

UML: یک زبان مدل سازی استاندارد است که برای طراحی و مستند سازی سیستم ها

نرم افزارهای استفاده می شود. UML به تفاصیل دهنده و مجهز آن نظر از تراجم دارد

کا ساختار و رفتار سیستم ها را بجاییه را به صورت بصری ترسیم و تحلیله کند. این زبان

شامل انواع مختلفی از نهادهای مختلف است که جنبه های مختلف یک سیستم را نمایش می دهند

اینچنانچه مدل سازی و طراحی (UML)

قابلیت هایی را که آنها در نرم افزار واقعی Lucidchart

دارند. مثلاً قابلیت هایی که با Microsoft Visio

UML 2 متن باز - پیشگانه از Star UML

دارند. قابلیت هایی که برای تحلیل و طراحی و مستند سازی

آنها را می دهند Draw.io

UML 1.4 متن باز و از پیشگانه از Argot UML

دارند. قابلیت هایی که برای طراحی و مستند سازی و همکاری های

نمودار کلاس (Class): نمایش ساختار کلاس‌ها و روابط بین آن‌ها.  
Subject: Day.

نمودار مولفه (Component): نمایش اجزای سیستم و روابط بین آن‌ها.

نمودار بُنی (Object): نمایش نمونه‌هایی از کلاس‌ها و وضعیت‌های آن‌ها در یک زمان خاص.

نمودار ترکیب (Composite): نمایش اجزای درونی یک کلاس و تعاملات بین آن‌ها.

نمودار بسته (Package): گروه بندی منطقه کلاس‌ها و موارد غیرعلوی که ممکن است در یک پروژه مورد استفاده قرار گیرند.

نمودار استقرار (Deployment): نمایش نحوه استقرار نرم‌افزار بر روی سخت‌افزار.

نمودار پروفایل (Profile): نمایش استفاده از پروفایل‌های UML.

نمودارهای ساختاری (Structural)  
نمایش ساختار اجزای سیستم را باعث  
نمودند.

نمودارهای رفتاری (Behavioral)  
این نمودارها رفتار و تعاملات داخل سیستم را نمایش می‌دهند.

نمودار مورد کاربرد (Use Case): نمایش تعاملات بین کاربران و سیستم.

نمودار فعالیت (Activity): نمایش چرخان فعالیت‌های فرآیندهای سیستم.

نمودارهای رفتاری (State): نمایش وضعیت‌های مختلف یک شیء را نشانه می‌گردند.

نمودار توالی (Sequence): نمایش ترتیب تعاملات بین اشیاء مختلف.

نمودار همکاری (Communication): نمایش تعاملات بین اشیاء با تمرکز بر استعمالات.

نمودار زمانی بندی (Timing): نمایش تغییرات وضعیت اشیاء با لذت نظر.

نمودار کلی تعامل (Interaction Overview): ترکیب نمودارهای مختلف تعاملات.

لاربرد UML: فرآور طراحی، تحلیل و مسند سازی سیستم‌های نرم افزاری

محدود استفاده مرار مرگرد و ب توجه به دستگاه کمپیوچر لذت‌نامه سیستم را به صورت

گرافیکی نهایی دهنده

مزایای UML:

۰ استاندارد سازی طراحی سیستم

۰ تسهیل ارتباط بین اختصاصات

۰ کم برداشت سیستم عملی از بیان سازی

۰ امکان استفاده مجدد از مدل‌ها

معایب UML:

۰ پیچیدگی بیش نمودارها

۰ نیاز به دانش تخصصی برای تفسیر صحیح مدل‌ها

۰ هزینه زیاد برای ایجاد و بروزرسانی مدل‌ها

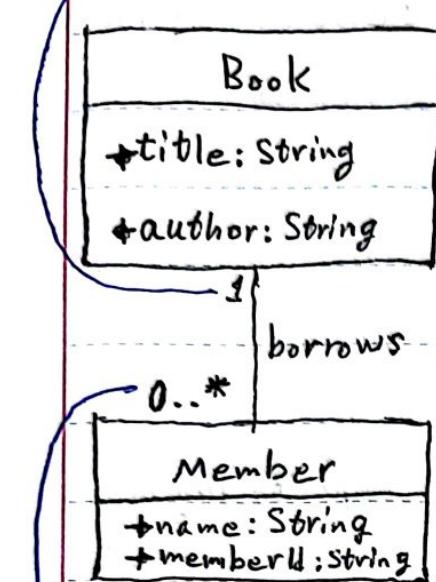
جنبه‌های ایستادوپیا در UML:

ایستاد: مدل ساختار سیستم شامل کلاس‌های اسما و روابط آنها

(Class Diagram)

پوپیا: مدل رفتار سیستم در زمان اجرا است (Sequence Diagram)

