

به نام خدا
دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)
دانشکده مهندسی کامپیوتر و فناوری اطلاعات



طرح پروژه

درس مهندسی نرم افزار ۱

موضوع پروژه: طراحی سایت رفع نیاز های دانشجویی (sneeds)

نام گروه: احسان و شرکا

اعضای گروه:

۱- عرفان رحیمی زاده

۲- سید محمد هادی حسینی

۳- احسان سوری

۴- آریا خلیق

استاد درس: دکتر کلباسی

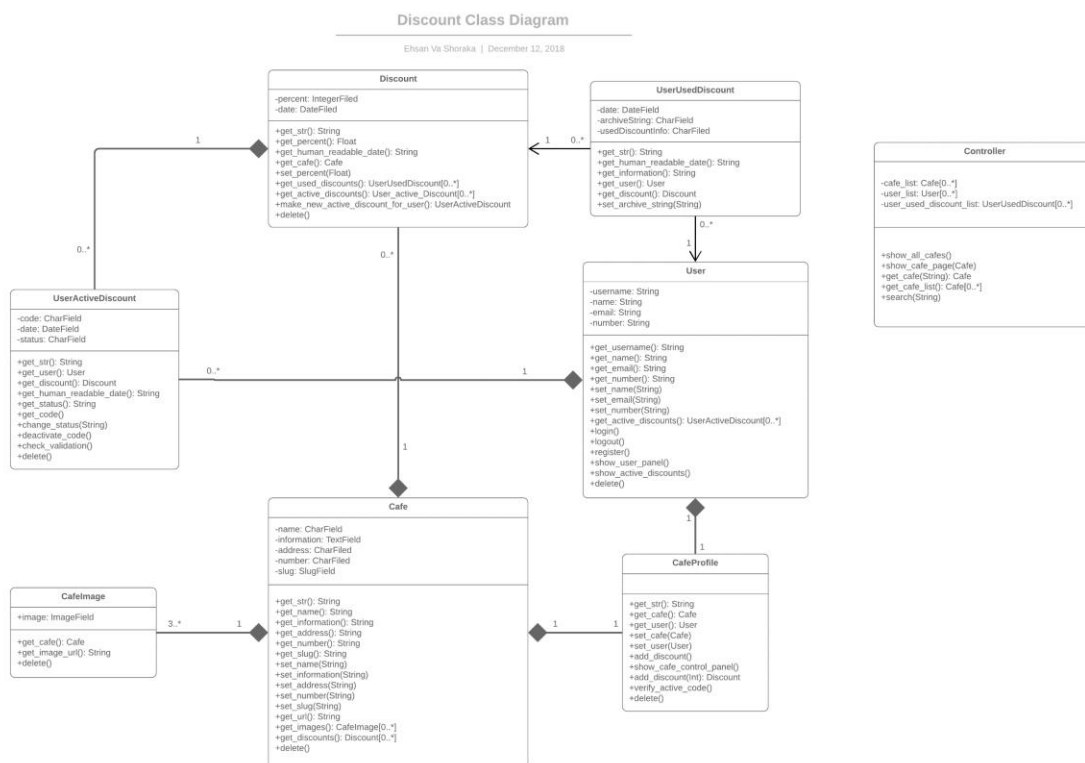
تدریسار درس: خانم مهتاب عزتی

نکته : عکس هامون رو هر کاری کردیم نمیشد واضح تر روی word قرار داد واسه همین میتونید

عکس های واضح ترو از لینک زیر ببینید.

[https://github.com/sNeeds/System-Software-Design/tree/master/Class Diagram](https://github.com/sNeeds/System-Software-Design/tree/master/Class%20Diagram)

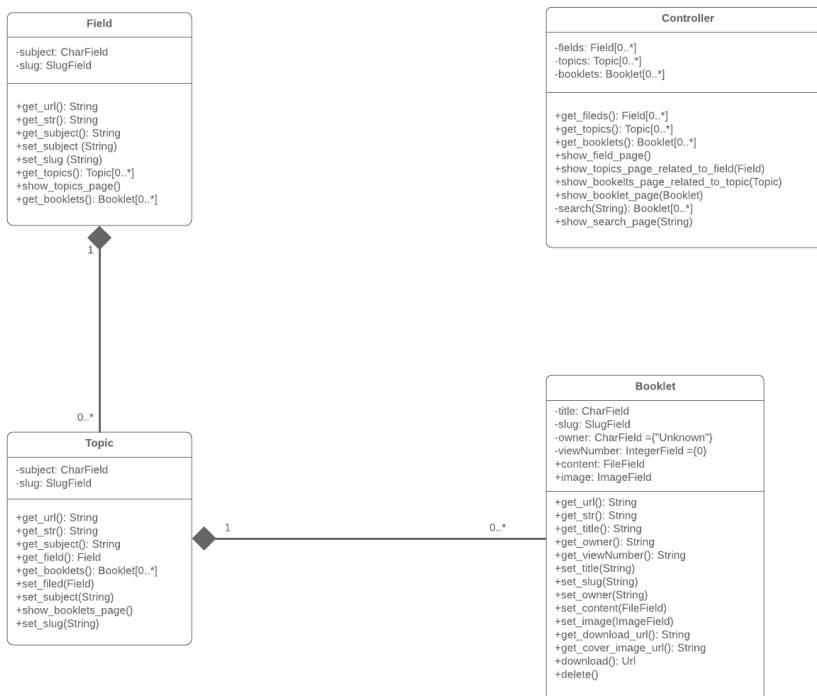
کلاس دیاگرام برای قسمت کافه



کلاس دیاگرام برای قسمت جزوه

[Booklet Class Diagram]

Ehsan Va Shoraka | December 12, 2018



توضیح کلاس های نمودار:

Cafe

کلاس کافه دارای اسم و مشخصات یک کافه میباشد(این کلاس به عنوان یک کاربر عمل نمی کند).

