, фризьорите имали специално признание в обществото, като те не само рязали и оформяли косата, но и се занимавали с магически и религиозни ритуали. В Рим, фризьорите са били известни с блясъка на своите изработки и богатия асортимент от прически.

На по-късен етап, през средновековието, фризьорите са запазили своята важност, но са били подложени на различни закони и правила. През този период, прическите се използвали като символ на обществено и политическо положение.

През Възраждането и Просвещението, фризьорството се развива като професия, която е изпълнена с творчество и иновации. Техниките за оформяне на прическите се разнообразяват, а стилистите стават ключови фигури в обществото.

В 20-ти и 21-ви век, фризьорството се трансформира в индустрия с високи стандарти за красота и стил. Технологиите играят важна роля в професията, като създават нови възможности за оформяне и грижа за косата, както и подобряват клиентското изживяване, което за един успешен бизнес е от голямо значение.

1. **Необходимостта от модернизация на фризьорските салони**

Модернизацията на фризьорските салони в днешно време е ключова за тяхната конкурентоспособност и предоставяне на висококачествени услуги. Някои от основните причини, които подтикват нуждата от модернизация са:

- **Технологични Иновации**. Съвременните технологии предлагат възможности за по-ефективно и удобно управление на салоните. Използването на софтуерни приложения или уеб сайтове за управление записването на часове, управлението на клиентски профили, както и много други, значително подобряват работните процеси.

**- Персонализирано Обслужване**. Модернизацията позволява на салоните да предоставят по-персонализирани услуги на клиентите си. Софтуерните системи могат да запаметяват предпочитанията на клиентите за прически, услуги и продукти, което води до по-добро удовлетворение и лоялност.

**- Онлайн Присъствие и Маркетинг**. Модернизацията включва и развитие на присъствието в онлайн пространството. Начините, с които един салон би могъл да промотира услугите си, включват уебсайтове, социални мрежи, онлайн реклами и много други. Това увеличава възможността от достигане до повече потенциални клиенти.

**- Ефективно Управление на Бизнеса**. Софтуерните системи за управление на фризьорски салон предлагат съхранение и анализ на данни. Извършването на статистика, снабдява служителите на салона с информация и резултати свързани с развитието на салона.

Всички от гореизброените предпоставки отговарят на въпроса защо един фризьорски бизнес трябва да бъде модернизиран. Интегрирането на нови технологии и иновации с фризьорски салон гарантира неговия финансов успех, както осигурява и неговите конкурентоспособност и развитие.

**II. Обзор на съществуващи приложения**

1. **Общ преглед на проекта**

Разработен е уеб-сайт на тема „Фризьорски салон“, или както в днешно време е по-модерно да се нарича – „Бръснарница“.

**1.1.** **Предназначение и цели на уеб-сайта**

Целта на проекта е чрез създаването на подобна уеб апликация да се предостави информация, удобство и персонализирано обслужване, както на потенциални или вече съществуващи клиенти, така и на служители. Неговото цели и предназначение могат да бъдат:

- **Предоставяне на Информация**. Уеб-сайтът служи като виртуална витрина, представяйки услугите, цените, работното време и местоположението на салона. Той предоставя ясна представа за стилът на работата на фризьорите.

**- Запазване на Часове**. Един от ключовите аспекти на сайта е възможността за онлайн записване на часове. Това улеснява клиентите, като им предоставя възможност да изберат удобно време за посещение на салона, без да се налага да звънят или да посещават на място.

**- Персонализирано Обслужване**. Уеб-сайтът може да бъде използван за съхранение на данни на клиентите с техните предпочитания за прически, удобни дни и часове, както и през какъв период от време посещават салона. Това позволява на Бръснарницата да предоставя персонализирани консултации и услуги за своите клиенти при интерес от тяхна страна.

**- Маркетинг и Реклама**. Уеб апликацията освен всичко останало, служи и като инструмент за маркетинг и реклама. Публикуването на статии, съвети за грижа за косата, промоции и специални оферти привлича потенциални клиенти и утвърждава имиджа на салона като място за качествени услуги.

**- Поддържане на Комуникация с клиентите**. Чрез уеб-сайта, представителите на салона могат да поддържат активна комуникация с клиентите си. Предоставена е възможност за връзка със служител на Бръснарницата по един от изброените социални канали или мейл. Друга възможност е попълването на формата за директно съобщение, което може да послужи, за изпращането на критики, похвали или за получаването на информация, преди евентуално посещение в бъдеще.

* 1. **Плюсове и минуси на уеб-сайт за фризьорски салон**

Уеб сайт за управлението на фризьорски салон би помогнал неимоверно на бизнеса да се разрастне и да се наложи сред конкуренцията. До сега в предходните точки се набляга единствено и само на положителните аспекти, докато лошите са оставени на заден план. В тази точка се отбелязват едни от най- големите затруднения, които потребителите изпитват при работа с една такава онлайн система системата:

**- Необходимост от поддръжка**. Редовната актуализация и поддръжка на уеб-сайта изисква време и усилия, особено за обновяване на информацията и съдържанието му.

**- Компютърна грамотност**. Предполага се че в днешно време голяма част от хората, особено младежите, са компютърно-грамотни, но все още има доста, на всякаква възраст, които не успяват да се справят с новите технологии. Поради тази причина, е възможно някои клиенти да се почувстват неудобно при използването на уеб-сайта.

**- Зависимост от интернет връзка**. В случай на проблеми с интернет връзката, клиентите може да изпитат затруднения в достъпа до информацията или записването на часове.

* 1. **Използвани технологии**

Уеб сайтът е разработен, благодарение на технологията Python. С помощта на този иновативен и модерен език за програмиране са изготвени голяма част от функционалностите, налични в уеб-апликацията. С Python е направена и връзката към MySQL база данни, от където апликацията чете данни, съхранявани в таблици и изгледи. Както четене, така съществува възможността и за създаване на нови записи в таблиците. За управление на пакетите и библиотеките в Python се използва технологията Anaconda, която е добре интегрирана с програмния език. Чрез нейният терминал е възможно инсталирането и обновяването на необходими пакети, които се използват в бек-енда на сайта. Един от пакетите който се използва е всъщност Streamlit, чрез който са направени повечето визуални елементи. Други технологии, които са използвани са HTML и CSS, благодарение на които е направен стилът на сайта, както и разположението на някои от визуалните елементи.

1. **Сравнение с конкуренти на пазара**

Пазарът във фризьорската индустрия е динамичен и постоянно променящ се. С течение на времето той е претърпял значителни трансформации, отразяващи се както в стиловете на подстрижките и грижата за косата, така и в технологичните иновации и услугите, които се предлагат в салоните. Тенденциите във фризьорската индустрия се развиват спрямо желанията на клиентите за индивидуалност и изразяване на личен стил. Потърсените услуги включват персонализирани подстрижки, оцветявания с различни техники и тенденции като боядисване в нестандартни цветове. В същото време, клиентите се интересуват все повече от естествената и органичната грижа за косата с продукти, които са екологично чисти и без агресивни съставки. Технологичните иновации също променят начина, по който салоните функционират. Модерните фризьорски салони на пазара се възползват от онлайн системи за записване, социални медии и дори виртуални консултации, за да предложат по-удобни и персонализирани услуги на клиентите си. Със завишаването на конкуренцията, фризьорските салони се стремят към създаване на уникални клиентски преживявания, които включват не само стилови промени, но и специализирани терапии за косата, масажи на скалпа и други услуги за релаксация и благополучие. Този динамичен пазар изисква фризьорите да бъдат в крак с последните тенденции, да имат умения в различни техники и да предоставят персонализирани решения, отговарящи на нуждите на всеки клиент. Това изисква постоянно обучение, креативност и иновации в услугите, предлагани в индустрията, за да се запази конкурентоспособността и удовлетворението на клиентите.

Повечето конкуренти на пазара са вече дигитализирани с цел по-добри резултати. По този начин те гарантират доброто преживяване на клиента . Преди започването на разработката на сайта за фризьорски салон, е направен анализ на всички преки конкурентите, които използват същия метод на дигитализация. В обхвата на анализа влизат само салони, които разполагат с уеб платформа. Прегледани са техните системи и приложения с цел взимане на идеи и изглаждане на неточности. Не всички сайтове са перфектни, но стигат за изпълнение на услугите, които един фризьорски салон предлага. Проектът, обаче се стреми към нещо повече. Не е достатъчно просто да има разработена платформа, която да функционира, а е нужно тя да е направена на база интересите на клиентите. Те са най- важният ползвател на сайта и техният интерес е с приоритет. Ако салонът иска да бъде сред най- успешните на пазара, то трябва да се адаптира и да бъде с една крачка пред останалите. Това ще се случи, като предлага по- иновативни услуги, по-интересен и лесен за използване интерфейс, както и да поддържа по-добра комуникация със своите клиенти. Всички тези идеи са предпоставки за цялостното развитието на салона.

# **III. Проектиране**

Уеб сайтът представлява съвкупност от файлове(снимки), страници, менюта, подстраници, които се обединяват под едно общо домейн име(името на сайта). Един сайт може да има безброй много функционалности, които да улеснят работата на потребителите си. Зад красивия и подреден дизайн на всеки сайт, стоят стотици, дори хиляди редове код. Разработването на една такава уеб апликация не е лесно и много често бизнеси наемат опитни професионалисти в сферата за тази работа.

Сайтът, който е обект на този проект предоставя информация и услуги в индустрията на фризьорството, като подпомага клиенти и служители със своите функционалности. Преди самата разработка се преминава през важни стъпки от проектирането, които полагат основите на една хубава уеб апликация.

Самото проектиране започва с предварително проучване, анализ и стратегия за планиране, уточняване, формиране и обособяване на целите, концепцията, тематиката и дейността на сайта, от страна на клиента, в зависимост от неговите критерии и изисквания по отношение на структура, функционалност, дизайн и насоченост на уеб сайта към определена потребителска интернет аудитория. След това следва определянето на бюджета за сайта и изготвянето на план, с който да се изчисли кога би трябвало сайтът да е завършен и готов за експлоатация. Разбира се, особено важен е и изборът на технологии, с които сайтът да бъде разработен, защото от това зависи какъв ще бъде бюджета. За целта на проекта са избрани само безплатни за ползване технологии, за да се избегнат нежелани допълнителни разходи. Следва процесът по проектиране на архитектурата на сайта, който ще определи как избраните технологии ще комуникират помежду си и как ще бъдат интегрирани заедно, за да създадат желания продукт. След избора на технологии, следващите етапи са от проектирането е свързан с определяне на ключовите функционалности, които ще бъдат налични за използване. Следва проектиране на потребителския интерфейс, както и на базата данни.

1. **Архитектурен дизайн**

Както вече стана ясно разработването на Архитектурния дизайн е важна част от проектирането на един сайт. Тук се включват етапите по организация на информацията и структурата на сайта, които улесняват навигацията на потребителите и осигуряват функционалността на уебсайта. Архитектурния дизайн включва няколко важни аспекта:

* **Информационна архитектура.** Определя начина, по който информацията е организирана и представена на сайта. Тук се включват менютата, категориите, подкатегориите и начина, по който потребителите навигират в сайта.
* **Функционална архитектура.** Определя функциите и възможностите, които уебсайтът предлага. Това включва функционалности като търсене, филтриране на информация, интерактивни форми, онлайн плащания и други.
* **Дизайн на потребителски интерфейс (UI).** Това е визуалната част на уебсайта - цветове, шрифтове, изображения, бутони и други елементи, които потребителите виждат и с които взаимодействат.
* **Дизайн на потребителски преживявания (UX).** Така нареченият „UX дизайн“ се фокусира върху удобството и удовлетворението на потребителите при използването на сайта. Той се стреми да улесни навигацията и да осигури приятно и лесно използване на функциите.
* **Техническа архитектура.** Включва избора на технологии, използвани за разработката на сайта.

Архитектурният дизайн обединява тези аспекти, за да създаде цялостна структура на уебсайта, която е интуитивна за потребителите, функционалността на която е ефективна и предлага приятно преживяване при използване.

* 1. **Описание на използваните технологии**

Използваните технологии са част от архитектурния дизайн на един уеб сайт, като благодарение на тях се осъществява реализирането на желания резултат. Технологиите, използвани за изготвянето на този проект са Python, Streamlit, MySQL, Anaconda, CSS и HTML. Всяка една от тях има както силни така и слаби страни. В тази точка от документацията, всички технологии ще бъдат описани какво представляват и какви са техните плюсове и недостатъци.

**Python**. Според литература 13, това е език за програмиране от високо ниво, интерпретативен, интерактивен, обектно ориентиран, създаден от Гуидо ван Росум в началото на 90-те години. Неговото име идва от фен на британския комедиен сериал "Монти Пайтън". Често бива сравняван с други програмни езици като *Perl*, *Scheme* и *Java* .

