

چراغ دل به نور جان برافروخت

به نام آنکه جان را فکرت آموخت



آزمایشگاه مهندسی نرم افزار

مدرس : مریم ربیعی

نیمسال دوم سال تحصیلی ۹۸-۹۹



تئوری سیستمها

- روشی برای شناخت دنیای واقعی است.
- برنامه نویسان نیاز به شناختن محیطی که کاربران با آن کار می کنند دارند.
- برای شناخت محیط باید نگرش سیستمی به مسئله داشته باشیم.
- چند مثال از انواع سیستم شامل: سیستم حسابداری، سیستم ثبت نام، سیستم کنترل ترافیک و ...



خصوصیات سیستم‌ها

۱. Environment یا محیط

هر سیستم درون یک محیط قرار دارد.

Environment

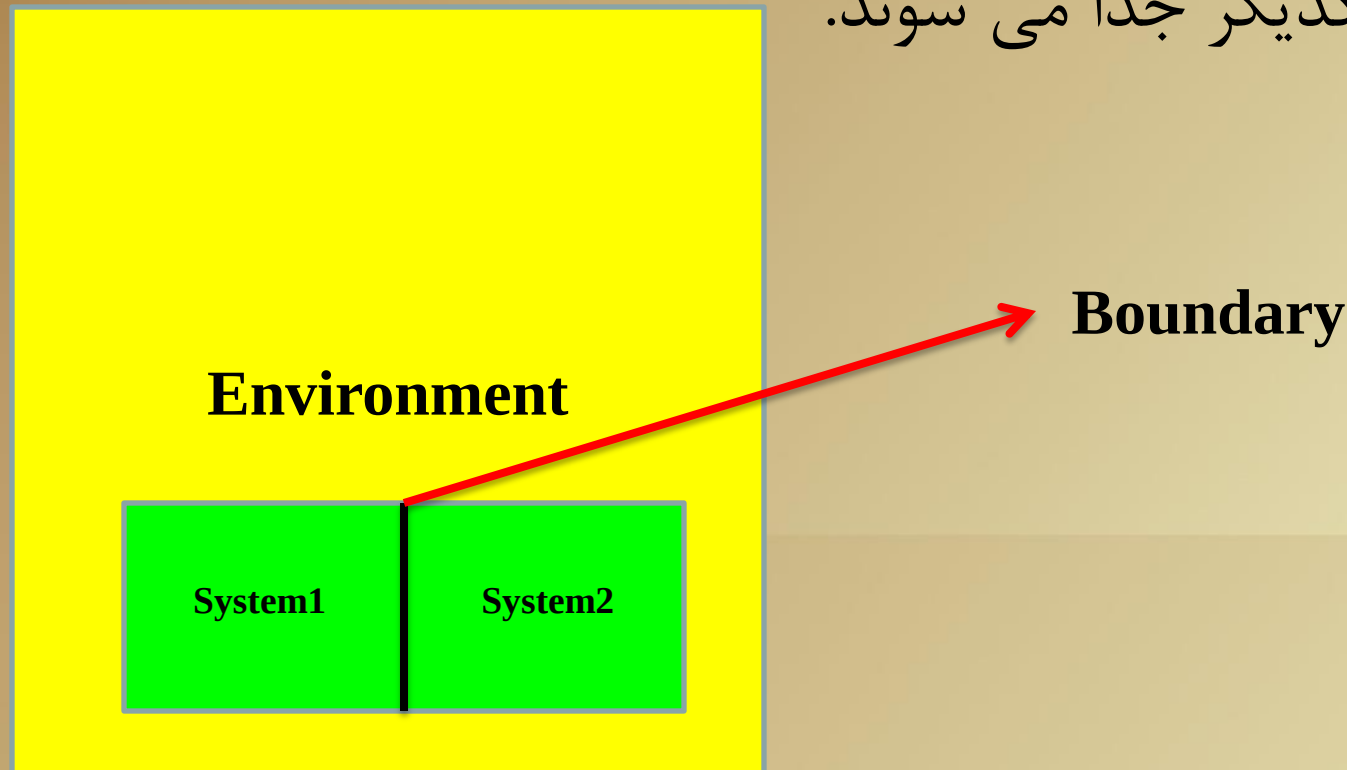
System



...خصوصیات سیستمها

۲. Boundary یا مرز

سیستمهای موجود در یک محیط توسط boundary از یکدیگر جدا می شوند.