Discount

کلاس نشان دهنده یک تخفیف است که هر شی از این کلاس مربوط به یک کافه می باشد. مثلا تخفیف ۲۰٪
یک کلاس است که مربوط به کلاس "کافه آریا" است .

UserActiveDiscount

این کلاس در واقع برای ساختن اشیای کد تخفیف برای کاربران استفاده می شود . مثلا یک نمونه از این شی
میتواند "کد تخفیف ۲۰٪ با کد "g7r6 ۴۴" برای یک کاربر باشد.

UserUsedDiscount

این کلاس برای نگه داری اشیا کلاس **UserActiveDiscount** استفاده میشود. در واقع شی کلاس
UserUsedDiscount حذف میشود و یک شی از این کلاس ساخته میشود . اشیای این کلاس به عنوان
آرشیو استفاده میشوند .

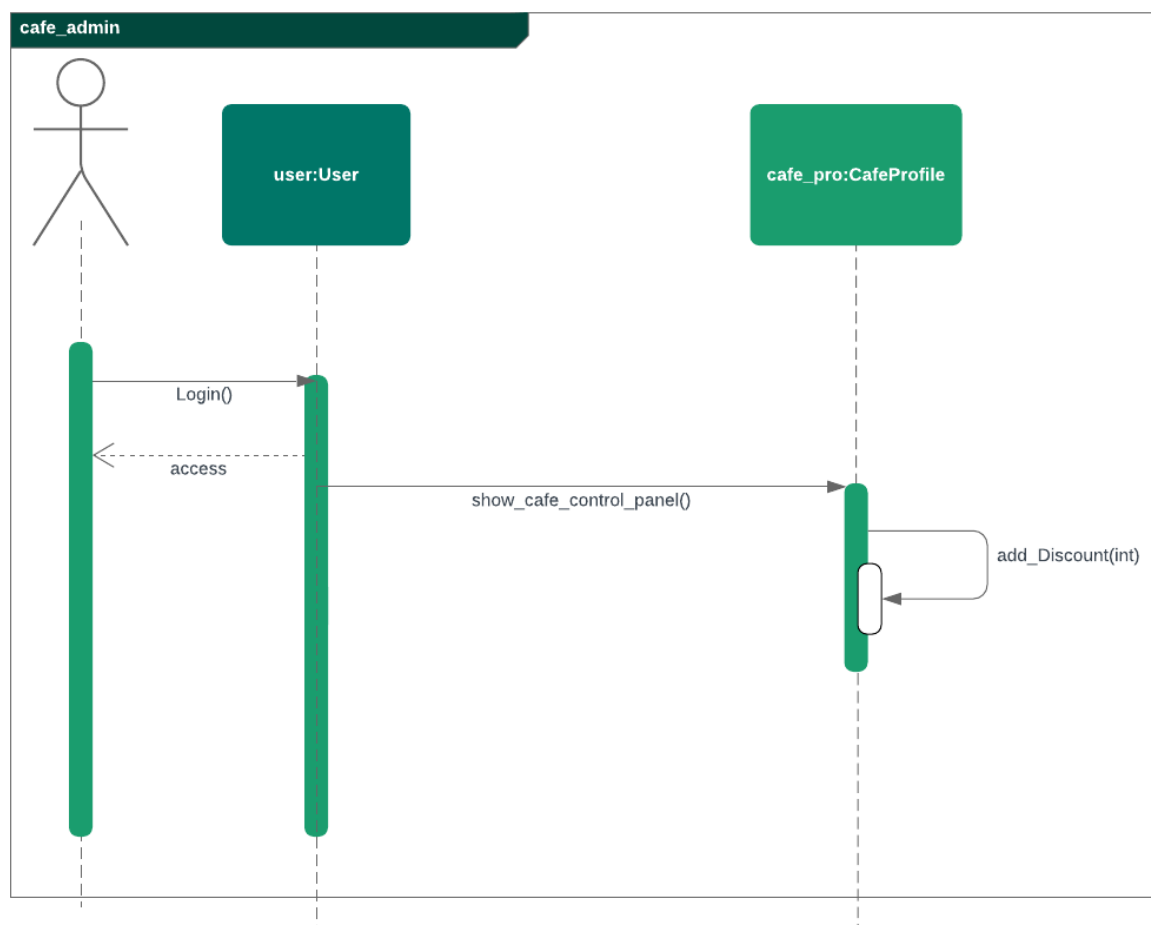
Controller

به عنوان یک کلاس اصلی (main) وجود دارد و اشیای کلاس های دیگر را تولید و نگهداری و مدیریت می کند و برای مثال صفحات مختلف که چند شی باید در آن نمایش داده شوند را مدیریت میکند و لیستی از اشیا را در خود نگه میدارد. در حقیقت اشیای دیتابیس به نوعی در این کلاس و به عنوان یک list نگه داری میشوند .

