

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

مروری کلی بر زبان مدل سازی UML

مدرس : مهندس افشین رفوآ

دوره آموزش UML

مروری کلی بر زبان مدل سازی UML

**UML** یک زبان مدل سازی همه منظوره می باشد. این زبان در ابتدا به منظور نمایش تصویری رفتار سیستم های نرم افزاری و غیر نرم افزاری پیچیده اختراع شده و ارائه گردید اما هم اکنون یک استاندارد **OMG** (گروه مدیریت آبجکت) تلقی می شود.

**UML** المان ها و **component** هایی ارائه می کند که قادر به برآورده ساختن نیازهای سیستم های پیچیده می باشد. **UML** از مفهیم پایه ای و متدولوژی شی گرا پیروی می کند، بدین معنی که با بهره گیری از زبان های تصویر نما همچون **UML** می توان آن ها را مدل سازی کرد.

نمودارهای **UML** از دیدگاه های مختلف نظیر طراحی، پیاده سازی، استقرار/توزیع و غیره ... ترسیم می شوند. در خلاصه، زبان **UML** را می توان یک زبان مدل سازی که قادر به نمایش جنبه هایی همانند معماری، رفتار و ساختار سیستم می باشد، تعریف کرد.

اشیا اساس و بنیاد دنیای شی گرا محسوب می شوند. اولین کاری که در تجزیه، تحلیل و طراحی شی گرا بایستی انجام داد، شناسایی کارآمد اشیا می باشد، پس از آن فقط بایستی مسئولیت ها را به اشیا تخصیص داد. در مرحله ی بعد، به طراحی با استفاده از خروجی تجزیه و تحلیل خواهیم پرداخت.

**UML** نقش مهمی در تحلیل و طراحی شی گرا ایفا می نماید. نمودارهای این زبان نیز طراحی را مدل سازی کرده و به صورت گرافیکی ارائه می کند. بنابراین می توان نتیجه گرفت که **UML** نقش بسیار مهمی را ایفا کرده و جایگاه بسیار مهمی دارد.

## نشانه گذاری های زبان مدل سازی UML (UML notations)

**Notation** ها مهم ترین المان ها در مدل سازی محسوب می شوند. استفاده ی کارآمد و بجا از **notation** ها در جهت ایجاد یک مدل کامل و معنی دار کمک شایانی می کند. یک مدل تا زمانی که هدف و مورد کاربرد خود را به درستی به تصویر نکشد، کاملاً بی ارزش قلمداد می شود.

از این رو تاکید بر یادگیری **notation** ها از اهمیت خاصی برخوردار است. برای موارد و رابطه های مختلف طبیعتاً از **notation** های مختلف بهره گرفته می شود. نمودارهای UML نیز از همین نشانه گذاری های (**notation**) اشیا و رابطه ی بین آن ها تشکیل می شوند. توسعه پذیری (**extensibility**) یکی دیگر از ویژگی ها و قابلیت هایی است که به کارایی و انعطاف پذیری هرچه بیشتر UML کمک می کند.

### نمودارهای UML

نمودارها قلب UML هستند. این نمودارها به طور کلی تحت عناوین ساختاری (**Structural**) و رفتاری (**behavioral**) رده بندی می شوند.

1. دیاگرام های ساختاری خود به نمودارهای ایستا (نمودارهایی که ساختار ایستای سیستم را نمایش می دهند) از قبیل نمودار کلاس، شی و غیره ... گروه بندی می شوند.

2. نمودارهای رفتاری نیز به دیاگرام های پویایی (نمودارهایی که تاکید برنمایش رفتار پویای سیستم با نشان دادن هماهنگی میان اشیا و تغییرات وضعیت داخلی اشیا دارند) نظیر توالی (**sequence**)، همکاری (**collaboration**) و غیره ... دسته بندی می شوند.

با توجه به آنچه گفته شد، توسط این نمودارها می توان ماهیت پویا و ایستای یک سیستم را به تصویر کشید.

### نمودار کلاس (class diagram)

نمودار کلاس پرکاربردترین دیاگرام در UML می باشد که توسط جامعه ی شی گرا مورد استفاده قرار می گیرد. این نمودار به شرح اشیا در یک سیستم و رابطه ی بین آن ها می پردازد. دیاگرام کلاس از توابع (**function**) و متغیرهای عضو (**attribute**) تشکیل می شود.

یک نمودار کلاس تنها قادر به نمایش یک جنبه از سیستم مورد نظر می باشد درحالی که چندین نمودار کلاس می توانند نمایی ایستا از کل سیستم بدست دهند. همان طور که گفته شد نمودار کلاس در اصل نمایی ایستا از سیستم را ارائه می دهد.

نمودارهای کلاس تنها نمودارهای UML هستند که به طور مستقیم قابل نگاشت توسط زبان های شی گرا همچون C# می باشد. از این رو به طور متداول توسط جامعه ی برنامه نویسان و توسعه دهندگان مورد استفاده قرار می گیرد.

### نمودار شی (object diagram)

نمودار شی درواقع نمونه ای از دیاگرام کلاس می باشد و به همین خاطر بسیاری از المان های پایه ای بین این دو نمودار یکسان و مشترک می باشد. دیاگرام های شی از اشیا و پیوندها (link) تشکیل می شوند؛ به بیان دیگر نشانگر یک دید کامل یا جزئی از ساختار سیستم مدل سازی شده در یک زمان مشخص است. نمودارهای شی برای پیش الگو سازی (prototyping)، مهندسی معکوس و نیز مدل سازی سناریوهای کاربردی مورد استفاده قرار می گیرد.