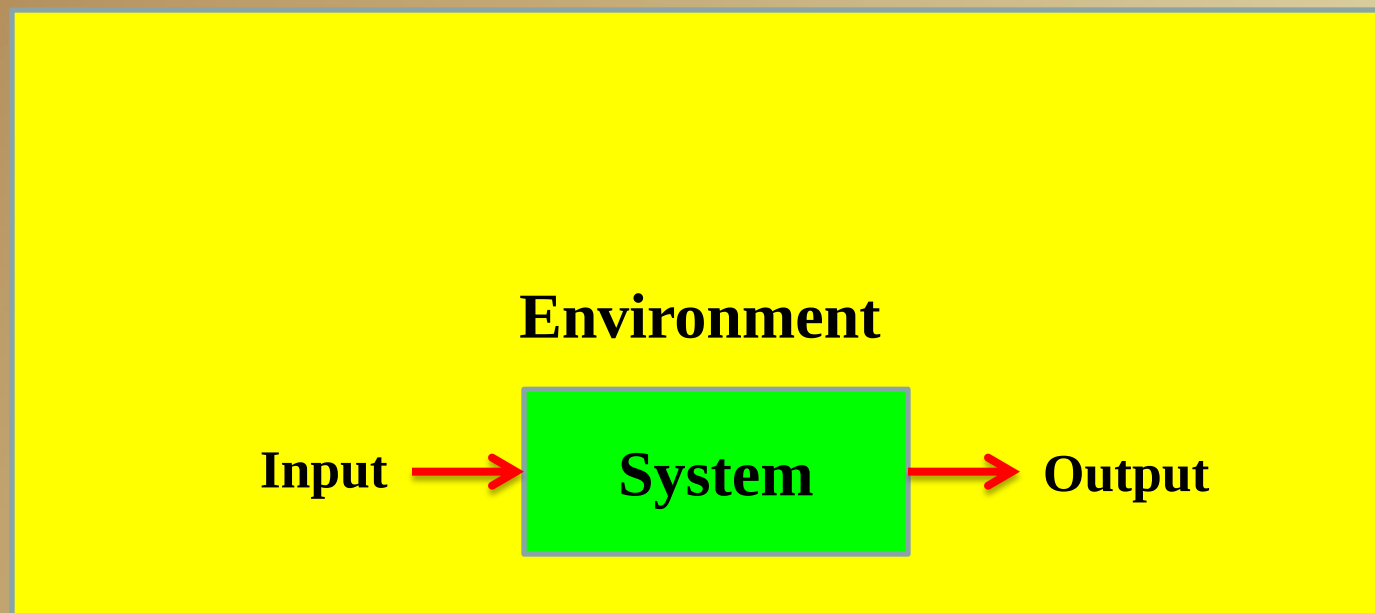




...خصوصیات سیستم‌ها

۳. Input و Output یا ورودی و خروجی

هر سیستم ورودی‌هایی را از محیط می‌گیرد و خروجی‌هایی را به محیط پس می‌دهد.