1. اضافه کردن تخفیف توسط ادمین کافه

Add_Discount

[Ehsan vs Shoraka] | December 12, 2018



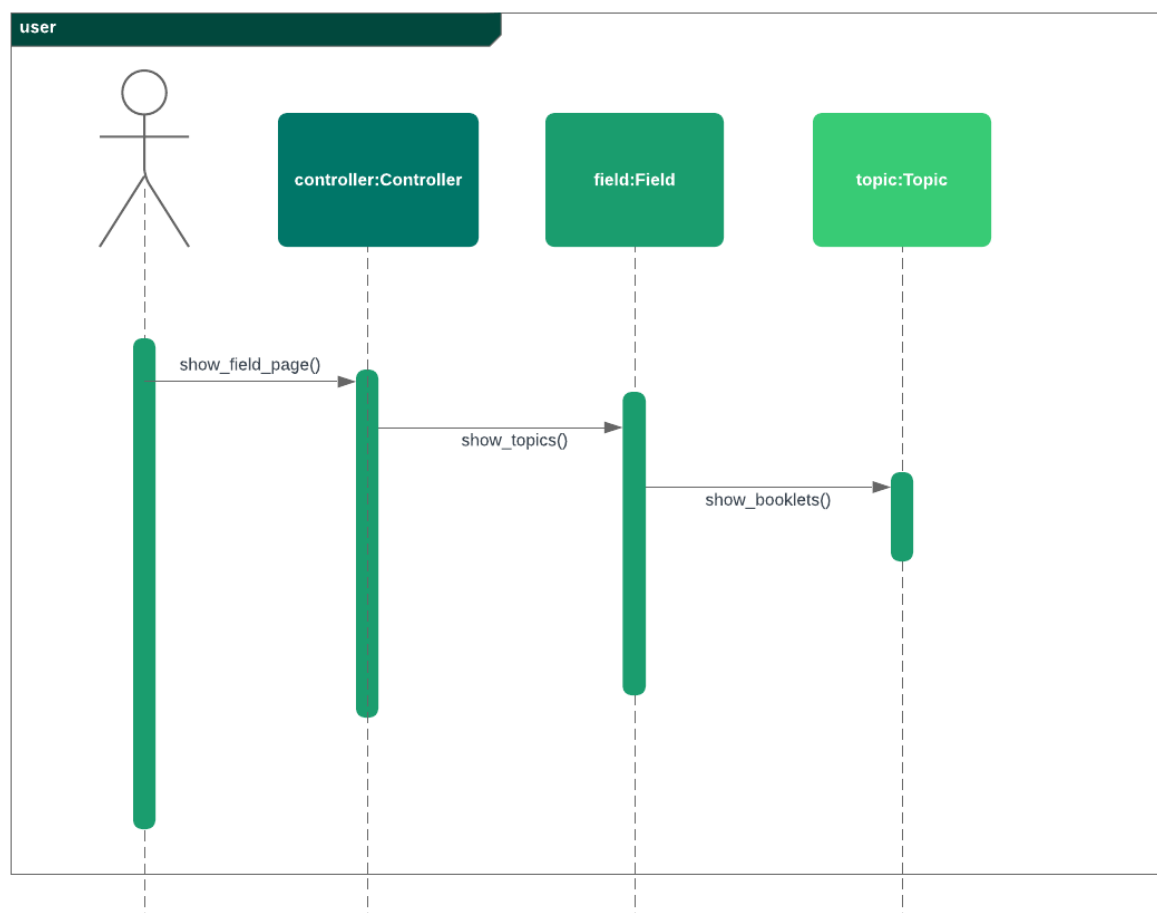
توضیح مختصری درباره شیوه فرایند :

به این صورت میباشد که ابتدا شخص لاگین میکند و سپس وارد قسمت پنل کاربری مربوط به ادمین خودش میشود و سپس در اون قسمت یک تخفیف اضافه میکند .