Идеята за *Python* се заражда в края на 1980-те години, като реалното осъществяване започва през декември 1989 г. от Гуидо ван Росум в *CWI (Centrum Wiskunde & Informatica)* – международно признат изследователски институт по математика и компютърни науки, локализиран в Амстердам, Холандия. *Python* имал за цел да се превърне в наследник на *ABC* (език за програмиране, от своя страна вдъхновен от *SETL*), който да бъде способен да обработва изключения и да е съвместим с операционната система *Amoeba*. Ван Росум е основният автор на *Python*, а неговата продължаваща централна роля в развитието на езика е ясно отразена в титлата, дадена му от *Python* общността – „пожизнен доброжелателен диктатор“.

*Python* позволява разделянето на една програма на модули, които могат да се използват отново в други програми. Също така притежава голям набор от стандартни модули, които да се използват от потребителя като основа на даден проект или програма.

Поради факта, че *Python* е интерпретативен език, се спестява значително време за разработка, тъй като не са необходими компилиране и свързване (linking) за тестването на дадено приложение. Програмната му идеология е сходна с тази на Java и всяко приложение, написано на него, е сравнително лесно преносимо и в други платформи (или операционни системи). Програмите написани на *Python*, са доста компактни и четими, като често те са и по-кратки от своите еквиваленти, написани на *C/C++*. Това е така, понеже, наличните сложни типове данни позволяват изразяването на сложни действия с един-единствен оператор. Групирането на изразите се извършва чрез отстъп, вместо чрез начални и крайни скоби или някакви ключови думи. Не са необходими декларации на променливи или аргументи. *Python* съдържа прости конструкции, характерни за функционалния стил на програмиране, които му придават допълнителна гъвкавост. Програмите, написани на *Python*, представляват съвкупност от файлове с изходен код. При първото си изпълнение този код се компилира до байткод, а при всяко следващо се използва кеширана версия. Байткодът се изпълнява от интерпретатор на *Python*. Езикът е строго типизиран, тоест при несъответствие между типовете е необходимо изрично конвертиране. Поддържа се и динамична типизаци, или с други думи казано, типовете на данните се определят по време на изпълнението. Работата се основава на принципа duck typing – типът на обектите се оценява според техните свойства. Някои от силните страни на езика, и до голяма степен причините, че точно той е избран за разработката на този проект са:

* **Четим и лесен за усвояване**. Синтаксисът на *Python* е интуитивен и подобен на човешкия език, което го прави достъпен за начинаещи.
* **Голяма общност и библиотеки**. Има активна общност и обширна екосистема от библиотеки, които позволяват разработка на широк спектър от приложения.
* **Възможност за разработка на множество приложения**. *Python* е надежден и се използва в различни области като уеб разработка, наука, изкуствен интелект, машинно самообучение, анализ на данни и много други.
* **Платформено независим**. *Python* е платформено независим, което означава, че кодът може да се изпълнява на различни операционни системи, както е и съвместим с множество компилатори и платформи.
* **Гъвкавост**. Езикът предлага голяма гъвкавост при разработката и позволява използването на различни стилове на програмиране (обектно-ориентирано, функционално и други).

Както всяко едно нещо, Python, освен положителните страни има и своите лоши такива. Някои от недостатъците на езика, поради които професионалисти избягват работа с него са:

* **Изпълнение.**  В сравнение с други езици като *C* или *C++***,** *Python* може да бъде по-бавен поради интерпретирането на кода, което го прави по-малко подходящ за използване в комплексни приложения, изискващи висока производителност.
* **Мобилни приложения**. Изграждането на мобилни приложения с *Python* не е толкова доминиращо, както в други езици като *Java* или *Swift*.

Въпреки слабостите си, *Python* остава изключително популярен и предпочитан език, особено за изучаване на програмиране от начинаещи и за разработка в различни области. Езикът е в основата на този проект, като благодарение на неговият набор от библиотеки и модули са изработени голяма чат от функционалностите на сайта, както и връзката към базата данни. Много известена и често използвана библиотека на *Python* е именно *Streamlit*, която заема важно място в разработката на уеб-сайта. *Streamlit* e фреймуърк на *Python* и е създаден за цели в областта на интерактивните уеб приложения.

**Streamlit**. Според литература 11 и 12 , това е безплатна и лесно достъпна библиотека на *Python*, предназначена за създаване на уеб приложения с много по- малко код, от колкото е нужно на други езици или библиотеки. Тя позволява на потребителите си лесно да създават интерактивни приложения за визуализация на данни, създаване да модели от данните, проектирането на приложения използващи изкуствен интелект и много други. Наборът от модули и методи, с които технологията разполага, предлагат на потребителя едно изключително преживяване при използване и възможността за постигане на интересни и интерактивни апликации. Именно и в този проект голяма част от визуалните елементи, форми, полета, бутони са реалност благодарение на Streamlit. При започване на работа с библиотеката, потребителят трябва да инсталира необходимия пакет от своя терминал или конзола. След това е необходимо в началото на програмата, на някой първите редове, да се изпише командата „*import*“ и името на библиотеката, с което се вмъкват функционалностите на *Streamlit*. По този начин на потребителя се предоставят голяма вариация от команди и услуги, с които процесът на разработка започва. Технологията предлага следните предимства, които подпомагат привличането на повече както професионални разработчици, целящи сложни и целенасочени приложения, така и потребители, имащи за цел създаването на по-прости и любителски апликации:

* **Лесен за употреба.**  *Streamlit* е много лесен за научаване и използване, особено за хора с неголям опит в уеб разработката. Със сравнително малко усилия може да се създаде функциониращо уеб приложение.
* **Бързо прототипиране**: Позволява бързо създаване на интерактивни уеб приложения, които визуализират данни или резултати от машинно обучение. Идеален инструмент за демонстриране на модели от данни и проекти.
* **Интеграция с Python**: Тъй като е базиран на *Python*, *Streamlit* лесно се интегрира с широка гама от библиотеки за обработка на данни и машинно обучение като *Pandas*, *Matplotlib*, *TensorFlow* и други.
* **Динамични компоненти**: Позволява гъвкавост в изграждането на приложения, като позволява на потребителите си да променят и да взаимодействат с елементите на интерфейса.
* **Поддръжка и общност**: Има активна и отзивчива общност, както и редовни обновления, които поддържат платформата актуална и съвместима с най- новите тенденции.

Освен положителните черти на *Streamlit*, за съжаление има и отрицателни такива, които предизвикват сериозни главоблъсканици, при потребителите си, по време на разработка:

* **Ограничена гъвкавост за сложни приложения**. Въпреки че е лесен за употреба, *Streamlit* може да има ограничения за по-сложни приложения, които изискват специфични и сложни функционалности.
* **Ограничена персонализация**. Има ограничени възможности за персонализация на интерфейса или дизайна на приложенията, което може да бъде ограничаващо за някои потребители.
* **Нестабилност при обновяване**. Понякога обновяванията на апликацията могат да доведат до нестабилност или да нарушат съвместимостта с предходно изпълнени парчета код.

**Anaconda.** Безплатна и отворена дистрибуция на *Python*, която включва *Python* интерпретатор, множество от основните библиотеки и инструменти, свързани с данните и машинното обучение, както и удобна среда за управление на пакети и среди. В основата си, *Anaconda* е платформа, която улеснява работата с *Python*, особено за анализ на данни. *Conda* е инструмент за управление на пакети, който е включен в екосистемата на *Anaconda.* Благодарение на него, потребителят може да създава отделни виртуални среди за различните си проекти, по които работи, като във всяка среда се съхраняват и модифицират множество библиотеки и пакети. Основните функционалности на *Conda* биват:

- **Управление на Пакети**. Позволява лесно инсталиране, актуализиране, изтриване и търсене на пакети. Автоматично управлява зависимостите между пакетите и гарантира тяхната съвместимост.

- **Създаване на Среди (Environments)**. С *Conda* може да създавате отделни среди, които изолират различните проекти. Това позволява на потребителите да работят с различни версии на пакети без конфликти помежду им.

- **Изисквания на Пакети (Package Requirements)**. Позволява ви да дефинирате изрично списък със зависимости и версии на пакети, които вашето приложение изисква за да функционира.

- **Скорост и Ефективност**. Бърз при инсталиране на пакети и управление на зависимости, което прави процеса на разработка по-ефективен.

- **Широка Съвместимост**. *Conda* е наличен за различни операционни системи като *Windows*, *macOS* и *Linux*.

*Conda* е мощен инструмент, особено когато става въпрос за управление на пакети и среди при разработка на софтуер с *Python*, предоставяйки гъвкавост, контрол и улеснения при работа с различни проекти и зависимости.

**MySQL.** Една от най- разпространените и използвани системи за управление на база данни. Бази данни наричаме структурирано количество от данни. Бази данни наричаме всичко от най-обикновен списък за пазаруване, през изображения в галерия до огромните количества информация в корпоративните мрежи. За да може да се работи с данни, които са съхранени в базите данни на компютъра, е нужна система за управление на базите данни, като *MySQL* *Server*. Тъй като компютрите вече могат да боравят с огромни количества от данни, системите за управление на базите данни играят ключова и централна роля, било то като отделни приложения или като част от такива. *MySQL* база данни е от релационен тип, което ще рече, че данните се съхраняват в отделни таблици, вместо да се поставят в една голяма директория, без подредба или каквато и да е била йерархия между тях.  За по**-**бърза работа структурите от бази данни са организирани във физически файловe. Логическият модел, с обекти като бази данни, таблици, изгледи, редове и колони предлага гъвкава програмна среда. Съществува възможността за настройване на правила, управление на връзките между различните полета с данни, като „*one-to-one*“, „*one-to-many*“, „*unique*“, “*required*” или „*optional*“, както и различни видове ключове между различните таблици. Базите данни налагат тези правила, така че с добре-проектирана база данни разработваните приложения никога няма да се сблъскат с несъвместими, дублиращи се, остарели или липсващи данни. Някои от основните черти, които характеризират MySQL и й дават предимство пред останалите системи за обработка и поддръжка на данни са:

- **Релационна База Данни**. Система за управление на бази данни (СУБД), която се основава на релационен модел. Това позволява структуриране на данните в таблици със свързани връзки между тях.

- **Безплатна за използване**: *MySQL* е безплатна и отворена за използване технология, която се разпространява под лиценза GPL (*General Public License).* Това позволява на потребителите да я използват, модифицират и разпространяват свободно.

- **Поддръжка от множество Операционни Системи**. Съвместима с различни операционни системи като *Windows*, *Linux*, *macOS* и други.

- **Език за Заявки (SQL).** Използва *SQL* (*Structured Query Language*) за манипулиране на данни, като предлага широк спектър от заявки за създаване, четене, актуализиране и изтриване на данни от базата данни.

- **Скорост и Ефективност**. *MySQL* е известна със своята бързина и ефективност при обработката на големи обеми данни и при работа с множество потребители.

- **Голяма Общност и Поддръжка**. Има голяма общност от разработчици и потребители, което допринася за наличието на помощ, ресурси, разнообразие от разширения и инструменти.

Гореизброените съществени черти, които характеризират MySQL, са част от предпоставките, тази технология да играе важна роля в изготвянето на проекта. В кода на сайта е имплементирана връзка (*connection*) към базата, което позволява непрекъснатото взаимодействие с нея и извършването на операции като Четене, Записване, Изтриване и Обновяване. Всичките данни които приложението акумулира се записват в базата данни като по този начин могат да се изготвят както интерактивни функционалности, така и да се извършват анализи, жизнено важни за проследяване развитието на фризьорския салон.

Освен технологиите изброени до сега, не по-малки по важност са *HTML* и *CSS*. **HTML**. Това е най-популярният език за изработка и описание на уеб страници.Той е стандарт в Интернет, а правилата се определят от Международната организация по стандартизация (*World Wild Web Consortium*). Началото на *HTML* датира от навлизането на Световната мрежа през 1991 година. Изработването на HTML документа става с помощта на така наречените тагове. Чрез тях се създават и се определя разположението на отделните елементи в уеб страницата като заглавия, таблици, форми, бутони, текстови полета, текст, изображения, видеа и други. По своята същност това не е програмен език, а описание на това как искаме да изглежда съдържанието, на структурата на самото съдържание. *HTML* е текст, който ние подаваме на браузъра, а той от своя страна го интерпретира като последователност от команди/инструкции, които изпълнява и по този начин подрежда обектите в страницата. *HTML* таговете са команди, които се ограждат със знаците „<“ и „>“, например <p> за параграф. Те са фиксирани, за разлика от тези в *XML*. Често образуват двойка от отварящ и затварящ таг, например двойката тагове за параграф е: <p> и </p>. Текстът, заключен между <p> и </p> ще бъде представен на Интернет страницата като един параграф. Браузъри като *Internet Explorer*, *Firefox* и *Google Chrome* прочитат *HTML* документите и ги показват като уеб страници. Те не показват HTML таговете, а ги използват, за да интерпретират съдържанието на страницата.