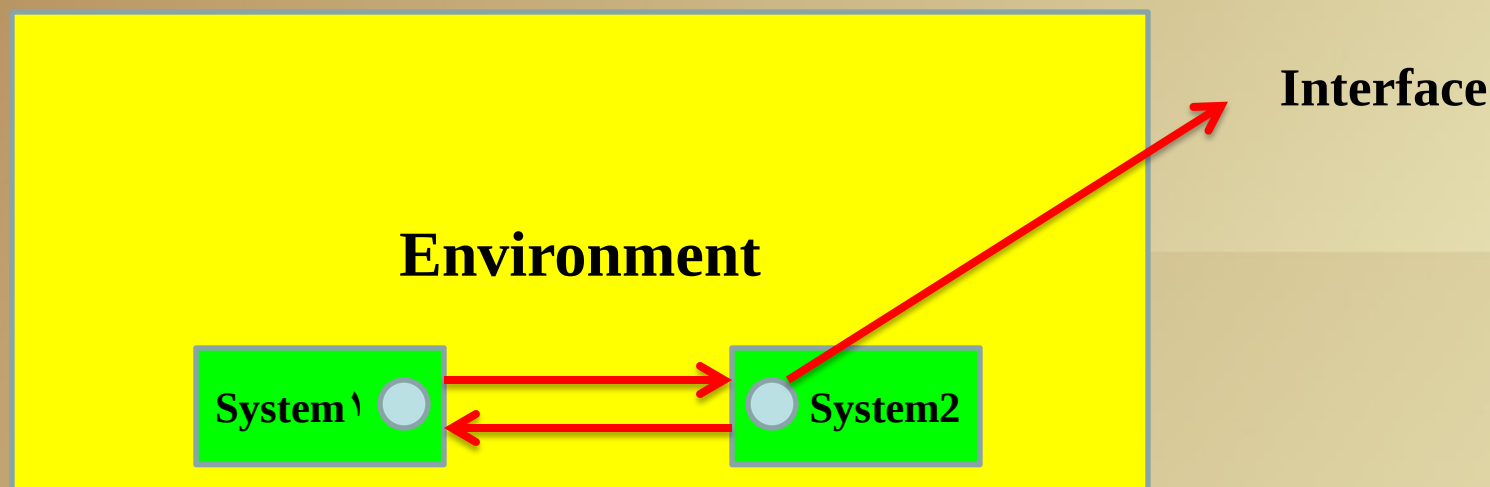


...خصوصیات سیستمها

۴. Interface یا رابط

هر سیستم با استفاده از رابط با سیستم دیگر ارتباط برقرار می کند.

زبانی است که سیستم بوسیله آن با سیستمهای دیگر، درون محیط محاوره می کند.





...خصوصیات سیستمها

۵. Subsystem یا زیر سیستم

هر سیستم می تواند از تعدادی زیر سیستم تشکیل شده باشد. زیر سیستمها تمامی خصوصیات یک سیستم کامل را دارا می باشند.

Environment

System

Subsystem1

Subsystem2



چرا UML؟

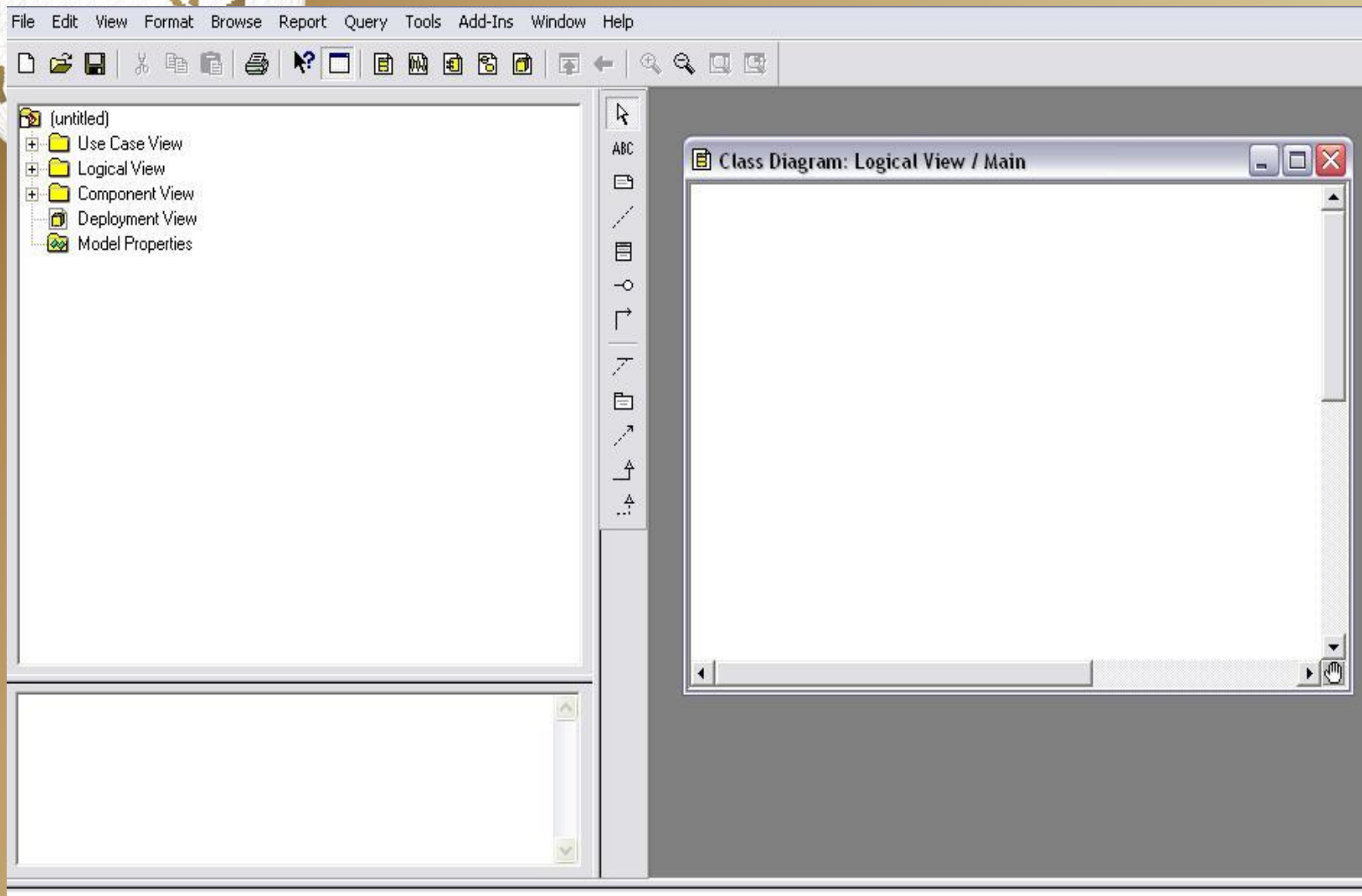
- مخفف Unified Modeling Language
- یک زبان تصویری برای تولیدکنندگان نرم افزار است.
- استاندارد برای طراحی و پیش بینی جزئیات فنی سیستم
- نرم افزاری و نحوه ارتباط اجزاء مختلف سیستم است.