2. مشاهده جزوات مربوط به یک درس

Show booklets related to a field

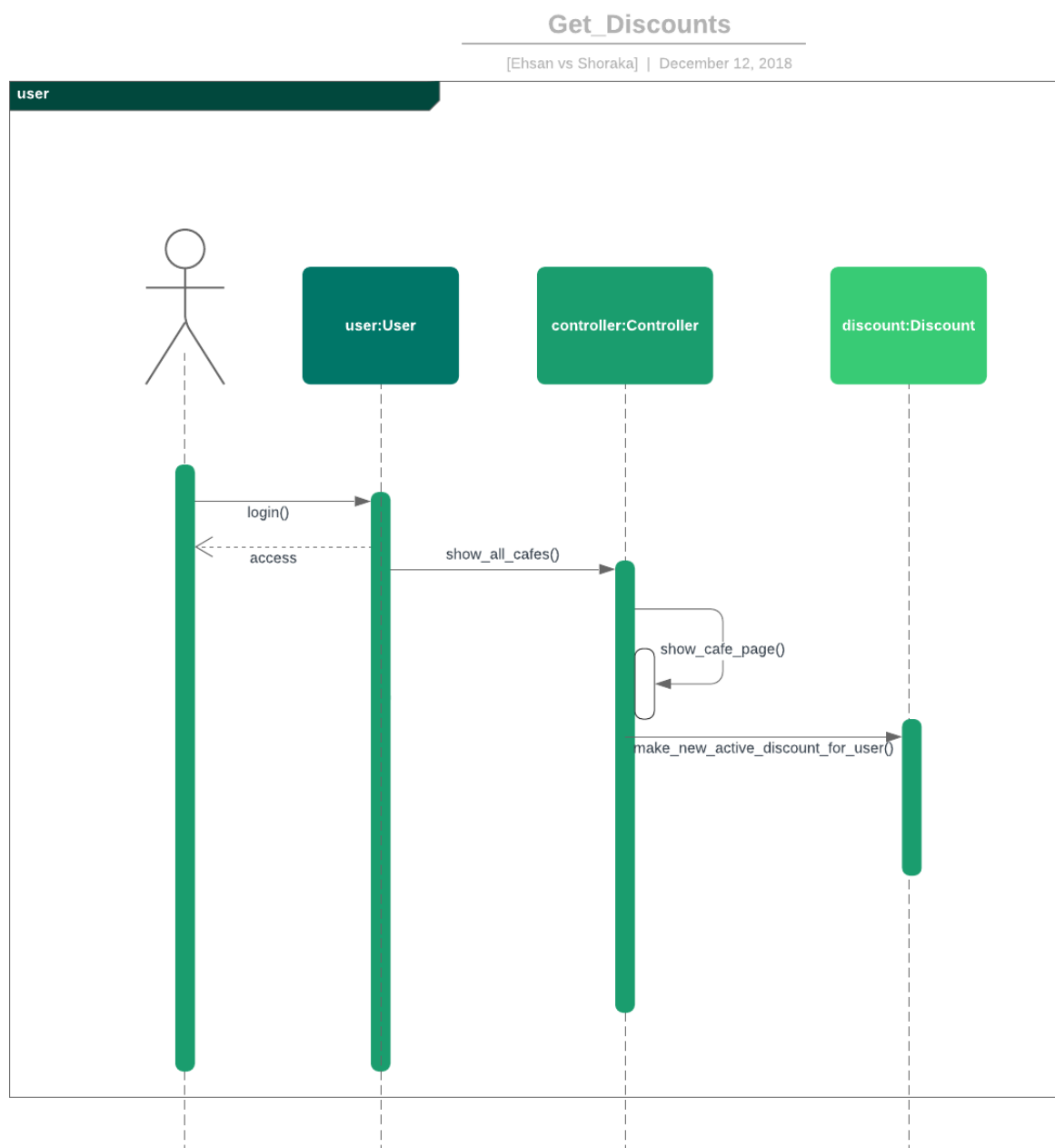
[Ehsan vs Shoraka] | December 12, 2018



توضیح مختصری درباره شیوه فرایند :

به این صورت می‌باشد که ابتدا کاربر صفحه مربوط به رشته را باز می‌کند و سپس در آن صفحه، صفحه مرتبط با درس را انتخاب می‌کند و سپس از صفحه مربوط به درس، صفحه جزوات مورد نظر را باز می‌کند.

3. گرفتن تخفیف کافه



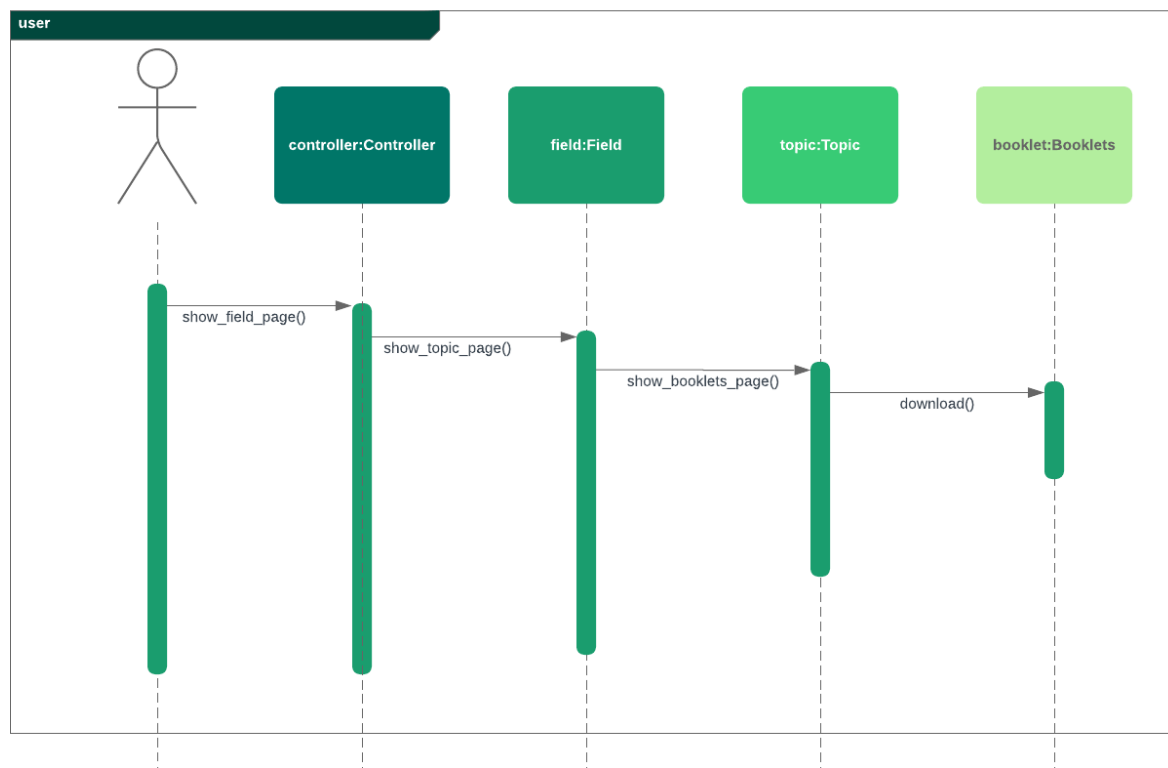
توضیح مختصری درباره شیوه فرایند :