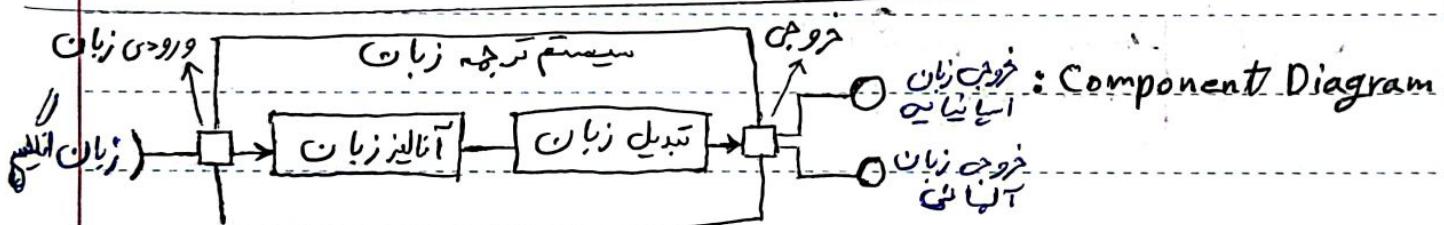
## Class Diagram



بطبع درسی قبل از نام و پسرگاه با محابیت های  
یک کلاس مه آید.  
+ عضویت: یعنی مه توان از خارج کسان بآور  
درسی داشت.  
- سره حفظیت:  
# محافظت شده

فوقیح: این مفهوم رنگارنگ می دهد که در یک سیستم مدیریت  
کتابخانه، هر کتاب دارای عنوان و نویسنده است و می تواند  
قرض بگیرد.

به عضویت متعلق داشته باشد. هر عضو حلال نام و رشته منحضر برخواست و می تواند صحیح  
باشه کتاب بقرض بگیرد

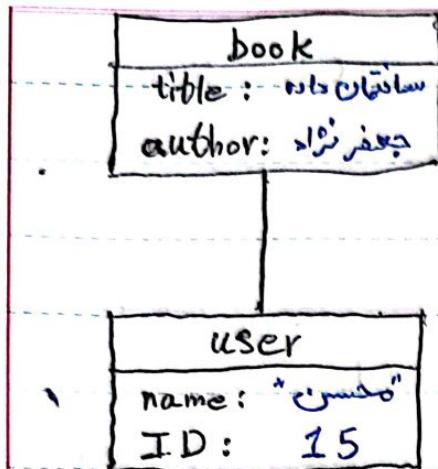


فوقیح: این سیستم ترجمه زبان را نشان می دهد که می تواند زبان انگلیسی را

به زبان اصلی ترجمه کند که هر زیر مؤلفه های خود یعنی

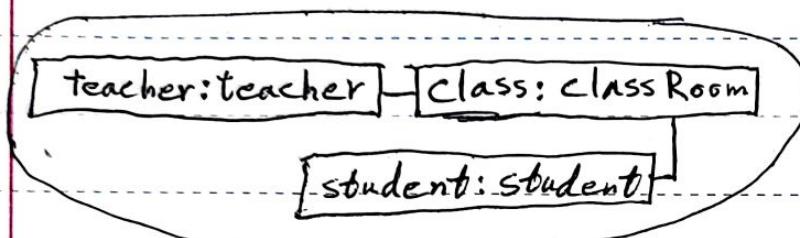
تبدیل و آنالیز زبان حاده های ما از آنها عبور می کند تا تبدیل زبان

انجام شود.



: Object Diagram

توضیح: کتاب با عنوان ساخته‌اند دارد  
به کاربر 15 با نام مخصوص موصوع شود

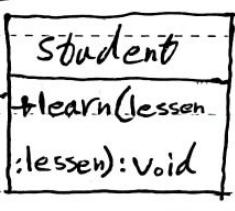


: Composite Diagram

توضیح: این نمودار یک سیستم

تدريس را نشان می‌دهد  
که معلم در گرایش درس

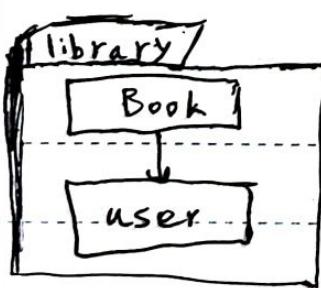
و دانشجو اسے مدد فراز



است که ساختار داخلی و جلوگیری تغالم

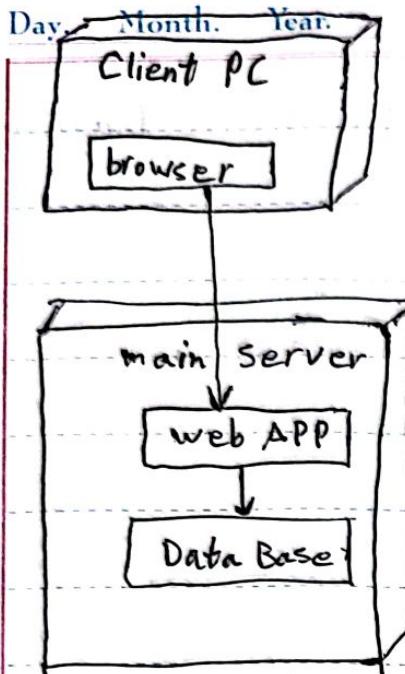
این بخشی ها در یک فرآیند تدریس

تفصیل دارد.



: Package Diagram

توضیح: این نمودار شبیه به نتیجه بخشی های مختلف  
نرم افزار مثل کاسن ها و نمودارها و یا مؤلفه های تکونه در  
بعضی از گروه بینز شده اند و باهم ارتباط دارند.