تعریف Business

- مجموعه فعالیتهایی هستند که افراد یک سازمان برای پیش برد اهداف سازمان انجام می دهند.
- تمام کارها و فعالیتهایی که اتفاق می افتد تا کسب و کار یک سازمان اجرا شود.

آشنایی با محیط رشنال رز





...آشنایی با محیط رشنال رز

• انواع View:

۱. Use Case View

۲. Logical View

۳. Component View

۴. Deployment View



Use Case View

- در برگیرنده کل کارهایی است که سیستم قرار است انجام دهد.
- در فارسی به معنی مورد کاربرد است.
- هر UseCase نشان دهنده سرویسی است که سیستم در اختیار کاربران قرار می دهد. کاربر می تواند شامل یک فرد یا یک سیستم باشد.
- از بالا تمام فعالیتهای سیستم را مدل می کند.



Logical View

- نقشه اصلی سیستم یا طرحی که باید پیاده سازی شود.
- آنچه در UseCase View نمایش داده شده اکنون تخصصی تر می شود.
- با مفاهیم برنامه نویسی بیشتر سروکار خواهیم داشت.
- عناصر آن شامل Class ها به همراه Attribute ها و متدها هستند.
- اصلی ترین کار: نمایش ارتباط واقعی میان کلاس ها در سیستم است.



Component View

- کلاسهای ایجاد شده در مرحله قبل باید به Component تبدیل شوند.
- ارتباط میان Component ها باید مشخص شود.
- عمده فعالیتها در این View باید به کدهای برنامه نویسی تبدیل شود.
- وابستگی زیاد به بستر برنامه نویسی دارد.



Deployment View

- در این View باید محل قرار گرفتن Component های مرحله قبل را دقیقاً مشخص کنیم.
- به بررسی بستر نرم افزاری و سخت افزاری سیستم می پردازد.
- مورد استفاده افرادی است که قصد توسعه یک نرم افزار قدیمی را دارند.



معرفی کامل UseCase View

• مفاهیم اولیه:

۱. UseCase

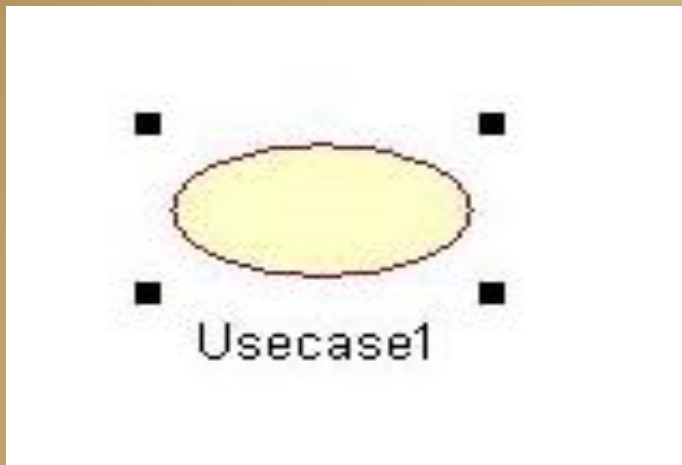
۲. سناریو

۳. Actor



۱. UseCase

- سرویسی است که هر سیستم در اختیار کاربران قرار می دهد.