ابتدا این نکته رو یادآور میشویم که تنها کسی میتواند از تخفیف کافه ها استفاده کند که ثبت نام کرده باشد ، پس برای گرفتن تخفیف ابتدا شخص باید لاگین کند و سپس صفحه تمام کافه ها را باز میکند و از اون صفحه ، صفحه کافه مورد نظر خود را باز میکند و در انجا تخفیف خود را میگیرد(گرفتن تخفیف به این صورت است که تخفیف فعال جدیدی در حساب ان شخص ساخته میشود ، که دلیل نامگذاری متد اخر نیز به همین علت میباشد).

4. دانلود جزوه براساس دسته بندی

Download_Booklet_with_category

[Ehsan vs Shoraka] | December 12, 2018



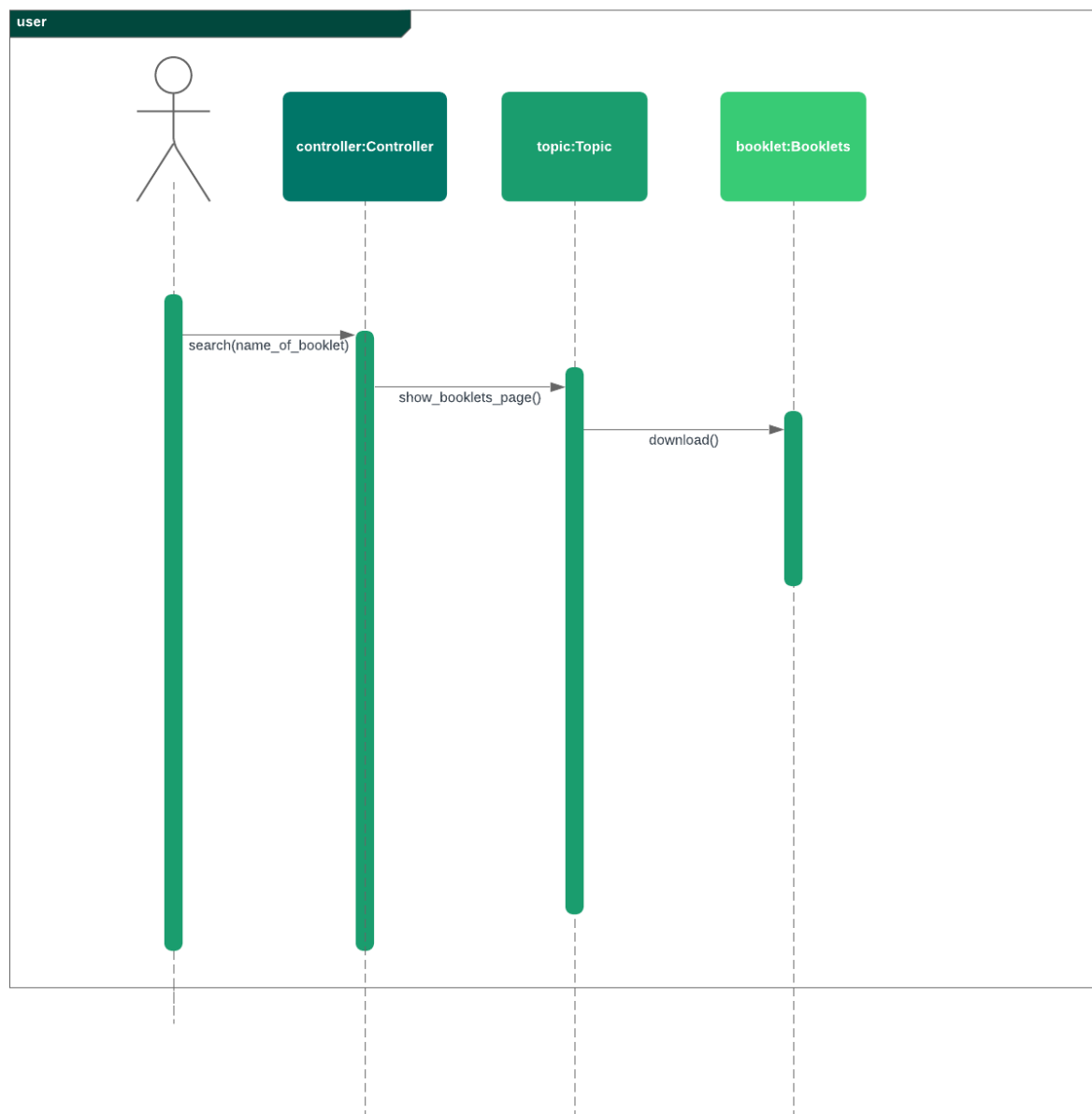
توضیح مختصری درباره شیوه فرایند :

کاربر ابتدا وارد صفحه رشته مورد نظر خود میشود و سپس از آن وارد درس مورد نظر میشود و سپس از آنجا وارد قسمت جزوات میشود و از آنجا جزوه مورد نظرش را دانلود میکند.