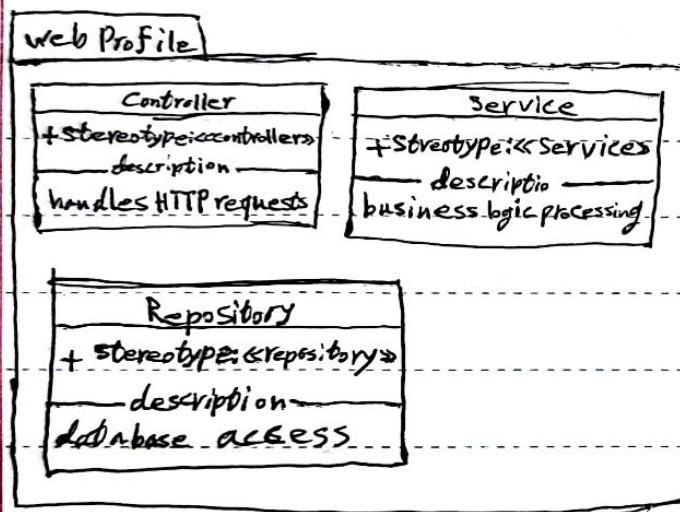
Subject.

### : Deployment Diagram

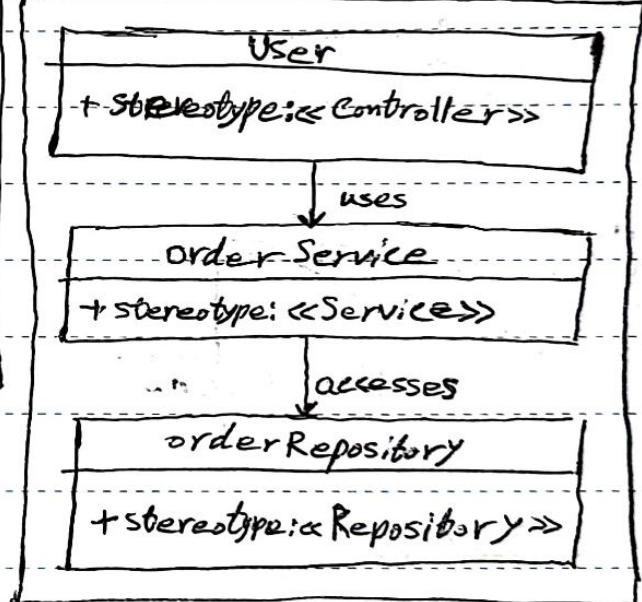
تفصیل: این نمودار که می‌کند به بیان نرم افزار

در کجا اجرا می‌شود، به چه ساخت افزارهای ناز

در وحدهای مختلف طبق مخفف باهم ارتباط دارند



### : Profile Diagram

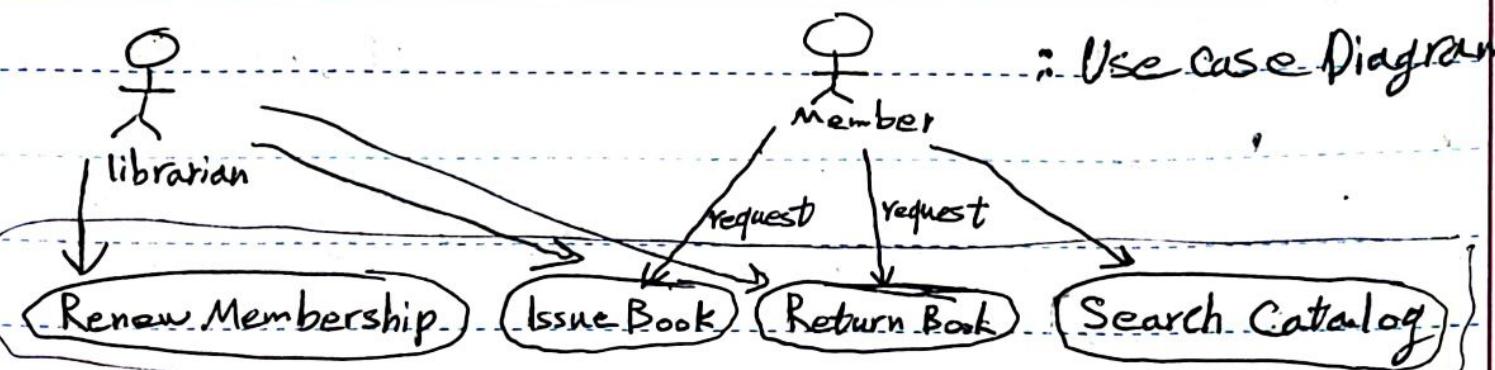


تفصیل: از استفاده یک دایگرام مختلف برای تئوری های متفاوت می‌باشد

سیستم مبتنی بر اسناد مداری است.  
HTTP: بین مدیریت درخواست و Controller ←  
استریوتایپ ها: Service ← بین روزارویی و تئوری ریزی ←  
Repository: بین داده هایی و تئوری داده ←

و این اتفاقاً در OrderService از User درست نموده شد  
orderrepository ~ orderService