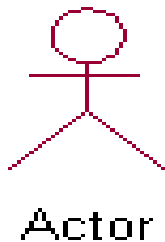
۲. سناریو

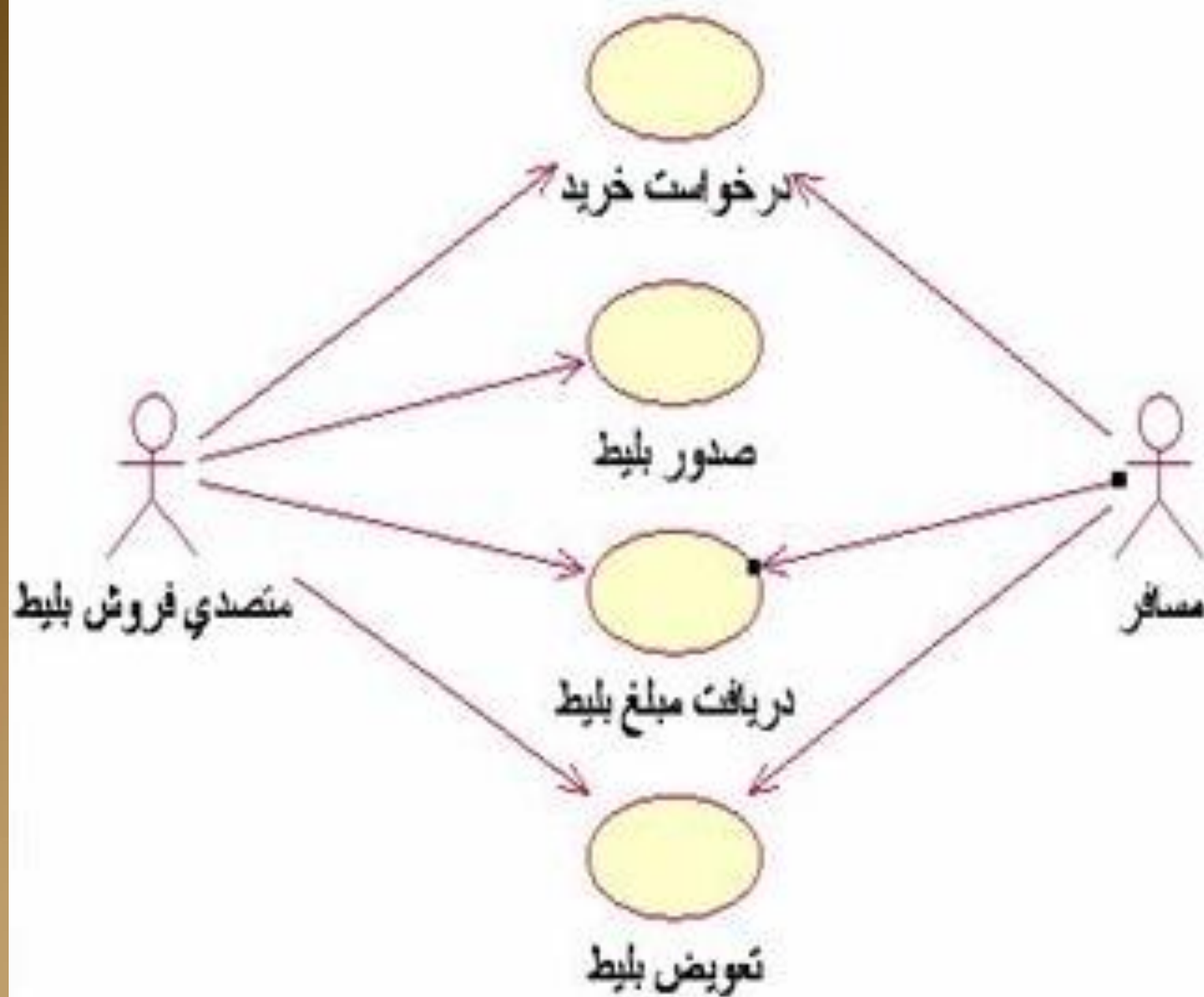
- متنی است که فعالیتهای UseCase را بطور کامل شرح می دهد.
- هر سناریو دارای استانداردهایی است که در ادامه توضیح خواهیم داد.