### نمودار اجزا (component diagram)

نمودار component یکی از مجموعه دیاگرام هایی است که جهت توصیف و ارائه ی دید ایستا از یک سیستم بکار می رود. این نمودار چگونگی تقسیم سیستم به اجزای آن و وابستگی بین اجزای سیستم را تشریح می نماید. نمودار component متشکل از مصنوعات و اجزای فیزیکی از قبیل کتابخانه ها، فایل ها، پوشه ها و غیره ... می باشد.

این نمودار از دیدگاه پیاده سازی و اجرایی مورد استفاده قرار می گیرد. به منظور نمایش گرافیکی کل یک سیستم به مجموعه ای از نمودارهای component نیاز است. با استفاده از تکنیک مهندسی رو به جلو و معکوس می توان از نمودار مزبور فایل های اجرایی (executable) ایجاد نمود.

## نمودار استقرار (deployment diagram)

نمودار های **deployment** سخت‌افزار بکار رفته در پیاده‌سازی سیستم و همچنین محیط‌های اجرا و سایر اجزایی که باید بر روی این سخت‌افزار قرار گیرند را شرح داده و به تصویر می‌کشد. این نمودارها عمدتاً مورد کاربرد مهندسين سیستم هستند.

نمودارهای **deployment** از گره‌ها (**node**) و رابطه‌ی میان آن‌ها شکل گرفته و ساخته می‌شود. یک نمودار **deployment** کارآمد، بخش جدایی‌ناپذیر چرخه‌ی توسعه‌ی نرم‌افزار کاربردی می‌باشد.

## نمودار مورد کاربرد (Use case)

نمودار **use case** به منظور نمایش ماهیت پویای یک سیستم بکار می‌رود. نمودار ذکر شده کارکرد ارائه شده توسط یک سیستم را در قالب عملگرها (**Actor**) و اهداف آنها که به صورت **use case** نمایش داده می‌شوند و وابستگی بین موردهای کاربرد، مدل و مصورسازی می‌کند. این نمودار از اجزایی همچون **use case** ها، **actor** ها و رابطه‌ی آن‌ها تشکیل می‌شود.

دیاگرام‌ها **use case** در سطح بالا برای مصورسازی کارکردهای سیستم و نیازهایی که برطرف می‌سازد، بکار می‌رود.

در خلاصه می‌توان گفت که نمودار **use case** کارکردهای سیستم و جریان آن‌ها را مدل‌سازی می‌کند.

## نمودار تعامل یا برهمکنش (interaction diagram)

نمودارهای **interaction** جهت ارائه‌ی نمایی از جنبه‌ی پویای سیستم بکار می‌رود. نمودارهای **sequence** و **collaboration** هر دو زیرمجموعه‌ی نمودارهای **interaction** هستند که برای نمایش گرافیکی ذات پویای سیستم بکار می‌روند. در واقع این نمودار بر گردش کنترل و داده‌ها بین چیزهای مختلف در سیستم مدل شده تاکید دارند.

نمودار **sequence** برای نمایش ترتیب زمانی بکار می‌رود یا به عبارتی دیگر نشان می‌دهد که اشیا چگونه با یکدیگر در قالب پیام‌هایی متوالی تعامل دارند و همچنین نمایشگر طول عمر اشیا نسبت به این پیام‌ها

می‌باشد. دیاگرام **collaboration** بیشتر بر روی رابطه بین اشیا تاکید دارد. در حالی که یک دیاگرام **sequence** اعمال اشیا را در یک توالی زمانی نشان می‌دهد و بر حسب زمان تنظیم می‌شود. برای نمایش کل یک سیستم معمولاً از ترکیبی از چندین نمودار **collaboration** و **sequence** استفاده می‌شود.

### نمودار حالت (statechart diagram)

نمودارهای **statechart** یکی از پنج دیاگرامی است که برای مدل‌سازی جنبه‌های پویای سیستم بکار می‌رود. این نمودارها به منظور مدل‌سازی کل چرخه‌ی حیات یک شی بکار می‌روند. نمودار گفته شده همان طور که از اسم آن پیداست مدلی از حالت‌های مختلفی که یک شی در آن قرار می‌گیرد، ارائه می‌دهد. در واقع این نمودار تصویری از چرخه حیات شی (**object life cycle**) را به نمایش می‌گذارد. نمودار **activity** زیرمجموعه‌ی این نمودار محسوب می‌باشد.

**State** را می‌توان وضعیت یا حالتی که شی در زمان مشخص در آن قرار می‌گیرد تعریف کرد. وضعیتی که شی در آن به سر می‌برد با فعال شدن **event**‌هایی تغییر می‌یابد. این نمودار برای مهندسی معکوس و رو به جلو نیز بکار می‌رود.

### نمودار activity

یکی دیگر از نمودارهای پرکاربرد و مهم **UML** که نمایشگر رفتار داینامیک یا پویای سیستم می‌باشد، نمودار **activity** است. این نمودار از عناصری نظیر **activity**‌ها (افعال سیستم)، پیوندها (**link**) و رابطه‌ها تشکیل می‌شود. نمودار ذکر شده همچنین انواع جریان‌ها همچون موازی (**parallel**)، واحد (**single**)، همروند (**concurrent**) و منشعب یا شاخه‌ای (**branched**) و غیره... را مدل‌سازی می‌کند.

نمودار مورد نظر کنترل جریان از یک **activity** به **activity** را بدون پیام‌ها نمایش می‌دهد. این نمودار برای توصیف قدم به قدم گردش کار تجاری و عملیاتی اجزا و مصنوعات سیستم استفاده می‌شود. این نمودار نمایشگر گردش کنترل در سراسر سیستم است.