## Use Case Diagram

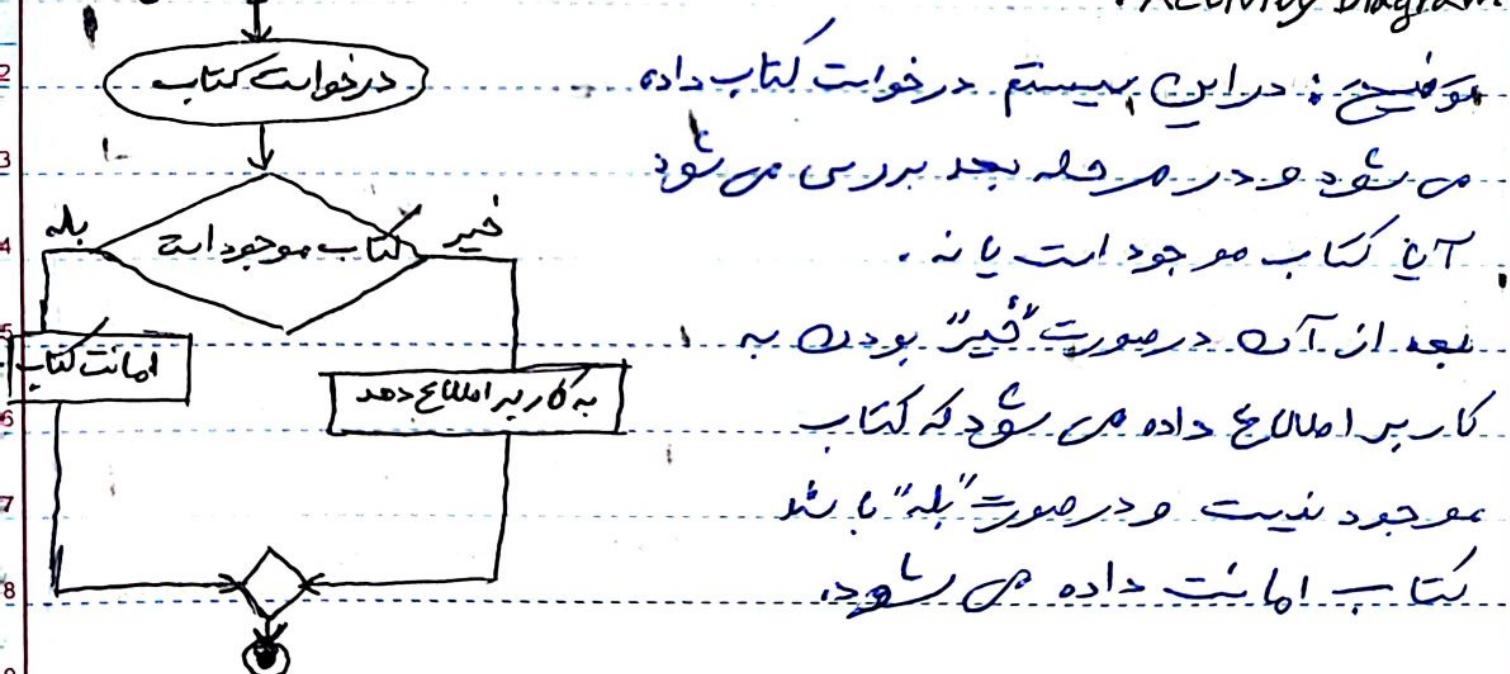


توضیح: این مدل سیستم کتابخانه را نشان می‌دهد

با زیگار و کتابدار: همچو کتابخانه ای داده و بازگردانده کتابها و تمدید عضویت.

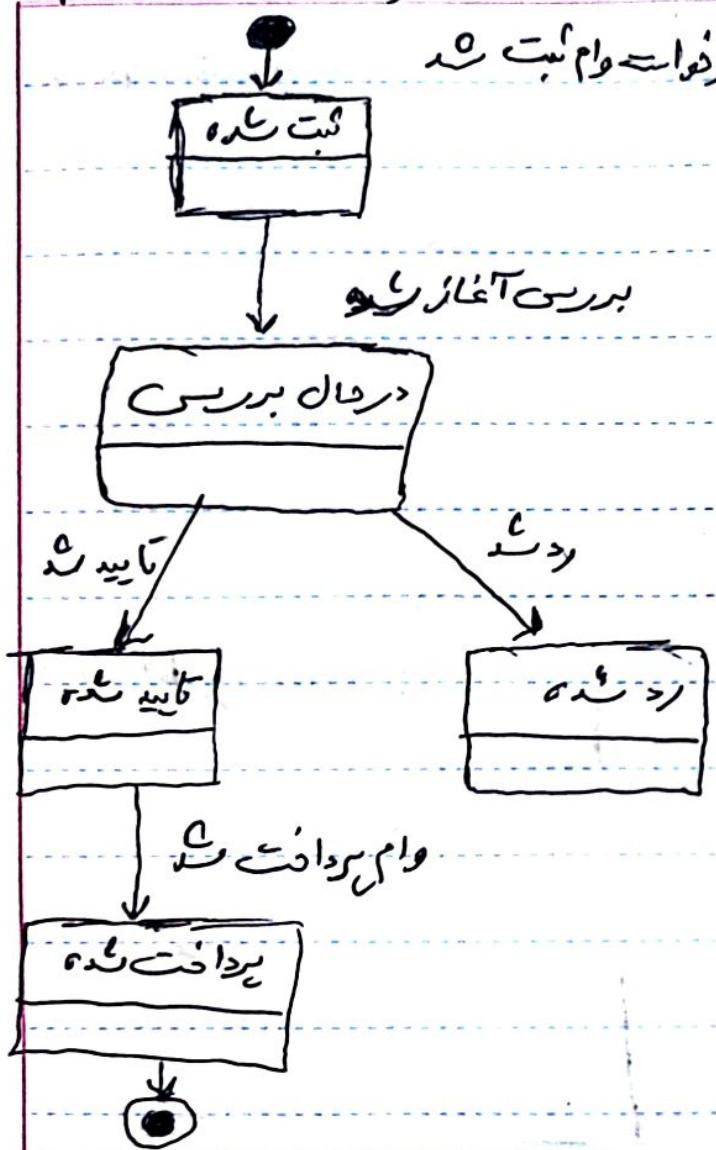
عنوان: من تواند به همچو غیر کتاب خود را درخواست کنم که بیرون دارد و درخواست ایامنت یا بازگردانه کتابها را داشته باشد.