**CSS.** Според литература 6, това представлява скриптов език за описание на стилове. *CSS* се използва за визуално оформление на документи. Най-често се имплементира за *HTML* документи. Чрез този език се задават различни атрибути на вече използваните тагове в *HTML* документа като цвят, размери, фон, разположение, прозрачност и много, много други. *CSS* има и своите стилови правила за изписване. Използва селектори, чрез които се избира върху кой елемент и таг да бъде приложен дадения стил. След това в специален блок, започващ със скобата „{“ и завършващ с “ }“, се изписват свойствата, които искаме да добавим към даден обект, последвани от символа „:“ и техните стойности. *CSS* е съкращение от *Cascading Style Sheets*, в свободен превод „каскадни стилови страници“. Стиловете, указани чрез него, определят как ще изглеждат на екрана елементите на един *HТМL* документ (Интернет страница). *CSS* се наричат каскадни, понеже стилове могат да указват браузърите, потребителите и авторите на Интернет страниците. Тези стилове се наслагват, в известен смисъл преливат един в друг, откъдето и термина „каскаден“. Чрез работата с този език се спестява много работа. Дизайнът на всички Интернет страници на един сайт може да се зададе само с един *CSS* файл, който да е външен и да се свързва към всеки един от *HTML* файловете. Голяма част от използваните стиловете, са много добре поддържани от всички широко употребявани и популярни браузъри. В този проект *CSS* и *HTML* са използвани по- малко от останалите технологии, но допринасят за цялостното изпълнение на заданието и позволяват сайта за фризьорския салон да придобие съвременна визия. Благодарение на тях е зададен фонът на уеб апликацията, както са добавени и множество снимки и подравнявания на текстове и полета. *CSS* и *HTML* са в основата на по- доброто преживяване на потребителя при използване функционалностите на сайта.

* 1. **Описание структурата на сайта**

Правилното структуриране на страниците в един уеб сайт дава възможност на неговите посетители да намерят по-лесно съдържанието, което търсят. Добре организираното съдържание помага и на търсачките да разберат кое е по-важното съдържание на сайта. При сайтове, които трупат съдържание в продължение на години, е твърде вероятно да настъпи момент, в който важните страници в навигацията са станали прекалено много, а това води по-скоро до объркване на потребителя. В такъв случай, може

тези стари страници от навигацията да бъдат премахнати и да се остави връзка към тях в някоя от по-новите. Ако информацията на някоя страница вече не е актуална, може да се пренасочи към близка по тема нова страница. Така потребителите пак ще имат достъп до тези по-стари страници без те да разсейват структурата на сайта или пък да оставят чувство за неактуалност.

Визуално структурата на един сайт се представя чрез навигацията (или менюто). Добрата навигация съдържа малък брой елементи – така могат да се отделят наистина важните категории и да се запази фокусът върху основната тема на сайта. Свикнали сме да виждаме „Начало“ и „Контакти“ при добре организираните и интуитивни сайтове – ако не намерим тези елементи в навигацията, оставаме с чувството, че нещо липсва. Останалите позиции между „Начало“ и „Контакти“ следва да бъдат запълнени с най-важните теми или продуктови групи. Силно препоръчително е да има само едно ниво на подменю, но това в много ситуации може да се окаже невъзможно. Въпреки това е добре да знаем, че има и други начини за свързване на една страница с друга освен главната навигация. Трябва да се помисли кои са най-важните теми за сайта? Те трябва да бъдат в навигацията. Кои са подтемите в главните теми? Понякога помага изчертаването структурата на сайта върху лист хартия, като се начертават основните позиции, така ще бъде по-лесно да се организират страниците му. Дори малките промени в структурата на сайта могат да имат огромно въздействие върху потребителя и класирането в търсачките.

В този проект, сайта за фризьорски салон се отличава с няколко важни компонента на структурата. Те биват “Фронт-енд”, „Бек-енд“, „База данни“ и „Автентикация на потребителите“.

- **“Фронт-енд” (Потребителски Интерфейс)**. Сайтът използва наличните *Streamlit* методи и функции, за да може да се създаде интерактивен потребителски интерфейс. Това включва визуализации на данни, форми за въвеждане на информация, графики, текст и други елементи, които потребителите могат да виждат и с които могат да взаимодействат. Използвани са множество методи от библиотеката *Streamlit,* като някои от тях биват форми за вписване на данни, форми с които са постигнати принципите на потребителска автентикация и оторизация, диаграми, различни видове бутони и много други. Със *Streamlit* е постигната и навигацията в сайта. Всяка избрана опция, визуализира различна страница, показваща съответната информация, която отговаря на нея. „Фронт-енд“ частта включва и всички стилове заложени в кода на сайта, с които потребителският интерфейс предпоставя за клиента по-добро изживяване, ползвайки услугите на уеб апликацията

- **„Бек-енд“** **(Логика и Връзка с Базата Данни)**. *Python* кодът съдържа бизнес логика и връзка с базата данни. Той обработва данни, извлича информация от базата данни, обработва заявки от потребителския интерфейс и прави манипулации с данни, като записване, четене, актуализиране и изтриване. Python кодът е гръбнака на този проект, като без него не би могло да се изпълни заданието. Използваните библиотеки, предлагани от програмния език са в основата на изработването на ключовите функционалности.

- **База Данни** **(*MySQL*)**. *MySQL* базата данни съхранява цялата информация, акумулирана от работата със уеб сайта, като потребителски данни, конфигурационни настройки, данни за продукти или услуги и други. Същността на релационния принцип на базата дава възможност за групирането на всичките данни в отделни таблици, свързани помежду си.

- **Връзка между Компонентите**. *Python* кодът, чрез подходящи библиотеки за връзка с *MySQL* базата данни, осигурява свързаността и взаимодействието между потребителския интерфейс (*Streamlit*) и базата данни (*MySQL*). Това включва изпълнение на SQL заявки, обработка на резултатите и предаване на данни между двата компонента. Повечето визуални елементи в интерфейса на сайта работят с данни. Това ще рече, че при всяко взаимодействие на потребителя с тях, *Streamlit* изпраща заявка до сървъра и той следователно модифицира данните в *MySQL* базата. При следваща интеракция на потребителя с даден визуален елемент той вижда обновените налични данни.

- **Сигурност и Управление на Данни**. В уеб сайта са включени механизми за автентикация и управление на достъпа за клиентите на фризьорския салон, за да се гарантира сигурността на данните. Потребителите могат да достъпват елементи и функционалности само ако са оторизирани за това действие.

1. **Функционалности на уеб сайта**

Уеб сайтът е съвкупност от различни функционалности, които определят неговата същност и предназначение. В този проект функционалностите на уеб апликацията се отнасят за фризьорската индустрия и съществуват за да улеснят клиента максимално и да подобрят неговото изживяване. Някои от най- важните функционалности на сайта за фризьорски салон са:

- **Записване на часове**. Това е най- важната функционалност, без която един сайт за фризьорски салон не може. Без записване на часове, сайтът губи своя смисъл. Когато клиентът реши да запази час, той има право на избор между 7 вида прически, описани както ценово, така и какво представляват. След като е готов с първия си избор, потребителят може да избере дали желае подстригване и оформяне на брадата. Следващият избор, пред който е поставен клиентът е селекцията на удобен час и ден. Часовете, вече запазени от други клиенти, не би трябвало да бъдат достъпни. Налична е и опция за плащане на прическата с кредитна карта. След като клиентът е направил избора си, се визуализира кратко обобщение на услугата, преди тя да бъде успешно изпратена и записана.

- **Поле за коментари**. В страницата си за контакти, сайтът разполага с поле, в което потребителят може да изпраща запитвания, коментари, да изразява мнение или да коментира своето изживяване след процедура. Съобщението на клиента не трябва да надвишава максималният брой допустими символи, а именно 500. След като съобщението е готово, то може да се изпрати до представител на са сайта с кликване на бутона под полето. Коментарът се записва в таблица от базата данни и потребителят получава отговор по посочения мейл при регистрация в най- скоро време.

- **Локация на салона**. Функционалност, отново разположена в страницата за контакти, указваща точни координати на фризьорския салон. На страницата е налична карта, която позволява на потребителя да увеличава, намалява размера на картата, както и да приближава и отдалечава. Ако клиентът не е запознат с района, то под картата е разположен бутон, с който може да се види точната локация в *Google Maps,* където са налични и наименования на улиците. По този начин, клиентът може да приложи местоположението на салона като желана дестинация и да получи необходимите напътствия.

Следните функционалности в сайта за фризьорски салон са налични в последната страница, където достъпът е ограничен и визуалните елементи са видими само за малка група от хора, които са оторизирани с необходимите права за тази цел:

- **Таблица с коментари**. За всички потребители на сайта, които имат необходимите права е наличен списък с коментари и съобщения от клиенти, изпратени чрез полето за коментари от страницата с контакти. Таблицата се състои от редове и колони и отбелязва едни от най-важните данни за клиентите – потребителско име и имейл. Има и допълнителни полета, в които са описани точната дата и час на коментара, самият коментар или съобщение, както и дали е даден отговор на даден клиент. Когато администратор изпрати отговор на съответния клиент по мейл, той трябва да обнови таблицата с коментари, като промени статуса му от „неотговорено“ на „отговорено“.

- **Таблица с регистрирани потребители**. Налична за администраторите на сайта е и таблица с регистрираните потребители или клиенти. Визуализирани са в 2 полета потребителското име и датата на регистрация. Под таблицата е разположен бутон, след натискането на който, е предоставена по-подробна информация за клиентите. Допълнителните полета, които се визуализират след натискането на бутона се отнасят съответно за името и фамилията на клиента, както и неговият имейл адрес.

- **Таблица със записани часове.** Предоставя се възможност за следене на бъдещи и изминали часове за подстригване. Отново под формата на 2 таблици е описана цялата необходима информация за записани часове в салона. По този начин фризьорите няма никога да пропуснат уговорка с клиент и ще могат по- лесно да разпределят работата помежду си. Полетата описани в таблиците с часове биват – потребителско име, име, фамилия, дата на уговорката, както и час, вид на прическата, дали клиентът е пожелал допълнително подстригване на брадата, какъв е методът на плащане( в брой или с карта) и финалната цена на процедурата в български левове.

- **Лентова графика**. Последната функционалност на сайта, но не последна по важност е статистическа графика, визуализираща броя на запазени часове в посочен диапазон от дни. Пред потребителя са поставени два избора, дали лентовата графика да описва броя на часове по месеци или по дни. Има два налични бутона за съответните опции. Когато един от двата бутона бъде натиснат графиката се илюстрира на екрана. Графиката е интерактивна и може да бъде както уголемявана, така и смалявана. Когато потребителят постави курсорът на мишката върху някоя от лентите от диаграмата, се предоставя информация за броя на часовете и името на съответния месец или ден. Тази функционалност предоставя възможността за следене развитието на салона, благодарение на съхранените данни.

**3. Дизайн на потребителския интерфейс**

В днешния дигитален свят, важността на дизайна на потребителския интерфейс (UI) не трябва да бъде подценявана. За по-голяма част от хората по света дизайнът на потребителския интерфейс оформя начина, по който действаме и взаимодействаме всеки ден, дори ежеминутно. [UI дизайнът](https://studioweb.bg/post/novostite-v-mobilniya-ui-dizayn-4060) е буквално естетическият дизайн на интерфейса на продукт: приложение или уебсайт. Успешният дизайн на потребителския интерфейс позволява на потребителите лесно да навигират от един екран към следващ, срещайки всички правилни визуални допирни точки по пътя. Добрият дизайн на потребителския интерфейс трябва да даде на потребителите незабавно усещане за комфорт и контрол. Това ще им позволи да научат и овладеят продукт по- бързо и лесно. Дизайнът на Сайтът за фризьорски салон е минималистичен, с малко, но функциониращи елементи. Спазва принципите за добър потребителски интерфейс. Избегнати са сложни елементи, и елементи, които не носят каквото и да е било информация за потребителя. Визуалните обекти в сайта са предназначени да послужат само и единствено за целите на един фризьорски салон. Ето някои основни аспекти на дизайна на *UI* на уеб сайт:

- **Интуитивен дизайн**. *UI* трябва да бъде лесен за разбиране и използване без значително усилие или обучение от страна на потребителя. Менютата, бутоните и елементите на уеб страницата трябва да бъдат подредени логично и интуитивно.

