

proposal valid to: 02.25.2022



proposal issued: 12.22.2021



Taghche company Business

IMPLEMENT THE INITIAL VERSION



درباره ما:

گام هفت پروژه تحلیل و طراحی سیستم ها

استاد درس: دکتر حسن بشیری

دستیار آموزشی: آقای امیرحسین باباییان

دانشگاه: صنعتی همدان

موضوع: پیاده سازی نسخه اولیه طاقچه

گروه Runtime terror

نام اعضا: محدثه پرویزی، مهسا سیروزیان، مریم شمالی، فاطمه معصومی

طاقچه، نزدیکترین کتابفروشی شهر

فهرست مطالب:



توضیحات گام هفتم

4	User: کلاس
4	Customer Page: کلاس
4	Author Page: کلاس
4	User Profile: کلاس
5	Marked Book: کلاس
5	Electronic Book: کلاس
5	CustomerAccount : کلاس
5	authorProfile : کلاس
6	authorAccount : کلاس
6	likeAComment: کلاس
7	dislikeAComment: کلاس
7	markAudioBookForPocket: کلاس
7	markForPocket: کلاس
7	Payment: کلاس
7	showCategoryOfBook: کلاس
8	deleteProduct: کلاس
8	my books: کلاس
8	Random Number: کلاس

توضیحات گام هفتم

در این گام ابتدا جلسه ای برای صحبت راجع به نحوه انجام کار تشکیل شد و تقسیم کار انجام شد. پس از آن جلسه دیگری برای رفع ابهامات تشکیل گردید. پس از مدتی، جلسه ای برای انتخاب دیتابیس مورد نظر و بلافاصله بعد از آن جلسه ای دیگر برای شرح جداول دیتابیس طراحی شده و نحوه کار با آن گذاشته شد. بعد از دادن توضیحات لازم راجع به دیتابیس، هر فرد وارد بخش کد زنی خود شد و روی آن کار کرد. در نهایت جلسه ای برای سرهم کردن کد های زده شده توسط افراد گروه و جمع بندی دیتابیس و همگام سازی داده های اولیه برگزار شد.

در ادامه توضیح برخی از کلاس ها و متدهای نهایی آورده شده است:

کلاس User:

این کلاس شامل یک متد منو برای نمایش منو و انتخاب نقش کاربر است.

کلاس Customer Page:

در این کلاس توانایی ورود با شماره تلفن و ایمیل و همچنین ثبت نام با شماره تلفن و ایمیل به کاربر داده شده است.

کلاس Author Page:

در این کلاس توانایی ورود با شماره تلفن و ایمیل و همچنین ثبت نام با شماره تلفن و ایمیل به ناشر داده شده است.

کلاس User Profile:

این کلاس شامل بخش هایی از جمله دیدن کتاب های قفسه (کتاب های خریداری شده)، افزایش اعتبار و... می باشد.

کلاس **Marked Book**:

در این کلاس، کتاب های نشان شده نمایش داده میشوند و همچنین قابلیت دیدن جزئیات اطلاعات یک کتاب وجود دارد.

کلاس **Electronic Book**:

در این کلاس قابلیت دیدن جزئیات یک کتاب، دیدن سایر کتاب های نویسنده کتاب و.... آورده شده است.

کلاس **CustomerAccount**:

در این کلاس یک متد menu داریم که شامل 2 گزینه نمایش پروفایل و ادیت پروفایل است.

همچنین یک متد `setCustomerID` داریم که ایدی مشتری را که در قسمت لاگین از دیتابیس گرفتیم برای استفاده در این دیتابیس در متغیری به اسم `customerID` ذخیره می کند تا در متدهای دیگر استفاده شود.

در متد `showProfile` چهار متغیر ایمیل، رمز عبور، نام کاربری و موجودی حساب را ایجاد کردیم. سپس به دیتابیس متصل می شویم و از طریق `query` به جدول مشتریان متصل می شویم و تمامی رکوردهایی را که تحت عنوان `customerID` ذکر شده وجود دارند را دریافت می کنیم و درون متغیرهای ایجاد شده میریزیم. سپس آن ها را چاپ می کنیم.

در متد `editProfile` نیز تمامی مشخصات حساب کاربر را دریافت می کنیم سپس تحت یک `query` تمامی رکوردهای قبلی که با آن `customerID` ست شده اند را آپدیت می کنیم.

کلاس `AuthorAccount` نیز همانند کلاس `CustomerAccount` است، فقط در متد های `showProfile` و `editProfile`، به جای متغیر `CustomerID` متغیر `authorID` را تعریف کردیم و در `query` ها اطلاعات از جدول `authorAccount` خوانده می شوند.

کلاس **authorProfile**:

این کلاس نیز دارای یک متغیر `authorID` است که هنگام `login` ست می شود.

3 متد `showMyBooks`، `editBook` و `deleteBook` نیز وجود دارند که به ترتیب عملیات های زیر را انجام می دهند:

1- نمایش کتاب های منتشر شده توسط نویسنده

2- ویرایش اطلاعات کتاب های منتشر شده توسط نویسنده

3- حذف کتاب های منتشر شده توسط نویسنده

در متد showMyBooks تمامی رکورد های ثبت شده در جداول textual book و audio book که شامل اسم کتاب، نام ناشر، قیمت، امتیاز، سال انتشار و... میشوند را تحت عنوان authorID در جداول جست و جو می کنیم و اطلاعات آنها را چاپ می کنیم.