## Activity Diagram



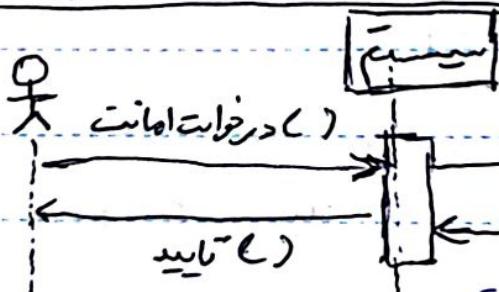
در فرآیند ایت شد

: State Machine Diagram



نحوی خوب: این نحوه وضاحت دارد،

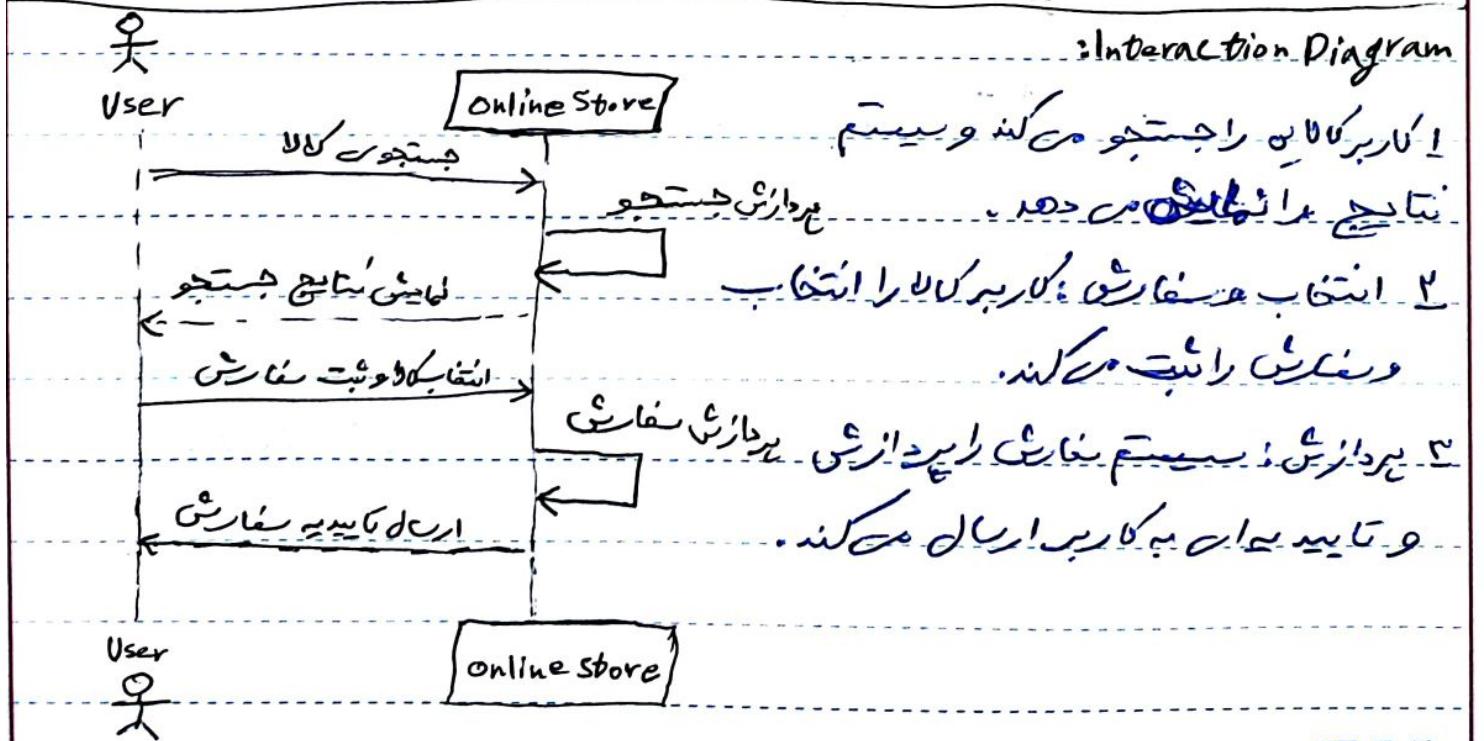
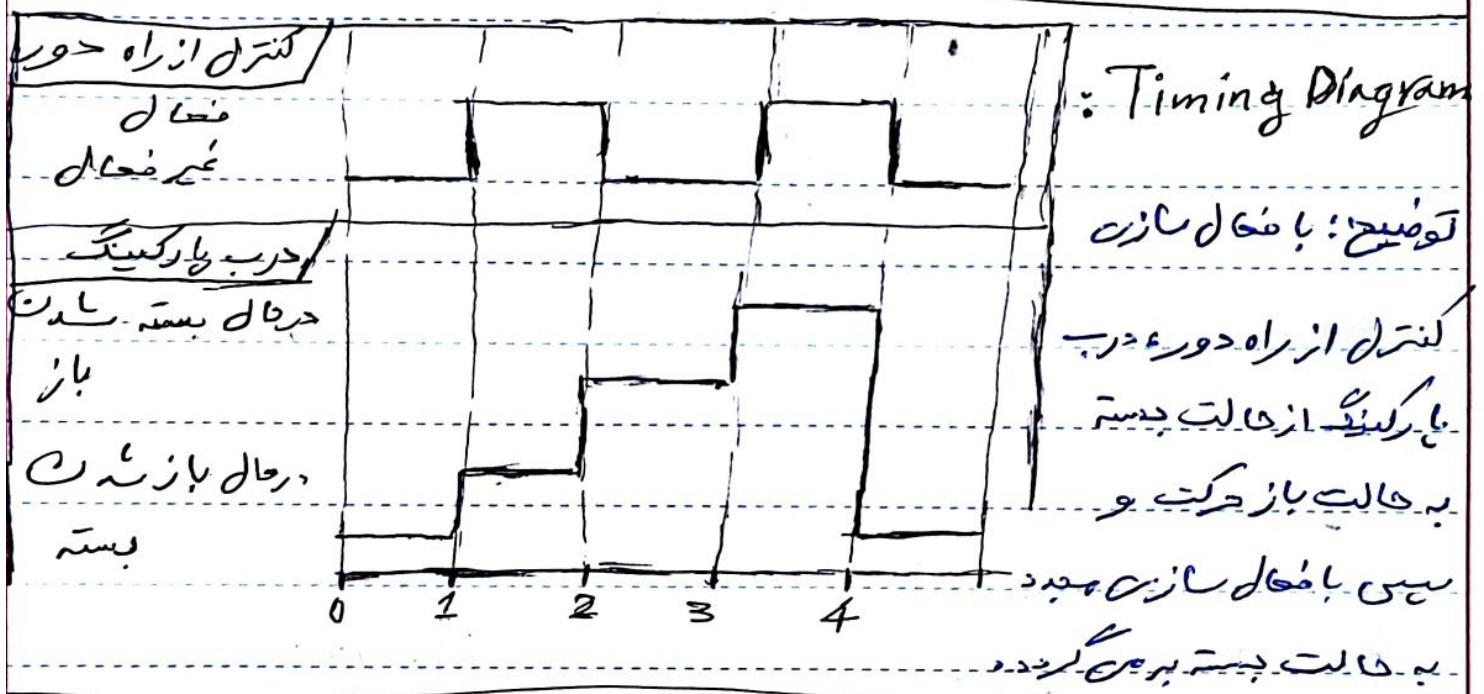
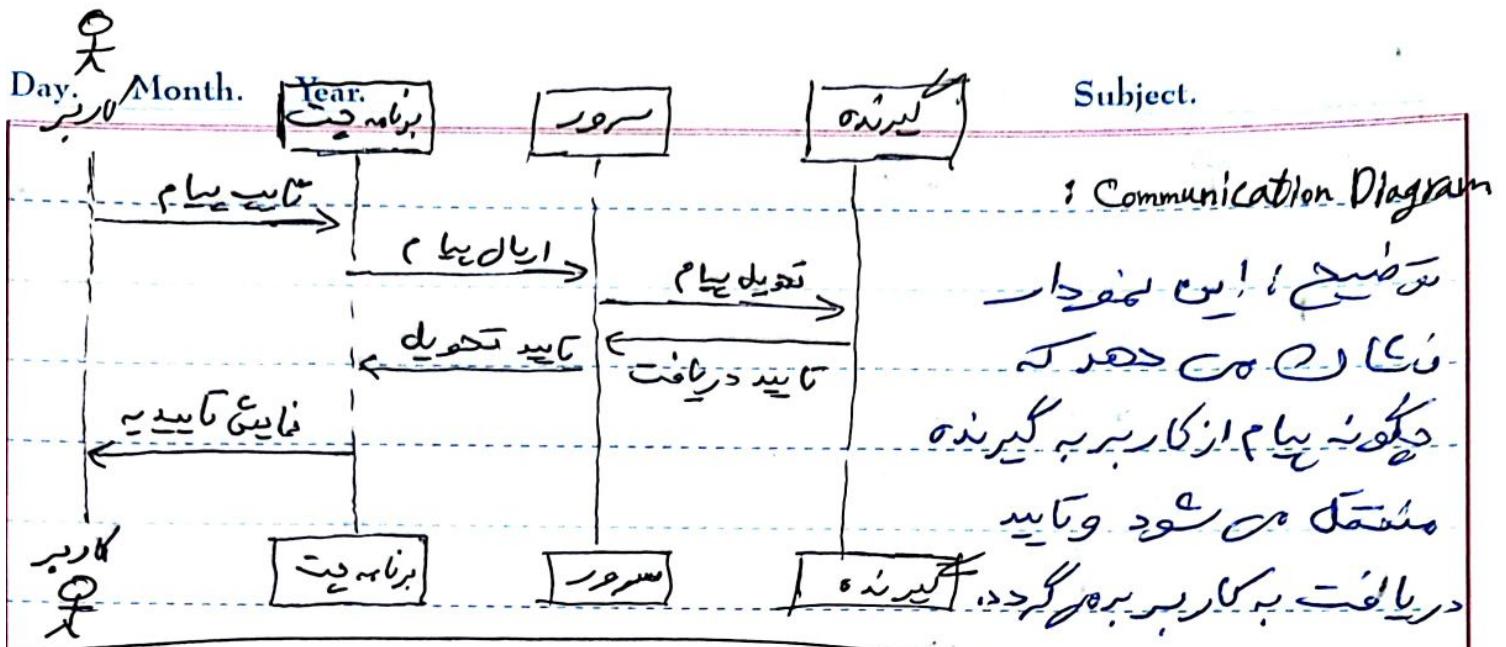
هر چند در خواست وام از بیت تا پرداخت را نشان می‌دهد.