- **Навигация**. Ясната и ефективна навигация е ключова за това потребителите да могат бързо да намерят желаната от тях информация. Менютата, линковете и търсачката трябва да бъдат видими и лесно достъпни.

- **Адаптивен дизайн**. Сайтовете трябва да бъдат адаптивни, така че да изглеждат и функционират добре на различни устройства - компютри, таблети и мобилни телефони.

**- Цветове и изображения**. Цветовете и изображенията са важни за визуалната привлекателност на сайта. Различните елементи на интерфейса трябва да се различават едни от други, например не е добра практика на черен фон да се поставят тъмни текстове и полета.

- **Типография**. Чистият и лесно четим текст е от съществено значение за един уеб сайт. Изборът на подходящи шрифтове и размери е важен за ясно представяне на съдържанието.

- **Функционалност и интерактивност**. Включването на интерактивни елементи като бутони, форми, анимации или слайд шоу може да подобри ангажирането на потребителите и да ги насочи към конкретни действия.

- **Консистентност**. Поддържането на еднакъв стил и изглед на целия сайт, от страница до страница, е важно за създаване на усещане за единен бранд и удобство за потребителите.

Тези аспекти на UI дизайна на уебсайт се комбинират, за да предложат приятно и удобно потребителско изживяване, което води до по-добра ангажираност и удовлетворение от използването на сайта. Някои от елементите на потребителския интерфейс, без които сайта за фризьорски салон няма да може да функционира, както се предполага са:

- **Навигация.** Безспорно най- важният елемент от интерфейса на един уебсайт. Навигацията представлява система от менюта и връзки, които служат за движение на посетителя в даден сайт. Всеки елемент от навигацията визуализира различна страница и носи различна информация. Използването на навигация в уебсайта на фризьорския салон позволява на потребителите да го обиколят с най-голяма ефективност и най-малка некомпетентност. Навигационната система е аналогична на пътната карта, която позволява на посетителите на уеб страниците да изследват и откриват различни области и информация, съдържащи се в уебсайта. Менюто на уеб апликацията се състои от 4 ключови страници. Първата е заглавната страница, където е предоставена информация за различни видове промоции. На второ място, но не по важност е страницата за запазване на часове, където са представени и описани всички видове услуги, които салонът предлага. Следва страницата за контакти, където са споменати социални канали за връзка на потребителя с представител на салона. Последната страница от менюто е тази за информация и статистика. Тя е достъпна само за служители на салона или хора с предоставен достъп. Както стана ясно, всяка страница, според предназначението й, носи различна информация за клиента.

- **Съобщения.** Дизайнът на интерфейса разполага и с множество съобщения, които изскачат след действие на потребителя, с цел улеснение на работата с уебсайта. Има 3 вида съобщения, изникващи при различни обстоятелства – съобщение за грешка, информационно съобщение и съобщение за успех. Съобщението за грешка се визуализира тогава, когато потребителят не е попълнил някое от полетата на сайта, както се предполага. То сигнализира, че нещо не е направено, както трябва. Следващото съобщение, а именно информационното, служи за поднасянето на някакъв вид информация на потребителя. Когато посетителя на сайта срещне съобщение за успех, то това сигнализира, че всичко е наред и след използването на дадена функционалност на платформата, това съобщение уверява че заявката е успешно изпратена до салона и данните му са записани и обработени. Всички използвани съобщения в уебсайта са създадени с идеята, потребителят да е непрекъснато информиран след всяко свое действие. Това изключва вариантът ползвателят да е изгубен в интерфейса и да не знае, какво последващо действие да предприеме.

Всички елементи на уебсайта са подравнени в средата на страницата. Дори, когато страницата бива минимизирана, съдържанието придържа позицията си. Всяка страница от апликацията има наименование, както и заглавие. Заглавието се намира най-отгоре, преди цялостното съдържание. То е значимо и само по себе си описва вида на функционалностите и визуалните елементи в съответната страница. Уеб сайтът разполага и със странична лента, където са разположените навигацията и формите за регистрация и вход. Когато потребителят е успешно достъпил своя профил, то формите за регистрация и вход са заменени с нов елемент, на който се изписва потребителското име на посетителя. За фон на страничната лента е избрано изображение, свързано с тематиката на проекта. Всички елементи в лентата са разположени върху изображението и са със добавен стайлинг, да бъдат леко прозрачни, за да може да не покриват изцяло фона. Шрифтът е внимателно подбран и съобразен с принципите на Типографията. Разположени са множество интерактивни елементи като бутони, графики, таблици, полета и други, които подпомагат потребителя при работата му с уебсайта. Уеб апликацията за фризьорски салон спазва всички принципи заложени в изработването на дизайн на потребителския интерфейс, като по този начин на всички ползватели, от обикновени потребители до служители на студиото, е предоставена възможност за едно приятно изживяване.

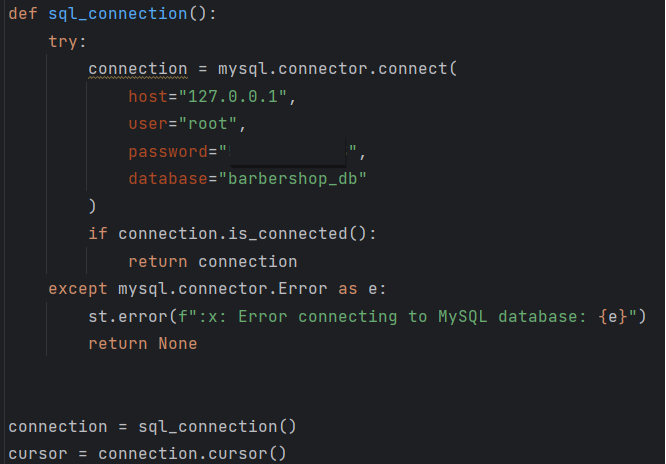
**VI. Програмна реализация**

Програмната реализация представлява процеса на превръщане на проектираните дизайн, архитектура и концепция в реално функциониращ софтуерен продукт. Тя включва изграждането, разработването и имплементацията на софтуерния продукт в съответствие със спецификациите и изискванията, определени по време на проектирането.

Процесът по реализиране на сайта за фризьорски салон се състои от няколко етапа:

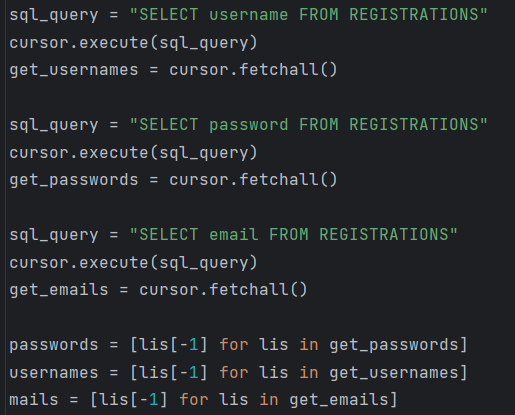
- **Програмиране и разработване. „Бек-енд“**. Този етап дава начало на същинската разработка на сайта. Изискванията са вече определени и е придобита ясна представа за целите и предназначението на апликацията. Така наречената „бекенд“ разработка полага основите на програмната реализация, като се програмират всички функционалности, връзката с базата данни, както и парчетата код, стоящи зад голяма част от визуалните елементи. Бек енд кодът е осъществен благодарение на програмният език *Python,* както и многобройният набор от съвместими библиотеки с него, като например– *Streamlit*.

Фигура 1 -Добавяне на библиотеката, необходима за връзка с базата данни



Фигура 2 -Функция, осъществяваща връзката с базата данни

На *Фигура 1* е визуализирано добавянето на една от най-важните библиотеки за проекта. Чрез нея е възможно осъществяването на връзката с базата данни. На *Фигура 2* пък е показана и функцията, необходима за свързването на апликацията с *MySQL*. Създава се променлива, която приема необходимите параметри на *MySQL* сървъра, като парола, хост, потребител и база. Тялото на функцията е създадено под формата на блок “try” и “catch“, за да може, ако евентуално има проблем с връзката, да изскочи съобщение с грешка, гласящо: „ Неуспешно свързване с базата данни“. След тялото на функцията, тя се извиква и записва в нова променлива. Благодарение на тази променлива се създава курсор, който да изпълнява *SQL* заявки.



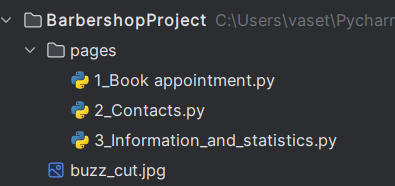
Фигура 3 – Демонстрация на работата с курсор

На *Фигура 3* е демонстрирана работата на курсорът и как той изпълнява заявки. В променлива е записана желаната SQL заявка, като след това тя се изпълнява с командата *cursor.execute*. Резултатите са записани в нова променлива под формата на *list of tuples,* необходимо е последващо действие, за да се вземат само стойностите на съответното поле, за това се изпълнява For цикъл, с който се обхожда целия списък от елементи и се взима само последната стойност от всяка итерация.

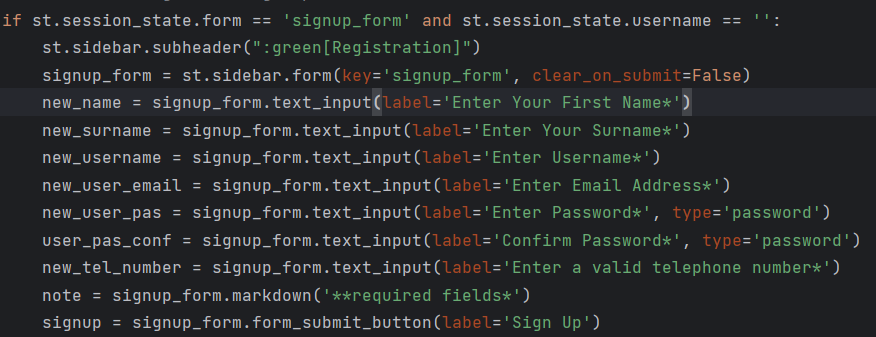
Голяма част от функционалностите на елементите са изработени с библиотеката *Streamlit*. За да може потребителя да използва неговите функции и методи, той трябва първо да се вмъкне в проекта. (*Фигура 4*)



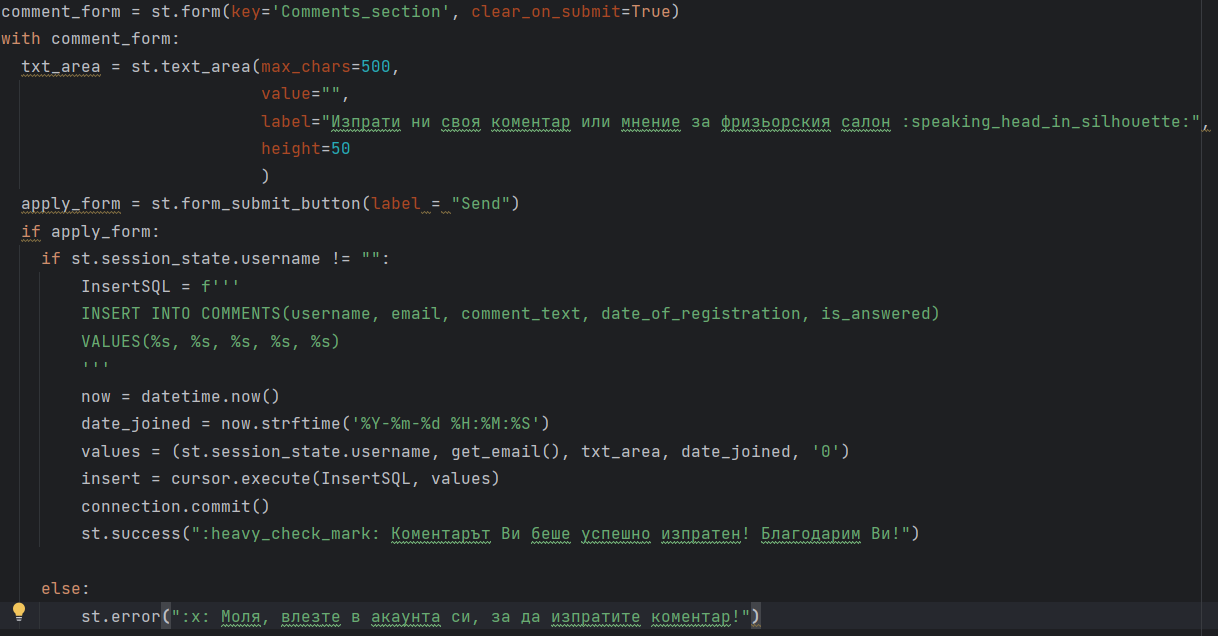
Фигура 4 – Добавяне на библиотеката за Streamlit



Фигура 5 – Python файловете за различните страници на сайта

След като вече програмата има вмъкнат *Streamlit ,* може да се започне с използването на неговите функционалности. На *Фигура 5* са изобразени 3 *Python* файла. Тези файлове представляват страниците на уеб сайта. Благодарение на модерният и иновативен подход към разработка, на *Streamlit* е необходимо само и единствено групирането на тези файлове в директория от проекта наименувана “pages”. По този начин библиотеката разбира, че трябва да разпредели подадените файлове в папката като отделни страници. Кодът във всеки файл се отнася за съответната страница. Числата в наименованието на файловете определят тяхното разположение в навигацията на сайта.