5. دانلود جزوه توسط سرچ کردن

Download Booklet with Search

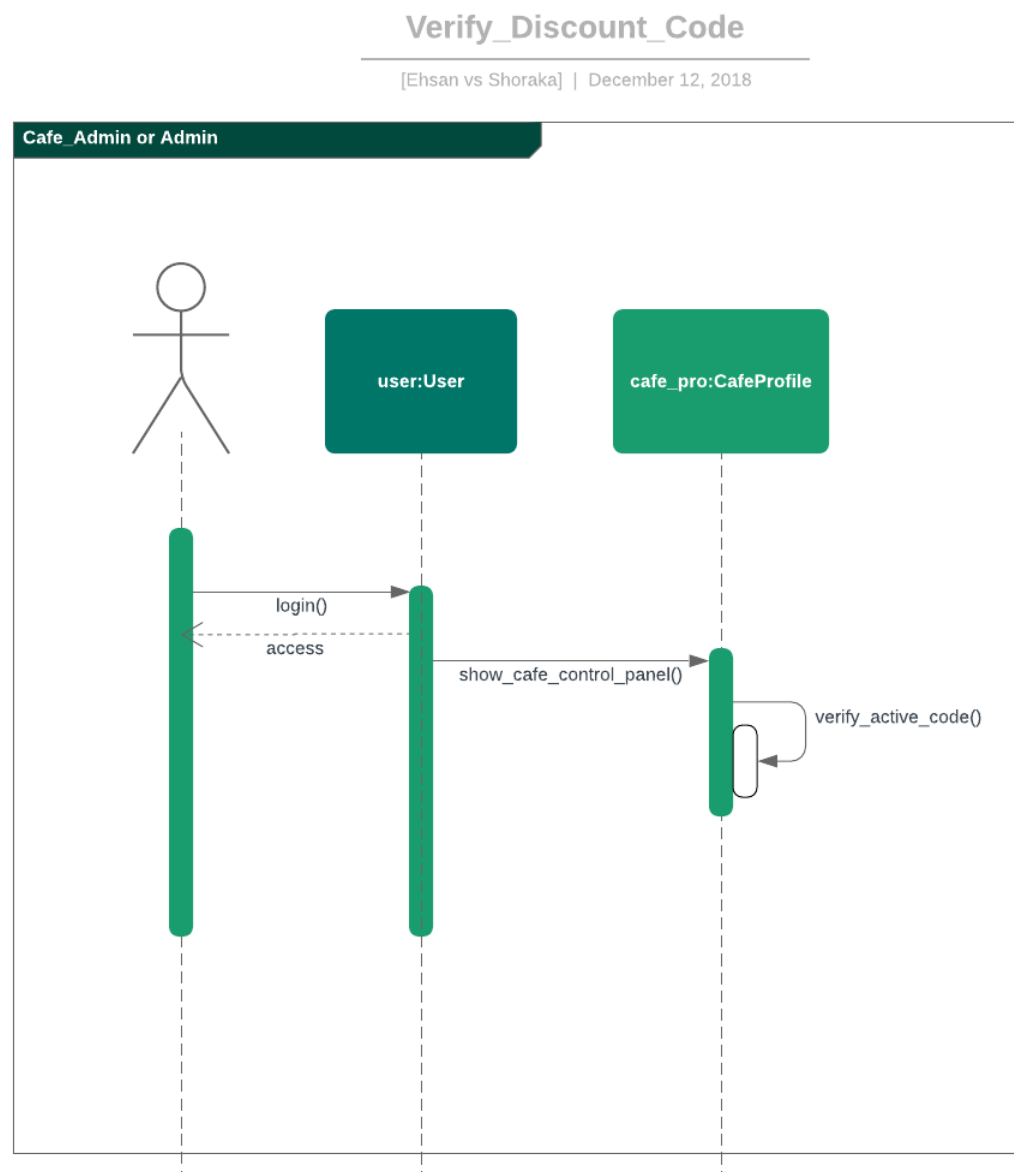
[Ehsan vs Shoraka] | December 12, 2018



توضیح مختصری درباره شیوه فرایند :

کاربر ابتدا نام جزوه ای را میخواهد دانلود کند را در قسمت سرچ ، مینویسد و آن را سرچ میکند و سپس صفحه جزوه برای کاربر نمایش داده میشود و سپس کاربر جزوه مورد نظر خود را دانلود میکند.

