**به‌نام خدا**

**امیرحسین کاظمی \_ تمرین UML Report**

**- 1 تعریف UML**

**UML (Unified Modeling Language)** یک زبان استاندارد برای مدل‌سازی سیستم‌های نرم‌افزاری است که در تحلیل، طراحی، و مستندسازی سیستم‌ها به کار می‌رود. این زبان کمک می‌کند تا اجزای مختلف یک سیستم به صورت گرافیکی و ساختارمند نمایش داده شوند.

**- 2 فهرست نمودارهای UML**

**UML** شامل **14** نمودار اصلی است که به دسته‌بندی‌های مختلف تقسیم می‌شود:

1. Use Case Diagram
2. Activity Diagram
3. State Diagram
4. Communication Diagram
5. Sequence Diagram
6. Timing Diagram
7. Interaction Overview Diagram
8. Profile Diagram
9. Composite Structure Diagram
10. Class Diagram
11. Package Diagram
12. Object Diagram
13. Component Diagram
14. Deployment Diagram

**3 –کاربردهای UML**

**UML** برای مدل‌سازی و طراحی سیستم‌ها به کار می‌رود. این زبان می‌تواند در تحلیل نیازمندی‌ها، طراحی معماری سیستم، توصیف تعاملات، و ایجاد مستندات پروژه‌های نرم‌افزاری استفاده شود.

**4 -مزایای UML**

استاندارد جهانی و قابل فهم برای تیم‌های توسعه نرم‌افزار

کمک به تحلیل و طراحی دقیق‌تر سیستم‌ها

امکان مستندسازی ساختار سیستم به صورت گرافیکی

کاهش پیچیدگی‌های پروژه‌های نرم‌افزاری

**5 - معایب UML**

پیچیدگی در یادگیری برای افراد مبتدی

ممکن است استفاده بیش از حد از نمودارها، به جای ساده‌سازی، باعث پیچیدگی بیشتر شود

در برخی پروژه‌های کوچک، استفاده از **UML** می‌تواند زمان‌بر و غیرضروری باشد

**6 - جنبه‌های ایستا و پویا**

**جنبه ایستا**: جنبه‌هایی که وضعیت ثابت سیستم را نشان می‌دهند. مانند ساختار کلاسی که روابط میان کلاس‌ها و اشیاء را نمایش می‌دهد.

**جنبه پویا**: جنبه‌هایی که رفتار سیستم در زمان اجرا را نشان می‌دهند. مانند نمودارهای Activity یا **Sequence** که جریان کنترل و تعاملات را نمایش می‌دهند.

**معرفی نمودارهای: UML**

**1 - Use Case Diagram**

**تعریف**: نمودار **Use Case** برای نمایش تعاملات کاربران با سیستم استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، یکی از **Use** **Case‌ها** می‌تواند "ثبت نام در کلاس" باشد که نشان می‌دهد یک دانشجو می‌تواند در کلاس‌ها ثبت‌نام کند.

**2 - Activity Diagram**

**تعریف**: این نمودار برای نمایش جریان فعالیت‌ها و کارهای مختلف در یک فرایند یا سیستم استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، فعالیت "ثبت‌نام در کلاس" ممکن است شامل فعالیت‌هایی مانند "جستجو برای کلاس" و "انتخاب کلاس" باشد.

**3 - State Diagram**

**تعریف**: نمودار **State** برای نمایش وضعیت‌های مختلف یک شیء و نحوه تغییر وضعیت آن استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجو، وضعیت "دانشجو" ممکن است از "ثبت‌نام نشده" به "ثبت‌نام شده" تغییر کند.

**4 - Communication Diagram**

**تعریف**: این نمودار برای نشان دادن تعاملات بین اشیاء در یک سیستم استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، تعاملات بین "دانشجو" و "سیستم" برای "ثبت‌نام در کلاس" نمایش داده می‌شود.

**5 - Sequence Diagram**

**تعریف**: نمودار **Sequence** برای نمایش ترتیب زمانی تعاملات بین اشیاء مختلف سیستم به کار می‌رود.  
**مثال**: در سیستم مدیریت دانشجویان، ترتیب زمانی فرآیند "ورود به سیستم" می‌تواند شامل تعاملات بین "دانشجو"، "سیستم احراز هویت" و "پایگاه داده" باشد.

**6 - Timing Diagram**

**تعریف**: این نمودار برای نمایش تغییرات وضعیت در طول زمان استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، نمایش زمانی فرآیند "پرداخت شهریه" ممکن است شامل تغییرات وضعیت از "در حال انتظار" به "پرداخت شده" باشد.

**7 - Interaction Overview Diagram**

**تعریف**: این نمودار ترکیبی از نمودارهای **Activity** و **Sequence** است که برای نمایش جریان‌های مختلف در یک فرآیند به کار می‌رود.  
**مثال**: در سیستم مدیریت دانشجویان، این نمودار می‌تواند روند کلی "ثبت‌نام دانشجو" را با استفاده از سایر نمودارهای جزئی‌تر مانند **Activity** و **Sequence** نشان دهد.

**8 - Profile Diagram**

**تعریف**: این نمودار برای تخصیص ویژگی‌های خاص به عناصر مدل استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، استفاده از پروفایل برای تخصیص سطح دسترسی‌های مختلف به کاربران مانند "مدیر" یا "دانشجو".

**9 - Composite Structure Diagram**

**تعریف**: این نمودار برای نمایش ساختار داخلی اجزای سیستم و نحوه ارتباط آن‌ها استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، این نمودار می‌تواند ساختار داخلی "سیستم ثبت‌نام" را نشان دهد.

**10 - Class Diagram**

**تعریف**: این نمودار برای نمایش ساختار کلاس‌ها و روابط بین آن‌ها در یک سیستم استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، نمودار کلاس می‌تواند کلاس‌هایی مانند "دانشجو"، "کلاس"، و "استاد" را نشان دهد.

**11 - Package Diagram**

**تعریف**: این نمودار برای نمایش گروه‌بندی کلاس‌ها و نحوه تقسیم سیستم به بخش‌های مختلف استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، یک بسته می‌تواند مربوط به "مدیریت دانشجویان" و دیگری مربوط به "مدیریت کلاس‌ها" باشد.

**12 - Object Diagram**

**تعریف**: نمودار **Object** برای نمایش نمونه‌های اشیاء و روابط آن‌ها در یک نقطه زمانی خاص استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، می‌توان نمونه‌ای از "دانشجو" و "کلاس" را به صورت اشیاء نمایش داد.

**13 - Component Diagram**

**تعریف**: نمودار **Component** برای نمایش اجزای مختلف سیستم و نحوه تعامل آن‌ها با هم استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، اجزای مختلف مانند "پایگاه داده"، "سیستم احراز هویت" و "سیستم مدیریت کلاس‌ها" می‌توانند به صورت کامپوننت‌ها نمایش داده شوند.

**14 - Deployment Diagram**

**تعریف**: این نمودار برای نمایش نحوه قرار گرفتن اجزای سیستم بر روی سخت‌افزار استفاده می‌شود.  
**مثال**: در سیستم مدیریت دانشجویان، نمودار می‌تواند نحوه نصب و قرارگیری "پایگاه داده" و "وب‌سایت" را روی سرورهای مختلف نشان دهد.