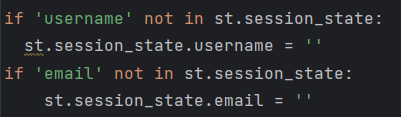
Фигура 6 – Парче код от формата за регистрация на потребители

На *Фигура 6* е визуализирано парче код от функционалността за регистрация на потребителите. Както формата за регистрация така и формата за вход се намират в страничната лента, разположена в лявата част на екрана. Лентата може да се смалява и уголемява, както и може напълно да изчезне. Създадени са текстови полета, които са записани в променливи. Променливите приемат стойността на текста попълнен в съответното поле. Полетата за „Парола“ и „Потвърди паролата“ са различен тип, в сравнение с останалите. Подаден е аргумент ( type = password), с който въведеният текст от потребителя е маскиран със звездички. Има опция за показване на паролата в десния край на полето. *Streamlit* е програмиран по такъв начин, че когато ползвателят взаимодейства с някой интерактивен елемент, то кодът от съответния файл се изпълнява отново. За да се избегнат неприятности съществува *session\_state.* Благодарение на тази функционалност на библиотеката могат да се запазват „сесийни“ променливи, които да запазват своите стойности през различните сесии или при обновяване на страницата и преизпълнение на кода. За да се визуализира формата за регистрация, то е необходимо, както може да се види на *Фигура 6*, „сесийната“ променлива „*form*“ да има стойност „*signup\_form*“, a променливата „username“ да не съдържа никаква стойност. „Сесийната“ променлива с потребителско име трябва да е празна, защото само ако даден потребител все още не е достъпил акаунта си, или няма такъв, то само тогава е възможна опцията за нова регистрация. Относно другата спомената променлива, тя може да приема 2 стойности – „*signup\_form*“ или „*signin\_form*“, като това зависи от бутона който е натиснат – за Регистрация или за Вход.

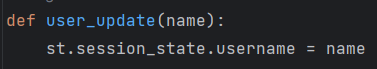
Фигура 7 – Кодът, осъществяващ формата за коментари

Друга функционалност спомената в една от предходните точки е възможността на потребителя да изпраща съобщения директно до представител на фризьорския салон. На *Фигура 7* може да се види, как изглежда кодът зад това поле за коментари. Цялата функционалност се изпълнява във форма – „*st.form*“ предоставена от *Streamlit*. Всяка форма е необходимо да има бутон (*form\_submit\_button*), за да функционира. Декларира се променлива която да съхранява стойността от текстовото поле. Това поле е по- различно от тези в регистрационната форма, поради наличието на различни аргументи. На потребителя е позволено да изпраща съобщения до 500 символа. С натискането на бутона съобщението се изпраща и записва в базата данни, с помощта на *Insert* заявка. Останалите полета, които се записват в таблицата с коментари в *MySql* са потребителското име и имейл на активния потребител, както и датата на коментара. При успешен запис, се визуализира съобщение за успех, с което клиентът разбира, че коментарът му е успешно изпратен и обработен.

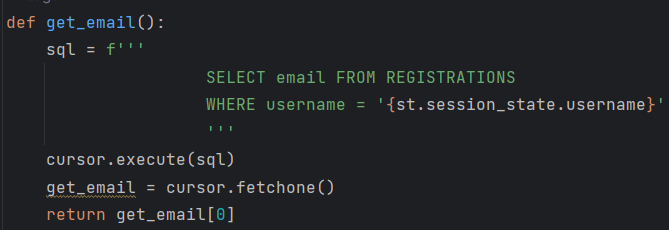
Когато ползвател на сайта достъпи профила си, то неговият мейл и потребителско име се записват в 2 отделни „сесийни“ променливи. На *Фигура 8*  може да се види как се декларират тези променливи:



Фигура 8 – Декларация на сесийни променливи за имейл и потребителско име

След успешен Вход, тези 2 променливи вече не са празни стрингове, а приемат потребителското име и имейла на съответния потребител. Промяната на тези 2 променливи става с помощта на 2 функции, за обновяване на „*сесийна*“ променлива на потребителското име на *Фигура 9* и за обновяване на имейла, съответно на *Фигура 10*.

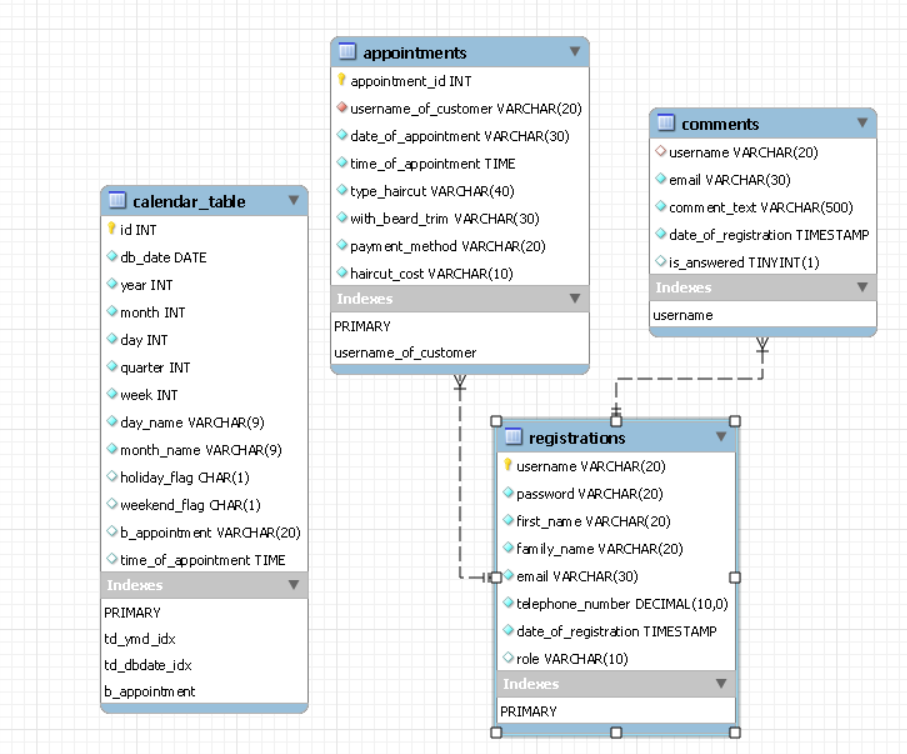
Фигура 9 – Дефиниране на функция за обновяване сесийната променлива на потребителското име



Фигура 10 – Дефиниране на функция за обновяване сесийната променлива на имейла

Фигура 11

Благодарение на тези променливи, както вече стана ясно, потребителят остава в акаунта си, дори след взаимодействие с визуалните елементи, намиращи се на страницата.

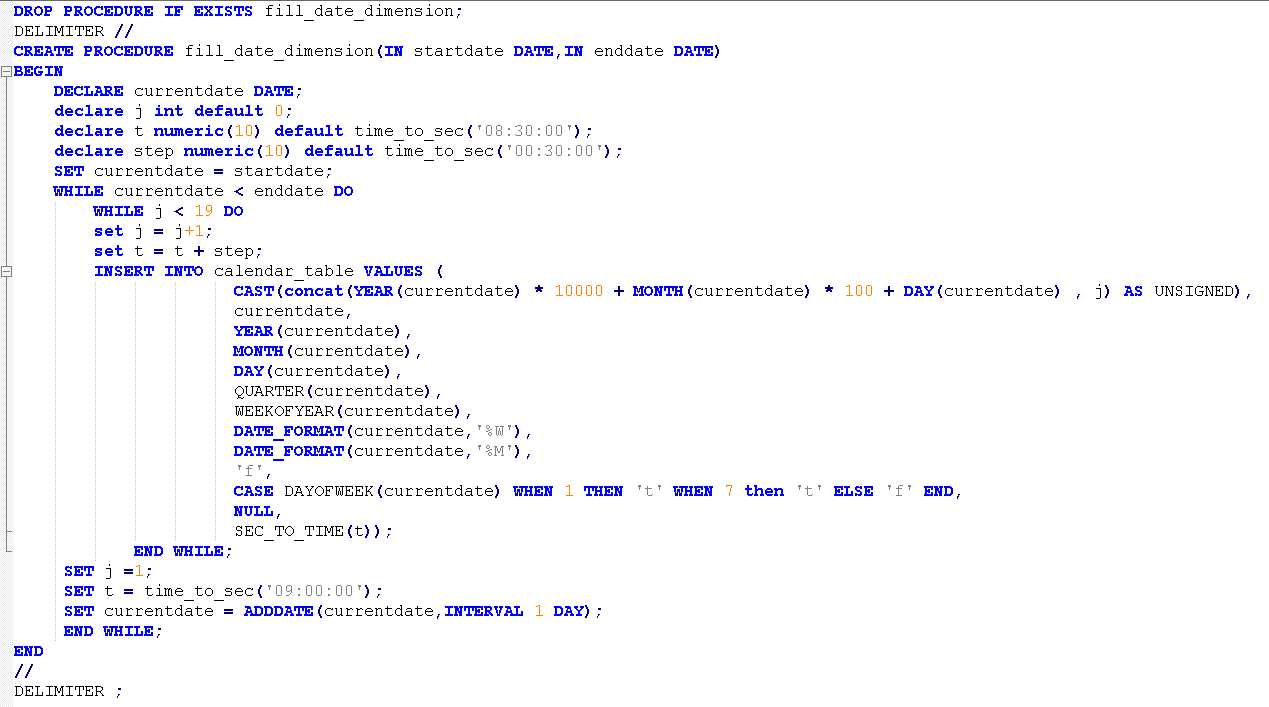
**- База Данни.** „Бек енд“ разработката върви ръка за ръка с разработването на базата данни. За система, с която данните да бъдат управлявани, е избрана *MySql*. В платформата на системата е създаден сървър, който да се достъпва с парола и да изпълнява всякакъв вид заявки и операции. В сървърът е създадена база, която да съхранява обектите, чиито данни използва уеб сайта. Обектите биват таблици, изгледи и процедури.

Фигура 11 - Диаграма с таблици от базата данни

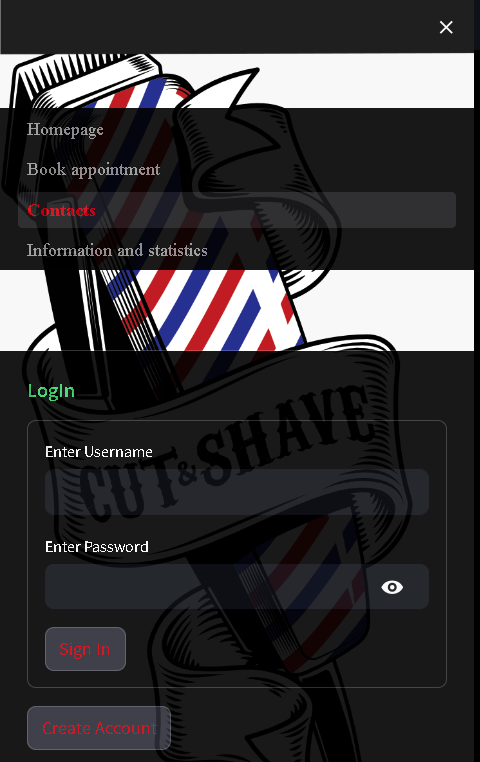
На *Фигура 11* е изобразена диаграма с таблиците налични в базата, техните колони и ключове, както и връзките между тях. Таблицата с данни на регистрирани потребители (*registrations*) разполага с полета за потребителско име, парола, имейл, лично и фамилно име, телефонен номер, дата на регистрация и роля. Съществува първичен ключ на колоната „username“, тъй като не може да съществуват 2 различни потребителя с едно и също потребителско име. По- голяма част от колоните допускат както числови символи, така и текст, но има и такива които приемат само числа, като например телефонният номер. Колоната за дата на регистриране е от тип “*timestamp*” с формат “*YYYY-MM-DD HH:MM:SS*”, тоест дата с час, например - 2023-01-08 00:00:01 (8-ми януари 2023 година, 12часа и 1 секунда). Всяка една колона е от изключително значение, особено тази за роля. Тя определя правата на регистрираните потребители и позволява достъпа до последната страница на уеб апликацията, за статистика и информация. Таблицата с данни от регистрирани ползватели на сайта взаимодейства с таблицата с данни за резервации. Връзката е 1 към много, защото един потребител може да резервира множество часове във фризьорския салон. Таблицата с резервации е съдържа следните полета – идентификатор на резервация, което е и първичен ключ, потребителско име на клиент, което е чужд ключ от таблицата с регистрации, дата на резервация, час, вид на избрана прическа, колона с предпочитания за сайлинг на брадата, метод на плащане и цена. В тази таблица се появяват нови записи, когато клиентите запазват успешно своите резервации. Следващата таблица е тази, съхраняваща информация за коментари на клиенти. Тя има отново има връзка от тип 1 към много с таблицата с регистрации. Това е така, защото един регистриран потребител може да попълва и изпраща множество коментари. Полетата на таблицата с коментари са следните: потребителско име, което играе ролята на чужд ключ, имейл, текст на коментар, дата на регистрация и флаг, показващ, дали даден коментар е отговорен или не. Последната таблица – „calendar\_table“ е от огромно значение за функционалността за запазване на часове, въпреки факта, че няма връзка към останалите обекти. Тя е изградена с целта да наподобява календар. Наличните полета са първичният ключ – идентификатор, текстова презентация на датите , година, месец, ден, четвърт от годината, номер на седмица, наименование на ден от седмицата, следват 2 флага оказващи дали денят е част от уикенд или почивен ден, колона с име на потребител и часове в диапазона от 09:30 до 18:00 часа.