6. تایید کد تخفیف



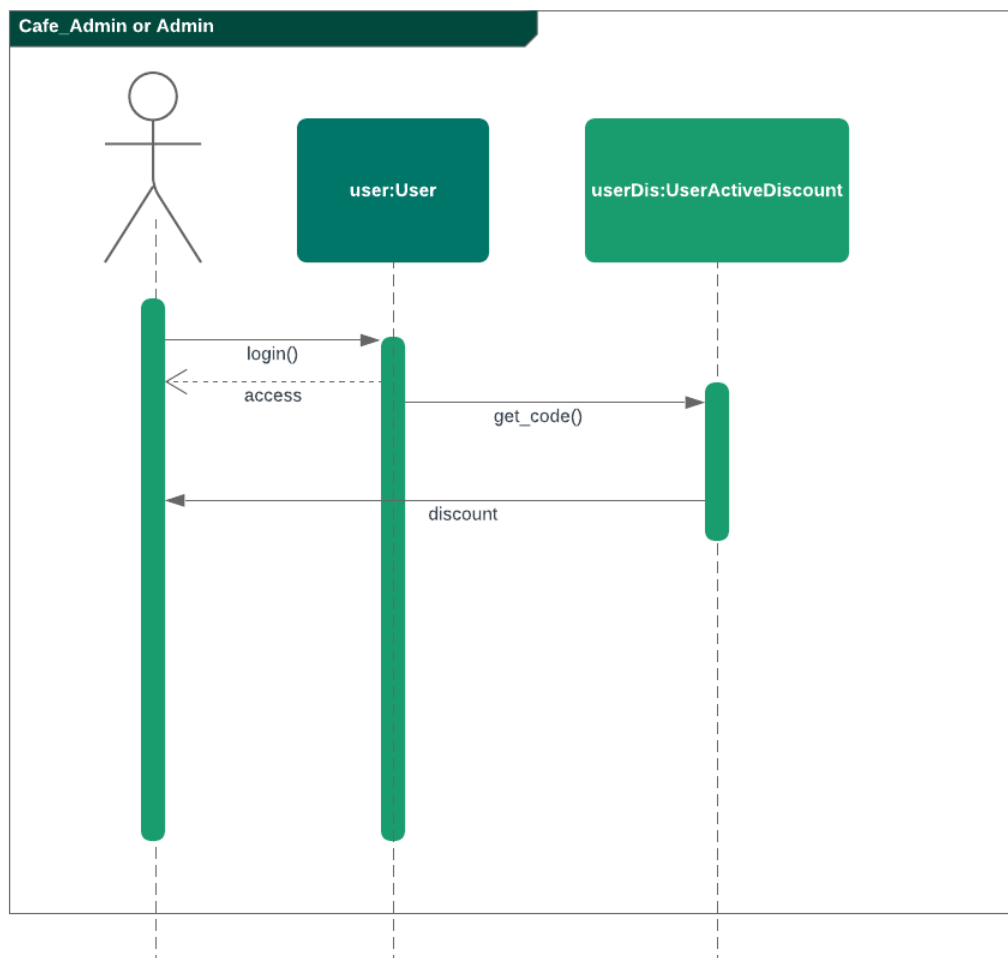
توضیح مختصری درباره شیوه فرایند :

این عملیات بالا اصلش برای ادمین کافه میباشد(اما برای ادمین اصلی نیز برقرار است) ، شیوه کار به این صورت است که در ابتدا ادمین کافه باید لاگین کند و سپس وارد پنل کاربری کافه اش شود و از آنجا کد تخفیف را تایید میکند .

7. مشاهده کردن تمام تخفیف ها

See_All_Discounts_that_Taken

[Ehsan vs Shoraka] | December 12, 2018



توضیح مختصری درباره شیوه فرایند :

ابتدا ادمین کافه (یا ادمین کل) لاگین میکند و سپس وارد قسمتی میشود که در آن تمام تخفیف های فعال که گرفته شده اند موجود میباشد و سپس با دادن کد مربوطه تخفیف های گرفته شده برایش فرستاده میشود .

تعیین وظایف

روش ما روش چابک است.

بک اند وب سایت اسنیدز با پایتون و فریم ورک جنگو و فرانت آن هم با html و CSS و جاوا اسکریپت نوشته شده است.

تعیین وظایف:

بک اند وب سایت با جنگو بخش حساب کاربری: آریا خلیق

بخش حساب کاربری به این صورت است که کاربر میتواند در سایت ثبت نام کند. کاربری که قبلا ثبت نام کرده است میتواند قبلا در این قسمت به سایت وارد شود. با وارد شدن به سایت امکانات بیشتری از سایت مثل تخفیف کافه ها در دسترس کاربر قرار میگیرد.

بک اند وب سایت با جنگو بخش گرفتن تخفیف کافه و نمایش تخفیف های فعال: احسان سوری
این بخش به این صورت است که کاربر پس از ورود به سایت با رفتن به این قسمت میتواند تخفیف های فعال کافه ها را ببیند و یکی را انتخاب کند. با انتخاب یک تخفیف ، تخفیف مورد نظر در صفحه ی کاربری آن قابل مشاهده میشود.

بک اند وب سایت با جنگو بخش آپلود و دانلود جزوه: محمد هادی حسینی
این بخش به این صورت است که کاربر میتواند به جزوه های مختلف دسترسی داشته باشد. میتواند در جزوه ها جستجو کند یا اینکه به قسمت مربوطه رفته و جزوه ی مورد نظر را انتخاب و دانلود کند.