در متد editBooks نیز ابتدا نوع کتاب از نوع صوتی یا متنی را جهت ویرایش انتخاب میکنیم، سپس لیستی از تمامی کتاب های منتشر شده توسط این ناشر را چاپ می کنیم. سپس ID کتاب مورد نظر را جهت ویرایش، از کاربر دریافت می کنیم. در ادامه تمامی اطلاعات جدید جهت ذخیره در دیتابیس را از کاربر دریافت می کنیم و تحت یک query اطلاعات داخل جدول textulBook یا audioBoo را که تحت عنوان این منتشر کننده وجود دارند آپدیت می کنیم.

در متد deleteBook نیز لیست تمامی کتاب های کاربر را نمایش می دهیم سپس id کتاب مورد نظر جهت حذف را از کاربر دریافت می کنیم. سپس کتاب مورد نظر را از جدول کتاب ها حذف می کنیم.

کلاس authorAccount :

این کلاس شامل متد showprofile است که اطلاعات مربوط به ناشر را از دیتابیس بر اساس آیدی میگیرد و نشان میدهد.

متد editprofile وظیفه ی ویرایش اطلاعات حساب کاربری مربوط به یک ناشر را دارد، که پس از تغییر آنها را ذخیره میکند.

کلاس likeAComment:

در این کلاس کاربر می تواند یک کامنت را لایک کند، با این کار تعداد لایک های کامنت مورد نظر در دیتابیس یک عدد اضافه می شود. در ابتدای کد تعداد لایک های کامنت خوانده می شود سپس یکی به آن اضافه می شود و دیتابیس در ادامه کد آپدیت می شود. ورودی کلاس آیدی کامنتی ست که می خواهیم آن را لایک کنیم.

کلاس dislikeAComment:

در این کلاس کاربر می تواند یک کامنت را دیسلایک کند، با این کار تعداد دیسلایک های کامنت مورد نظر در دیتابیس یک عدد اضافه می شود. در ابتدای کد، تعداد دیسلایک های کامنت خوانده می شود سپس یکی به آن اضافه می شود و دیتابیس در ادامه کد با مقدار جدید آپدیت می شود. ورودی کلاس آیدی کامنتی ست که می خواهیم آن را دیسلایک کنیم.

کلاس markAudioBookForPocket:

در این کلاس نشان گذاری کتاب صوتی انجام می شود، با انتخاب گزینه نشان کردن در پروفایل کتاب، وارد این کلاس می شویم و ورودی کلاس متغیر های bookID (آیدی کتابی که کاربر می خواهد آنرا نشاندار کند)، costumerID (آیدی کاربری که می خواهد کتاب را در پروفایل خود به عنوان نشان ذخیره کند) و bookName (نام کتاب مورد نظر) هستند و پس از اجرای کد، اطلاعات کتاب و کاربر مورد نظر در دیتابیس ذخیره خواهد شد.

کلاس markForPocket:

همانند کلاس markAudioBookForPocket عمل می کند، با این تفاوت که کتاب الکترونیکی را در نشان شده های کاربر ذخیره می کند، نه کتاب صوتی.

کلاس Payment:

همان درگاه پرداخت است که کاربر توسط آن کد تخفیف را در صورت وجود وارد می کند، پس از اعمال کد تخفیف، اطلاعات خرید در دیتابیس ذخیره می شود و کتاب به کاربر اضافه می شود. ورودی های کلاس شامل productID ، costumerID ، priceInRial ، productType است که آیدی محصول، آیدی خریدار، اینکه آیا به ریال پرداخت می شود یا خیر و نوع محصول را شامل می شوند.

کلاس showCategoryOfBook:

دسته بندی ای که کتاب در آن قرار دارد را نمایش می دهد، ورودی کلاس آیدی کتاب است تا با استفاده از آن دیتابیس جستجو شود و دسته بندی مربوط به کتاب پیدا و چاپ شود.

کلاس deleteProduct:

در این کلاس کار حذف یک کتاب از دیتابیس انجام می شود، اطلاعات یک کتاب در چندین جدول از دیتابیس وجود دارد، بنابراین در تک تک آنها حذف خواهد شد. ورودی کلاس آیدی کتاب است که با داشتن آن می توان اطلاعاتش را در هر جدولی که موجود است حذف نمود.

کلاس deleteProduct:

این کلاس کامنت هایی که کاربر لایک کرده را با اطلاعات جزئی نمایش میدهد. ورودی کلاس تنها آیدی کاربریست که در دیتابیس دنبال پست های لایک شده توسط او می گردیم.

کلاس my books:

شامل متد های:

Show book shelf: این متد قفسه های کتاب را نمایش میدهد

Create Shelf: قفسه کتاب جدیدی میسازد

Open Shelf: قفسه های موجود و کتاب های داخل آن را نشان میدهد

Delete: قفسه موجود را حذف میکند

Add Comment: برای کتاب مربوطه نظری گذاشته میشود از طریق کاربر و به دیگران نمایش می دهد

کلاس Random Number:

کار ایجاد اعداد تصادفی برای آیدی ها را انجام میدهد، با دادن تعداد ارقام به این کلاس می توان عددی رندوم با آن تعداد رقم ایجاد کرد.