В тази таблица съществуват записи за дати до 31-ви януари 2025 година, като това може да се промени благодарение на процедурата, с която данните се зареждат.

Фигура 12 – SQL процедура, за зареждане на calendar\_table от базата данни

Изобразената на *Фигура 12, SQL* процедура е в основата на записите в таблицата- календар. В началото си тя декларира няколко променливи, които след това да бъдат използвани във двата *while* цикъла. Като входни параметри приема 2 дати, които определят диапазона на записите. Двата цикъла са вложени един в друг, като външният се изпълнява N на брой пъти. Въвеждат се записи в таблицата календар чрез *Insert* заявка, като при всяка следваща итерация датата нараства с един ден. Външният цикъл се извършва докато датата не нарасне до толкова, че да е равна на крайната дата, зададена от втория входен параметър. Вътрешният цикъл обхожда кодът в него точно 18 пъти, колкото са и броят часове, в които могат да запишат своята резервация клиентите, за един работен ден. При всяка следваща итерация часът нараства, а стъпката на нарастване е точно половин час (времетраенето на една подстрижка). Равносметката от процедурата е, че в таблицата-календар се въвеждат N на брой дни, в зависимост от входните параметри, а за всеки ден има по 18 записа с часове в диапазона от 09:30ч. до 18:00ч. през половин час. Двата часа между 12:00 и 13:00 не могат да бъдат резервирани, защото тогава служителите на салона с в обедна почивка. Също така не е позволено да се записват две или повече резервации в един и същи час.

- „**Фронт-енд“ разработка**. Последен от най- важните компоненти при разработката на сайта за фризьорски салон, но не по значение е потребителския интерфейс.

На *Фигура 13* е изобразена страничната лента на сайта. Тя носи своето изключително значение, защото в нея се намират 2 от най- ключовите компоненти на уеб апликацията – навигацията и формите за регистрация и вход. Те са изработени с черен, леко прозрачен фон, с цел лесното открояване на полетата и опциите, но в същото време, изображението поставено като фон не се закрива изцяло. Лентата може да бъде както затваряна от хикса в горния десен ъгъл, така и смалявана и увеличавана. За фон на лентата е използвано изображение характерно за тематиката на сайта. Селектираната опция от менюто свети в различен цвят от останалите, което помага за ориентация на потребителите. Бутоните във формите за регистрация и вход са интерактивни и след взаимодействието на клиента с тях се извършва някакво действие. В случай на Вход( *Sign in*), при въведени правилни данни потребителят достъпва своят акаунт и на мястото на формата за вход изниква ново поле, илюстрирано на *Фигура 14*:

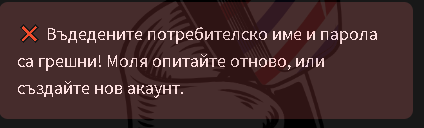
Фигура 13 – Страничната лента на сайта

Фигура 14 – Поле в страничната лента на сайта, визуализирано при успешен вход

Друг наличен елемент на потребителския интерфейс е формата за реклами и промоции. Същността на този елемент е слайд шоу с няколко прикачени изображения. Изображенията се сменят през 4 секунди и се въртят непрекъснато. Кодът зад слайд шоуто, който определя времето за смяна на изображенията, както и последователността на презентиране може да се види на *Фигура 15*:

Фигура 15 – Тялото на Java Script функцията, с която се сменяват изображенията в слайд шоуто

На *Фигура 15* се наблюдава функция реализирана с програмният език *Java Script.* Имплементирана е в тялото на *HTML* код, който определя размерите на рамката, в която се визуализират изображенията, както и самото им импортиране. Останалите визуални ефекти са постигнати със *CSS* код. Относно функцията, тя започва с декларацията на няколко променливи, за слайдовете и за точките отдолу под рамката със снимки. С *for* цикъл се обхождат слайдовете, като на всяка итерация стойността на променливата за слайдове се увеличава с едно. Първият цикъл се извършва непрекъснато, защото когато номерът на слайдът стане по-голям от общия брой, то стойността на променливата се задава отново да е 1. По този начин, когато се сменят всичките изображения и слайд шоуто презентира последното, то не прекъсва презентацията, а напротив, връща се на първоначалното изображение. И така снимките се сменят непрекъснато през период от 4 секунди. Вторият *for* цикъл обхожда точките под рамката със изображения, като променя точката от текущия обход на активна.

- **Тестване и валидация**. След като е изградена програмата, е необходимо да се проведат различни видове тестове (функционални, интеграционни, системни и други). Това се прави с цел сигурност, че софтуерът функционира правилно и отговаря на поставените изисквания. Направени са тестове на всички функционалности на сайта. Проверено е дали те работят както се предполага и дали данните от различните форми се записват успешно и правилно в таблиците в базата данни. Направена е валидация на полетата от формите за регистрация и вход, като всяко поле има своята специфичност и на потребителят не е позволено да въвежда каквото си иска. Когато входните данни от ползвателя не са правилни изкачат съответните съобщения, които го навигират за да може той да се поправи и успешно да завърши процеса, който е започнал.

Фигура 16 – Съобщение за грешка, при въведени невалидни данни за вход

На *Фигура 16* е показан пример за съобщение за грешни потребителски данни при опит за вход във вече съществуващ профил. Това съобщение ще продължи да се показва докато ползвателя не въведе правилна информация, съвпадаща с някой от записите в таблицата с регистрации от базата данни.

- **Отстраняване на грешки** (*debugging*). Ако в процеса на тестване се открият проблеми или грешки, те трябва да бъдат отстранени, за да подобрят стабилността и функционалността на софтуера.

- **Деплоймънт и инсталиране**. След успешното тестване на софтуера, той може да бъде пуснат в производствена среда или да бъде инсталиран на устройствата на потребителите. В случая на сайта за фризьорски салон, въпреки готовия продукт, който е разработен, той все още не е пуснат в производство, а се намира само локално. Това е една от целите на салона за красота, която да се осъществи в близко бъдеще.

- **Поддръжка и оптимизация**. Програмната реализация включва и непрекъснатото поддържане на софтуерния продукт, включително осигуряване на съвместимост с нови версии на операционните системи и използвани технологии, поправка на бъгове и оптимизация на производителността.

**V Ръководство за работа с уеб сайта**

Изключително важно е потребителят на уеб сайта за фризьорски салон да е запознат с наличните функционалности, за да може да използва апликацията пълноценно и без затруднения. Тази точка от документацията има за цел да направи пълен обход на елементите, намиращи се в сайта и да помогне както на обикновения ползвател, така и на администраторите и фризьорите. Служителите трябва да бъдат компетентни по всяка тема, свързана с модернизирането на салона. Те трябва да са на среща при затруднения от страна на клиента като му представят необходима информация и отговорят на неговите запитвания. Клиентът е поставен на първо място при всеки модерен и напредващ бизнес, което ще рече, че клиентското изживяване, използвайки платформата и по време на процедура е от изключително значение за добрия имидж на фризьорския салон.

1. **Инсталация и настройки**

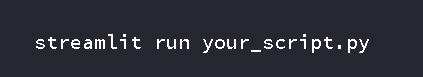
Както вече стана ясно сайтът е проектиран и реализиран на локално ниво, като все още не са предприети действия той да се реализира на производствена среда или да бъде инсталиран на устройствата на потребителите. Това води до няколко настройки, които ползвателят на сайта трябва да предприеме преди достъпването до същинската платформа.

Целият сорс код на проекта, включващ създаването на обектите от базата, Python файловете, описващи различните страници, CSS файлoве, както и използвани изображения, се намират в линкът към “GitHub” предоставен в точка „**VIII Приложения**“. Потребителят трябва да има налични инсталирани актуални версии на следните технологии : *Python*, *MySql*, *Conda*. Благодарение на Conda, която по естество е технология за управление на Python пакети и библиотеки, се създава виртуална среда, която да съхранява версиите на инсталираните пакети. Използваните пакети и библиотеки се намират в папка “*requirements.txt*” в “*GitHub*” проекта.

За създаване на нова виртуална среда е необходимо да се изпълни следната команда в терминала на *Conda*. (*Фигура 17*)

Фигура 17 – Команда за създаване на нова виртуална среда в Conda

На мястото на *„—name myenv*” се изписва желаното име на средата. След натискане на клавишът „*Enter*“ от клавиатурата, командата се изпълнява и средата се създава. След това е необходимо средата да се активира, за да може да се инсталират пакетите в нея. Това става с командата „*conda activate <име на среда>“*. Когато средата е активирана инсталирането на библиотеки може да започне. Всяка една библиотека се инсталира в терминала на *Conda* по отделно с командата „*conda install <име на пакет>*“ или с „ *pip install <име на пакет>.* Когато всичките пакети, описани в “*requirements.txt*” в “*GitHub*” проекта, са инсталирани, то следващата стъпка е стартирането на апликацията от *Conda* терминалът. Виртуалната среда, в която са инсталирани пакетите, трябва да е активна. След което се изпълнява командата „*cd <път към локалната директория>“,* за да може да се достъпи локалната директория, където проектът е запазен. След това, чрез командата изобразена на *Фигура 18* се стартира уеб апликацията.