: Sequence Diagram

نحوی خوب: در این سیستم بازگردانی سیستم در خواست امانت کتاب صورت دهد و در سیستم در خواست بررسی می‌شود و در صورت معرفی دو دل کتاب پیکار بر بیمه گرد و تایید می‌شود.





تعریف ASP.NET Core: این فریم ورک متن باز و جذب سوچ (Cross-platform)

برای ساخت اپلیکیشن های وب تحت شبکه و سرویس های وب است

توسعه مادروسات توسعه داده شده، این فریم ورک نسخه جدید و بهبود

یافته است ASP.NET Core ابتداء از کارکرد رون روی سیستم عامل های

مختلف منتشر شده است

: ASP.NET Core پیشتر نسخه های قدیمی می باشد

۱- جذب سوچ (Cross-platform) ۲- متن باز ۳- سریع و سبک ۴- قابل برترین

نرم افزارهای ASP.NET Core

۱- ساخت وب سایت های API ۲- ساخت اپلیکیشن های

تحت شبکه

منابع و مراجع ASP.NET Core

۱- منابع → بسیار سریع و مابل اعتماد برای ساخت اپلیکیشن های بزرگ بازمان

# C# کارهای کارخانه با ابزارهای ناشر و سافت میل AZURE خوبی حاصل شد.

۲- معابر براحتی پروژه های خالی موجود یا کری داشتند از آن پس بعد

است نایاب داشتند برنامه نویس دارد

بعدین است وسیله Active server Pages مخفف ASP: ASP

متاوره میان از ما در سافت براحتی سافت وب سایت ها ایجاد میشوند و بر

است و به برنامه نویسان کمترین کد مخفف احتراب دعوهای سپاه شد.

تعاقوت با ASP.NET -! : ASP.NET

Asp.NET میباشد است و بازیابی صفحه میل # C# کارهای کارخانه و قوه ترکیب اطلاعات

نیز برآمد.

بعدین dot Net : نیز پلتفرم نرم افزاری عرضی و درک توسعه از ما در سافت

است و براحتی سافت انفع المیکس ها استفاده میشود، این پلتفرم میتواند

اجازه و مجموعه ای از ابزارها و کتابخانه های را مراهم میکند.

ویژگیهای dotnet core : ۱- پلی‌اوپن ۲- یون‌باز ۳- قابل‌تغییر ۴- امنیت و همکاری بالا

\* این پلتفرم چیزی: ۱- CLR : یک محیط اجرایی که بزرگ باشد از جامد کند.

۲- FCL : مجموعه از نکات آماده که در پروژه را فایلهای خارجی و داده‌ها را ساده کند.

۳- ASP.NET Core : نسخه مدرن و هسته‌بار ASP.NET

۴- وب سایت‌های API های سریع و مقایسه‌پذیر استفاده می‌شود. این فناوری در پلتفرم ASP.NET Core

۵- Razor Pages و Dependency Injection و Middleware

۶- اجزایی نرم افزاری کوچکی که در طبقه برداری‌هایی به نام Middleware نامیده می‌شوند.