۳. Actor

- اطلاعاتی را از UseCase دریافت می کنند، و یا اطلاعاتی را به آن تزریق می کنند.

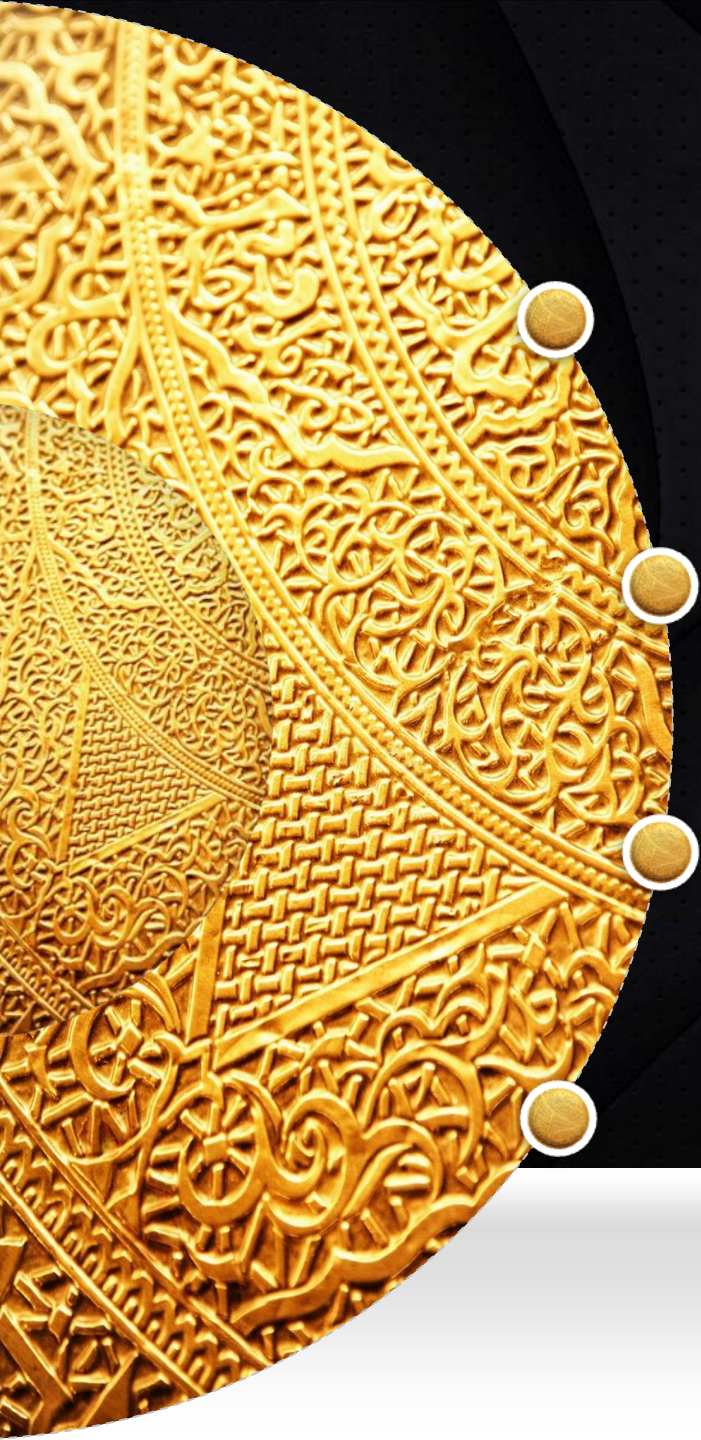






...هدف UseCase Diagram

- UseCase دیاگرام نقطه ورود یک سیستم است.
- مسئله تحلیل می شود و در اختیار طراحان و برنامه نویسان قرار می گیرد.
- Actorها و UseCaseها و روابط میان آنها باید مشخص شوند.