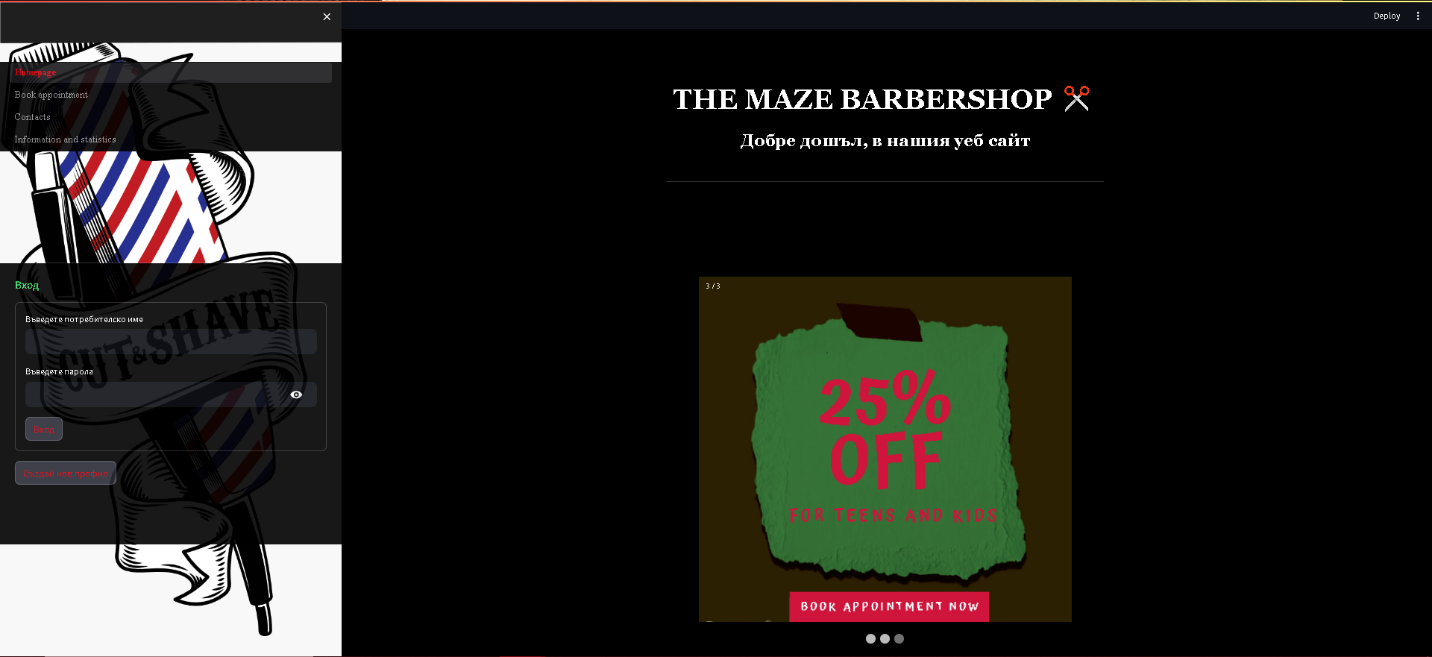
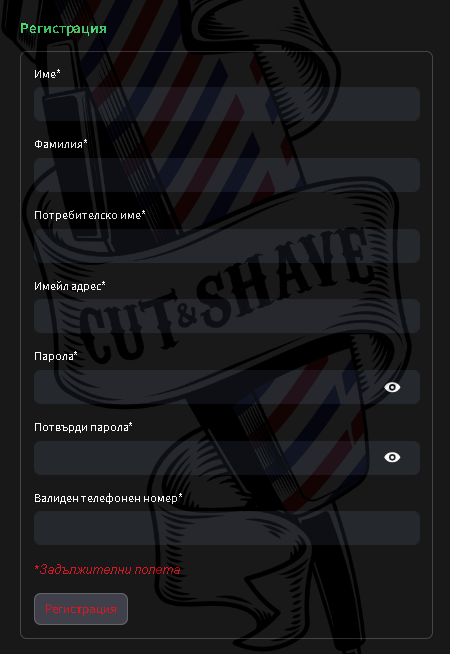


Фигура 18 – Команда за стартиране на уеб апликацията

На мястото на „*your\_script.py*“ от показаната команда на *Фигура 18* се поставя името на съответния файл, който желаем да стартираме. В конкретния случай е необходимо изпълняването на файла „*Homepage.py*“, където се намира кодът за заглавната страница на апликацията. Ако всичките стъпки до тук са изпълнени правилно, в браузъра на потребителя се отваря нова уеб страница, представляваща сайта за фризьорския салон.

Относно базата данни, потребителят трябва да създаде в платформата „*MySQL Workbench*“ нов сървър, с избрани от него име, парола, както и хост. След като сървърът е успешно създаден, следва създаването на база, в която да бъдат създадени и всичките обекти. Кодът необходим за създаването на четирите таблици, с необходимите връзки между тях, изгледа, както и процедурата се намират в линкът към “GitHub” предоставен в точка „**VIII Приложения**“. Те могат да бъдат създадени както заедно, така и по отделно. Препоръчително е обектите да се създават един по един, въпреки че могат и всичките заедно, като се поставя точка и запетайка след всяка операция. На Фигура 2 от „**VI. Програмна реализация“** е показана функционалността на свързване на сайта с базата данни. В полетата за хост, име на потребител, парола и име на базата е необходимо да се попълнят данните от „*MySQL Workbench*“, като тази информация за всеки потребител е различна и персонална. Когато връзката с базата данни е осъществена, то работата с уеб сайта може да започне.

1. **Ръководство за клиента**

След като апликацията е стартирана на екрана се визуализира началната(заглавна) страница. На *Фигура 19* може да се види как тя изглежда и какви са нейните визуални елементи. Страничната лента е разположена от лявата страна на екрана, където се намират и навигацията на сайта, както и под нея формата за Вход и формата за Регистрация. Най- отгоре на страницата са разположено заглавието и подзаглавието. Под тях се наблюдава слайд шоу с изображения, илюстриращо новини, свързани със салона и промоции, от които потребителите могат да се възползват. Ако ползвателят има вече създаден профил то той може да го достъпи през формата за Вход, като въведе данните за своето потребителско име и парола. При неправилни входни данни, се визуализира съобщение с грешка, което уведомява клиента, че е сбъркал и докато не се поправи, входът в неговият профил е невъзможен. Ако ползвателят на сайта няма все още регистриран акаунт, той може да го направи с кликване на бутона “Създай нов профил“. След като бутонът е натиснат в страничната лента, на мястото на формата за вход, се визуализира форма за регистрация. ****На *Фигура 20* е показана същността й. Формата съдържа следните седем полета, необходими за създаване на нов профил. Всичките те са отбелязани като задължителни и е необходимо клиентът да ги попълни, в противен случай, изскача съобщение за грешка, което пояснява за липсата на информация в някои от тях. Някои от полетата имат валидация на въведените данни, като например Паролата не трябва да бъде по-голяма от 20 символа или по-малка от 4. Потребителското име не трябва да бъде по- малко от 2 и по-голямо от 20 символа. Стойността в полето, гласящо „Потвърди парола“, трябва да бъде същата като стойността в „Парола“. При неправилно входни данни на клиента, спрямо изискванията, изскачат съответните съобщения за грешка, поясняващи какво не е на ред. При изцяло правилни входни данни, с натискането на бутона „Регистрация“, профилът е успешно създаден и записан в базата данни. От този момент нататък регистрираният потребител може да достъпва акаунта си само чрез потребителско име и парола. След достъпването на профилът си, началната страница се обновява и ползвателят може да забележи следните промени в интерфейса( *Фигура 21 и 22*)

Фигура 19 – Заглавна(Начална) страница

Фигура 20 – Регистрационна форма

Фигура 21 – Заглавие и подзаглавие на начална страница

Фигура 22 – Надпис в страничната лента, динамично сменящ се спрямо името на клиента

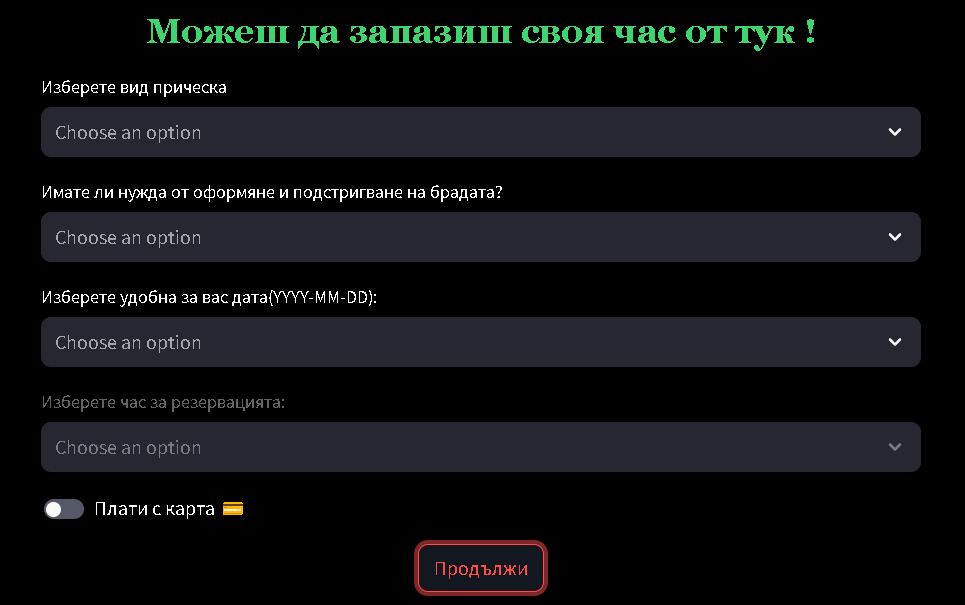
В подзаглавието на Началната страница (*Фигура 21*) се забелязва изписването на потребителското име. Същото явление се наблюдава и на *Фигура 22,* където на мястото на формите за регистрация/ вход се е появила нова форма с текст гласящ „Здравейте“ и името на потребителя, както и бутон за излизане от профила „*Log Out*“.

****Следващата страница от менюто е за Резервиране на час за подстригване. В нея също се наблюдава страничната лента, с характерните за нея визуални компоненти. На върха на страницата се намира Заглавието, което може да бъде видяно на *Фигура 23*.

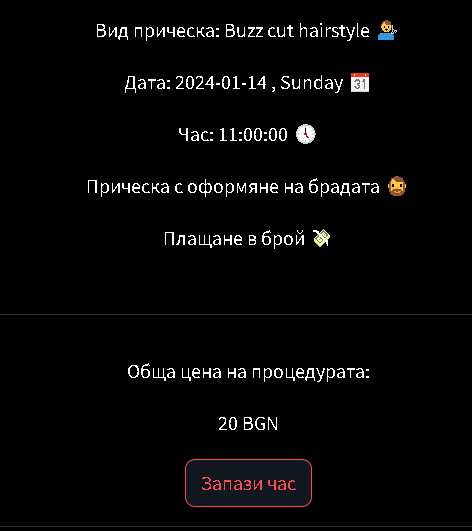
Фигура 23 – Заглавие на втора страница

****Под заглавието има списък с наличните прически, които салонът предлага. Списъкът съдържа наименованието, както и снимка на прическата, кратко описание и цена на услугата в български левове. На *Фигура 24* може да се види как изглежда един елемент от списъка със стилове.

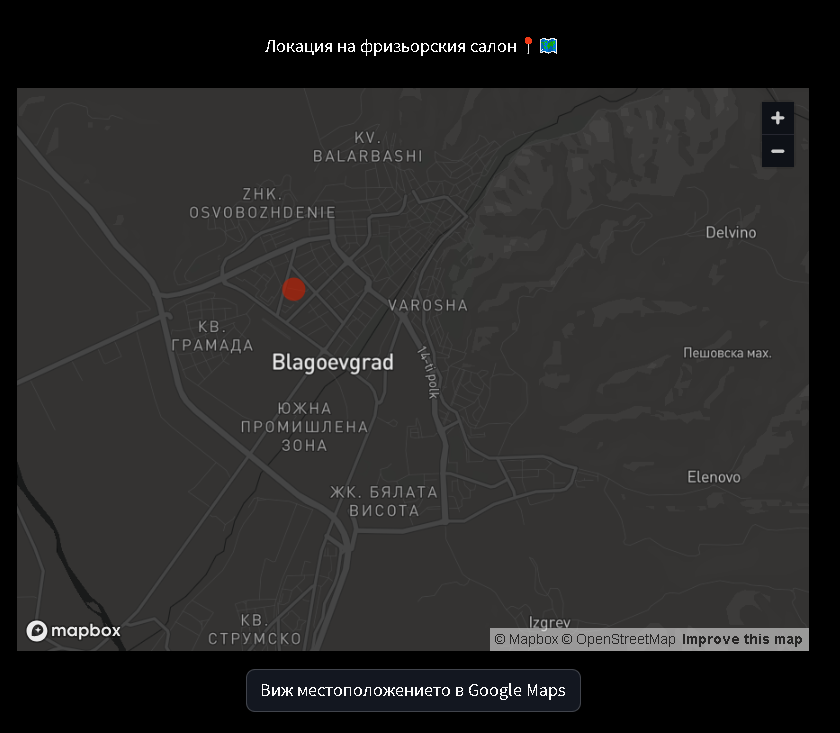
Фигура 24 – Елемент от списъка със стилове и прически

****Под наличните прически, е разположена и същинската форма за запазване на часове. Тя е илюстрирана на *Фигура 25*:

Фигура 25 – Форма за записване на резервации

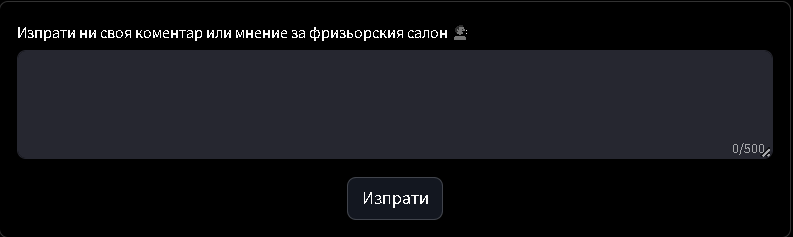
****На клиентът са предоставени опции за вид на прическа, предпочитание за брада, избор на дата и час. За избор на прическа са налични седемте стила описани в списъка над елемента за резервиране на часове. Опциите във второто поле са в зависимост, дали клиентът иска подстригване и оформяне на брадата или не. В следващия елемент е предоставена възможност за избор на удобен ден и час. В полето с часове се визуализират само свободните, незаети от други клиенти на салона. Съществува и опция за заплащане на процедурата с карта, като ако тя не е избрана, се предполага че клиентът ще плати в брой. Ако, обаче, е направен избор „Плати с карта “, то се визуализират допълнителни полета за номер на картата, трицифрен код и дата на валидност. С натискането на бутона „Продължи“, потребителят получава кратко обобщение на избраните от него опции за резервация. Обобщението може да се види на *Фигура 26,* като са избрани от полетата тестови данни. Наблюдават се всички предпочитания избрани от клиента, като под тях е калкулирана и общата цена на услугата. С натискане на бутона с долния край, потребителят запазва своя час и данните се обработват и записват в базата данни. Ползватели на сайта, които не са достъпили профилът си, не могат да резервират часове, като изскача съобщение за грешка „Моля, влезте в профила си, за да резервирате час“. Ако всичко е наред то съобщението е за успех и гласи „Резервацията е успешно записана! Благодарим Ви!“.