فرانت اند وب سایت بخش تم کلی سایت و نمایش جزوه ها: عرفان رحیمی زاده
این بخش جزوه هایی که در سایت موجود میباشد را لیست میکند و نمایش میدهد.

فرانت اند وب سایت بخش نمایش کافه ها و درس ها و رشته ها : آریا خلیق
این بخش کافه ها و درس ها و رشته هایی که در سایت موجود میباشد را لیست میکند و نمایش میدهد. در بخش کافه ها تخفیف های فعال و قابل انتخاب برای کاربر را لیست میکند و نمایش میدهد.

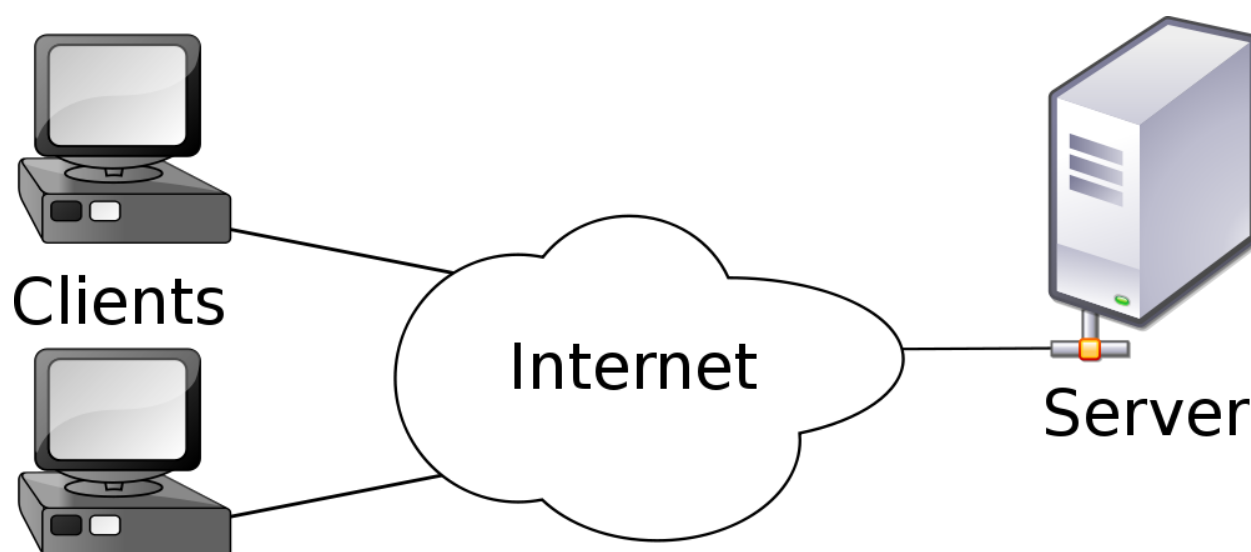
معماری نرم افزار

معماری اصلی ما Client_server است .

دلایل :

برنامه ما در یک سرور اجرا خواهد شد و لازم است کاربران بتوانند از طریق اینترنت به سرور و وبسایتمون دسترسی داشته باشند هر کاربر از طریق مرورگر خود به سایت ما دسترسی پیدا میکند و سرور به request آن پاسخ میدهد و در این بستر با هم تعامل خواهند داشت

و شکل معماری کلاینت_سرور ما شکل زیر میشود :



معماری داخلی بک اند ما MVC است .

دلایل اینکه از معماری MVC استفاده میکنیم :

- ما دارای صفحات زیادی مانند صفحه کافه ، صفحه جزوه و ... هستیم و این صفحات رو کاربر میبیند و این صفحات قرار است مداوم نمایش داده باشند و همین نیاز به این صفحات سبب شد که ما به یک

قسمت نیاز داشته باشیم که فقط وظیفه نمایش را برعهده داشته باشد و این قسمت چیزی است که کاربر میبیند (همان FRONT سایت ما میباشد)

- همانطور که در کلاس دیاگرام نیز دیدید ما نیاز به یک قسمتی داریم که وظیفه **logic** برنامه را برعهده داشته باشد به گونه ای که عملیات هایی که در برنامه انجام میشود رو ، کدشون رو در اینجا نوشته باشیم و از قسمت های دیگر جدا باشد و این قسمت با داده ها بیشتر سروکار داشته باشد .
- و به صورت واضح هم ما نیاز به یک قسمت داریم که **action** های مورد نیاز رو که توسط کاربر انجام میشود رو بگیرد و مشخص کند که این **action** مربوط به کدام قسمت ، از قسمت های بالا هست و کار مرتبط با اون **action** را انجام میدهد ، مثلا زمانیکه کاربر بروی دکمه صفحه کافه کلیک میکند ، این قسمت میاد به قسمت نمایش میگه که باید صفحه رو عوض کنی و صفحه ای که کاربر میخواد رونمایش بدی و در کل رابط بین قسمت **logic** و **view** برنامه میباشد.

با دلایل بالا و اینکه MVC نزدیکتر مدل به این دلایل میباشد ، پس این سبب شد که ما از MVC استفاده کردیم .

و شکل معماری مربوط به MVC ما شکل زیر میباشد .