۷- درخواست HTTP اجراء شده. هر Middleware خاص خاص را انجام می‌دهد. مانند:

۸- Authentication Middleware

۹- Logging Middleware

۱۰- Exception Handling Middleware

۱۱- تنظیم منابع شوند. در فایل Startup.cs در متد Configure() در Middleware

۱۲- ASP.NET Core در Routing : Web Routing

۱۳- Attribute Routing که از ارائه سهند. این قابلیت از

۱۴- Convention-based Routing

```
[Route("api/products/{id}")]
```

```
public IActionResult GetProduct(int id)
```

```
{  
    return OK($"Product ID: {id}");  
}
```

درخواست ۱۰ لایه از ارائه شود.

**Data Annotation**: مجموعه ای از ویژگی‌های قابل استفاده در مدل‌هاست که این است که برای اعتبارسنجی و رویکردهای تعریف محدودیت‌ها و ارائه تفضیل‌ها را برابر فیلترها بگیرد. این ویژگی‌ها در ASP.NET Core برای اطمینان از صحبت و سازگاری داده‌ها وجود دارد. این ویژگی‌ها امکنی نیاز که فیلتر اضافه برای اعتبارسنجی داده‌ها منع نموده باشند.

**DbContext**: کلاس اصلی در Entity Framework Core است که برای مدیریت ارتباط با داده‌ها استفاده می‌شود. این کلاس فضای نام‌گذاری مختلف روش‌های داده‌ها را مدیریت می‌کند. و پایه ناچاری الگوی Unit of work را به عده دارد. توجه داشته باشید که توانید با استفاده از DbContext مدل‌ها را تعریف کرده و یا یک داده را به عنوان نیاز به نوشتگی مستقیم کوئری‌های SQL مدیریت کنید.

**ConnectionString**: یک رشته متنی حاوی اطلاعات مورد نیاز برای اتصال به پایه داده است. این اطلاعات شامل نام سرویس نام یا یک داده، درونی احراز هویت و سایر تنظیمات مربوط به اتصال است. معمولاً در فایل JSON یا کنفرینس برنامه ذخیره شود تا آنکه در تغییر آسان تنظیمات بپور نیاز به اصلاح کد فراموش گردد.

**Dependency Injection**: یک الگوی طراحی است که در ASP.NET Core برای مدیریت وابستگی بین کلاس‌ها ایجاد می‌شود. این رویکرد باعث کاهش وابستگی‌های بین کلاس‌ها می‌شود و افرادی اخطلاف پذیر و سهولت تست نویسی می‌شود. در ASP.NET Core DI به صورت پیش‌فرض در هسته فریم‌ورک گنجانده شده و توجه داشته باشید که توانید وابستگی‌ها را به صورت خوکاوار به کلاس‌ها که دیگر تغیریگر کنند.

**Controller:** یک از اجزای اصلی ساختار MVC است که وظیفه پردازش درخواست کاربران و تولید پاسخ مناسب طبقه بوده دارد. در ASP.NET Core، کنترلرها برای مدیریت عملیات مختلف مسافت داده ها، پردازش اطلاعات و ارسال خروجی به کاربروند کنترلرها معمولاً با استفاده از Action Method درخواست ها را دریافت کرده و مبنی بر مطلع صورت نظرهای مسافت می کند.

**ORM (Object Relational Mapping):** یک تکنیک در برنامه نویسی است که امکان ایجاد با پایگاه داده را به صورت شیوه نویسی کرده است. به همین نویسی که کوئری های SQL، توابع دستگاهی از کلاس ها و اسیا برای مدیریت داده ها استفاده کنند. این رویی باعث کارکردن بسیاری از امنیتی و بهبود کیفیت کدنی کرد. ORM مجموعه نویسی فرم ورک های مانند Entity Framework می باشد.

**Entity Framework:** یک ORM رسمی مایکروسافت برای .NET است که ارتباط بین برنامه و پایگاه داده را تسهیل می کند. این فرم ورک به حوزه دستگاه اجازه ایجاد، نیاز به نویسی دستی کوئری های SQL، عملیات ایجاد، حذف، بروزرسانی و حذف داده ها را انجام می دهد. Entity Framework به صورت خودکار مدل داده های را به پذاروی اضافه و امکان مدیریت پایگاه داده را به شیوه ای که آنده فراهم می کند.

**Razor:** یک موتور قالب سازی در ASP.NET Core است که امکان ترکیب کدهای HTML و C# را در صفحات وب فراهم می کند. این فناوری به توسعه دهنده اجازه می دهد که از مبنی بر مدل داده نویسی در کنار HTML استفاده کنند و به عنوان اینکه نیاز به نویسی کدهای بسیاری جایجا ایجاد می کنند. سرور داشته باشند. Razor برای ساخت صفحات داتا مایک در MVC و Razor Pages استفاده می شود.

Day.	Month.	Year.	Subject.
Validation: فرآیندی است که در آن داده های ورودی بررسی شده و از صحت آنها اطمینان حاصل می شوند. در ASP.NET Core، اعتبارسنجی ممکن شوند به صورت مستقیم سرویس است که بین انجام شود. این فرآیند معمولاً با استفاده از Data Annotation و Middleware انجام می شود. اعتبارسنجی و مرتبط بر نامه نویسی اجراء شود و اعتبارسنجی باعث جلوگیری از ورود داده های نامعتبر و افرادیست امنیت سیستم را تأمین می کند.			