Фигура 26 – Обобщение на резервация

****Последната страница, достъпна за използване от клиентите на салона, е страницата за Контакти. Тя отново започва със заглавие, а под него са изброени социалните канали на фризьорския салон, по които клиентите могат да се свържат с представител. След това е илюстрирана карта, показваща локацията на салона. Картата може да се види на *Фигура 27*.

Фигура 27 – карта с точна локация на салона

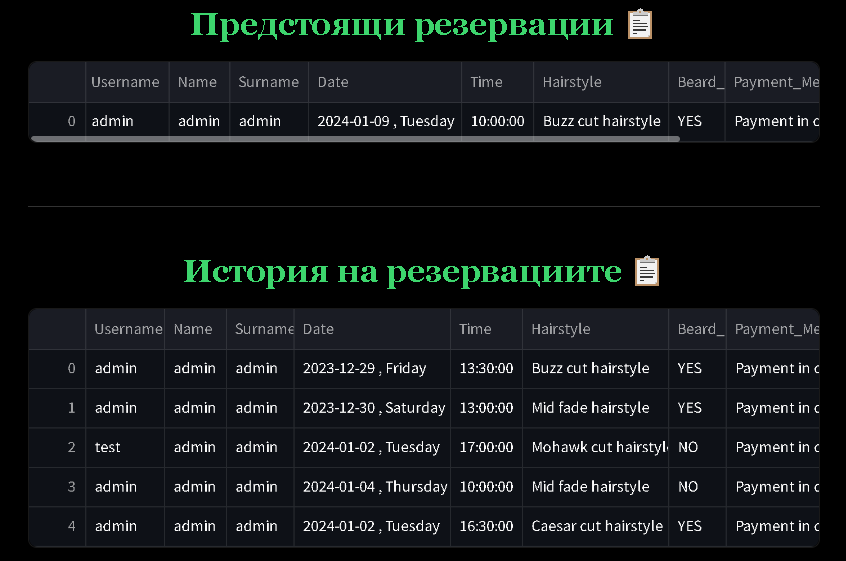
Картата е интерактивен елемент от интерфейса на апликацията. Тя може да бъде приближавана, отдалечавана, с двата бутона „+“ и „-“ в горния десен ъгъл, както и презентирана на цял екран. С червената точка се обозначава точното местоположение на салона. Ако клиентът не може да разбере къде е разположен, то той може да използва бутона „Виж местоположението в Google Maps“. След взаимодействието с бутона, се отваря нов прозорец в браузъра, показващ местоположението на салона на карта, предоставена от “Google Maps”. Където се поддържат имена на улици, различни видове изгледи на картата и много други опции, които ще ориентират клиента към правилното място.

Последният елемент от страницата с контакти е формата за изпращане на съобщение или коментар директно до представител на салона. Тя може да се види на *Фигура 28*.

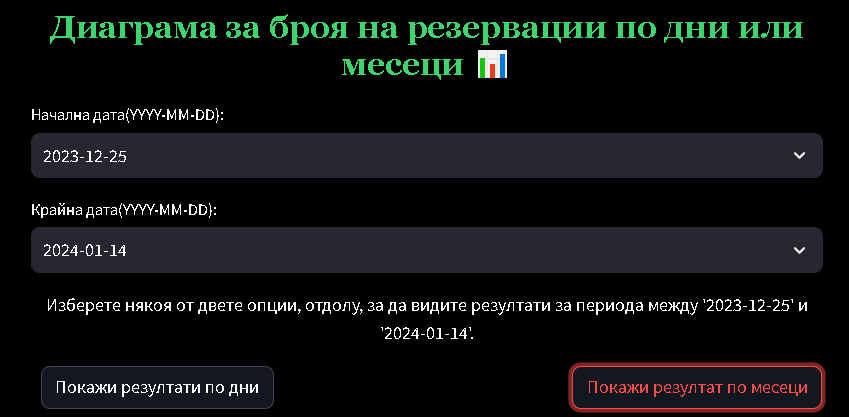
Фигура 28 – Форма за изпращане на коментари и съобщения

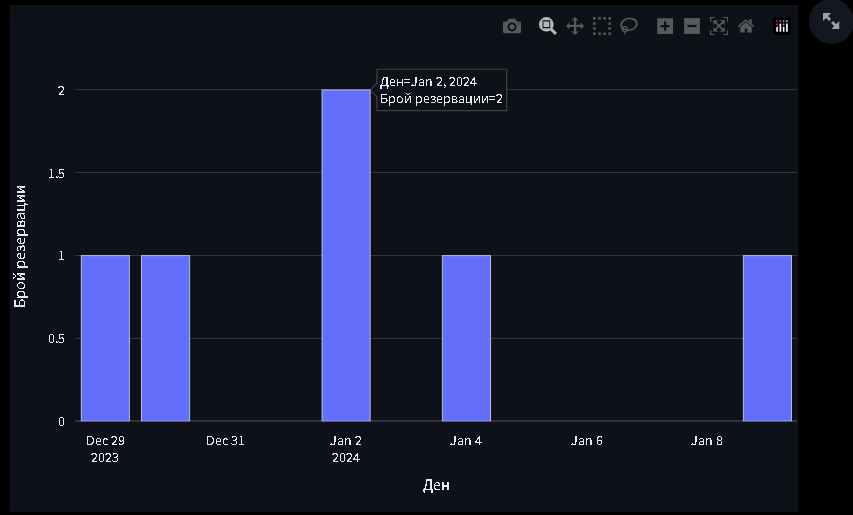
Формата за коментари представлява едно голямо текстово поле в което потребители могат, общо взето, да пишат каквото си поискат – коментари, мнения за салона и сайта, споделяне на впечатления от преживяването след процедура, благодарности и много други. Максималната допустима дължина на текста е 500 символа. С кликване на бутона „Изпрати“, съобщението се изпраща и на екрана под формата се визуализира съобщение, уверяващо ползвателя, че неговия коментар е успешно изпратен и обработен. В противен случай, ако например потребителя не е влязъл в профила си, изниква съобщение за грешка, което моли клиентът да достъпи първо профилът си преди да изпраща какъвто и да било текст чрез формата за коментари.

1. **Ръководство на служителя**

Служителите на фризьорския салон влизат в ролята и на администратори. Те имат достъп до всеки един компонент на уеб апликацията. В това число се включва и последната страница, а именно тази за статистика и информация, до която обикновения ползвател не може да достъпи. В ръководството на потребителя са вече описани първите страници на уеб сайта, за това в тази точка, се набляга единствено на описанието на последната. Както останалите, така и страницата за статистика и информация започва със необходимото заглавие. След това са разработени 5 визуални компонента, носещи статистическа информация относно потребителите на салона , като информация за направени коментари от формата за коментари и съобщения (*Фигура 28*) и информация за регистрирани потребители. Следващите 2 компонента предоставят информация, следователно за изминали резервации(история на резервациите) и предстоящи резервации.

Фигура 29 – Форми, предоставящи информация за изминали и предстоящи резервации

На *Фигура 29* са илюстрирани двете форми за изминали и предстоящи резервации. Данните в тях са извлечени директно от базата данни и са под формата на таблици. Информацията в тези таблици не може да се коригира и служи само за информативни цели. Колоните описват изцяло уговорките, като е налична опцията за увеличаване на размера им и изглед на таблиците на цял екран. На база на резервираните часове е разработен и следващият, последен елемент на страницата. Той може да се види на *Фигура 30*. Забелязват се две падащи менюта с множество дати. Потребителят трябва да избере Начална и Крайна дата, описващи желания период, за когото да се визуализира диаграмата с резултати. Началната дата не може да бъде по-голяма от крайната. След като служителят е направил своя избор за период, той има 2 опции – да симулира диаграмата с данни по дни или по месеци. Именно за това са и двата бутона. След натискането на някой от тях се визуализира съответната диаграма отдолу под бутоните. Диаграма с резултати по дни, за избрания ****период от *Фигура 30*, е визуализирана на *Фигура 31*.

****

Фигура 31 – Лентова диаграма, показваща информация относно броя на резервациите по дни

Фигура 30 – Селектор на начална и крайна дата (перииод от дати) за диаграмата

На оста Х на диаграмата са разположени дните, а на оста У, броя на резервациите. Когато потребителят постави курсора на мишката си върху някоя от лентите, се визуализира информация за броя на резервациите в съответния ден. Диаграмата е интерактивна и както може да се забележи, в нейния горен десен ъгъл има множество опции. Диаграмата може да се уголемява на цял екран, може да се приближава и отдалечава, дори е налична опцията за записване на диаграмата като снимка в локална директория.

**VI Заключение**

Модернизирането на фризьорските салони е ключово за тяхната конкурентоспособност на пазара и предоставяне на по-добри и иновативни услуги на клиентите. Автоматизацията на процесите и услугите, които един салон предлага, чрез разработването на уеб сайт, е основен елемент за развитието му. Предоставянето на удобство и приятно изживяване на клиентите чрез уеб платформата се осъществява посредством съвременен дизайн и функционалности, които оптимизират взаимодействието с потребителите. Въпреки простия си дизайн и ограничената функционалност, сайтът е готов да се конкурира с останалите уеб платформи за фризьорски салони на пазара. Непрекъснатата поддръжка и отстраняването на грешки и дефекти са от ключово значение за постоянното подобрение на сайта. Има редица идеи за подобрение на салона, които в близко бъдеще ще бъдат осъществени. Една от тях е реализирането на системата в производствена среда или инсталирането й на устройствата на потребителите, което ще позволи по- широк достъп до сайта и популяризация сред подрастващи и млади хора, които непрекъснато са в интернет пространството. Друга идея за разширяване на функционалностите на уеб апликацията е добавянето на нови удобства. Те могат да бъдат онлайн магазин с продукти за грижата на косата и брадата за мъжете, услуги за жени, като маникюр, педикюр и каталог от модерни прически и стайлинги. Допълнителни селекция за прически при мъжете и опция за избор на фризьор, при когото да бъде резервирана уговорката са също част обмисляните подобрения. С навлизането на Изкуствения интелект през последните няколко години, много конкуренти се възползват от предимствата, които предлага. За това идеята за внедряване на онлайн асистент в уеб платформата е в обхвата на проекта за бъдещо развитие. Асистентът ще бъде разработен на база изкуствен интелект и ще помага на потребителите при използване функционалностите на сайта и ще е наличен за запитвания и консултации.

**VII Използвана литература**

1. <https://docs.streamlit.io/>
2. <https://bg.wikipedia.org/wiki/%D0%A3%D0%B5%D0%B1%D1%81%D0%B0%D0%B9%D1%82>
3. <https://bdg.bg/glossary-term/%D0%BD%D0%B0%D0%B2%D0%B8%D0%B3%D0%B0%D1%86%D0%B8%D1%8F-%D0%BD%D0%B0-%D1%81%D0%B0%D0%B9%D1%82%D0%B0/>
4. <https://studioweb.bg/post/top-principi-za-dizayn-na-potrebitelskiya-interfeys-9092>
5. <https://uweb.bg/tag/%D1%81%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0-%D0%BD%D0%B0-%D1%83%D0%B5%D0%B1-%D1%81%D0%B0%D0%B9%D1%82/>
6. <https://aula.bg/blog/html-css/>
7. <https://bg.wikipedia.org/wiki/Python>
8. <https://websitedesign.bg/blog/arhitektura-na-sayt>
9. <https://savovdesign.com/etapi-pri-izrabotka-na-ueb-sait>
10. <https://help.superhosting.bg/what-is-a-website.html>
11. <https://bg.wikipedia.org/wiki/MySQL>
12. <https://dev.mysql.com/doc/>
13. <https://docs.python.org/3/>

**VIII Приложения**

[vkolevv23/WebSiteProject: Web Site developed with Python and Streamlit (github.com)](https://github.com/vkolevv23/WebSiteProject)