Thank You
End of Session1



انواع Actorها

- کسانی هستند که با سیستم کار می کنند.
- از آن اطلاعات می گیرند یا به آن اطلاعات می دهند.
- کسی است که UseCase در جهت سرویس دادن به آن حرکت می کند.
- Actorها مهم هستند چون سیستم ساخته شده باید جوابگوی نیازهای آنها باشد.
- اولین قدم برای تهیه نیازمندی های سیستم پیدا کردن Actorها است.

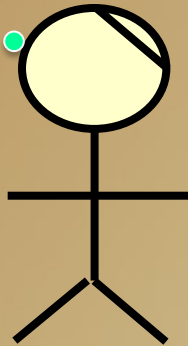
...انواع Actorها



فرد یا سیستمی که از
نتیجه کار سازمان بهره
مند می شود. (مشتری)

سیستم است و ممکن
از داخل سیستم باشد.

فرد یا سیستمی که با نرم
افزار کار می کند.
(متصدی)



Business Actor



Actor

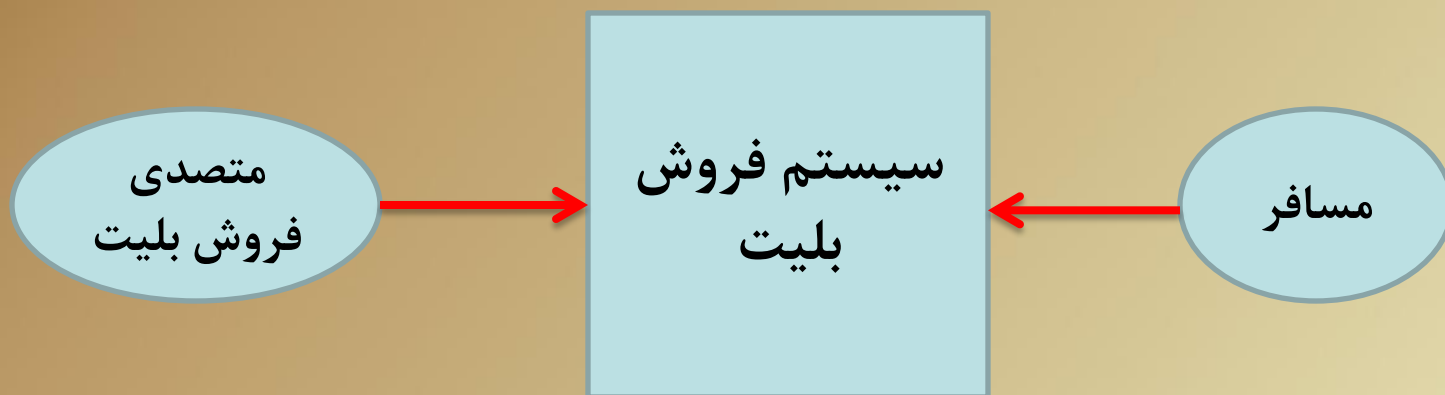


چگونه Actorها را بیابیم؟

- زمانیکه بدانید Actor کیست و چه نیازهای دارد براحتی می توان UseCaseهای موجود در یک سیستم را استخراج کرد.
- طراحی نمودار Context جهت استخراج Actorها
- Context Diagram جزو نمودارهای UML نیست.



...چگونه Actorها را بیابیم؟





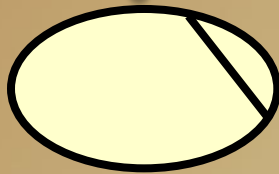
موجودیت خارجی یا External Entity

- در نمودار Context Diagram موجودیتهایی که با سیستم در ارتباط هستند را موجودیت خارجی می گویند.
- هر کس که ذینفع سیستم است و به سیستم نفع می رساند را موجودیت خارجی می گویند. مانند مسافر و متصدی
- موجودیتهای خارجی کاندیدهایی برای Actorها در سیستم باشند.



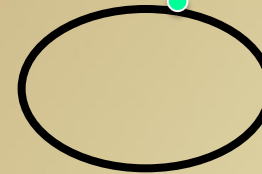
انواع Usecase

فعالیت‌هایی است که توسط
سازمان انجام می‌شود.
نتیجه خاصی را برای کاربر
دارد.



Business Usecase

یک فعالیت نرم افزاری
است که کار خاصی را برای
کاربر انجام می‌دهد.



Usecase



نتیجه گیری

- نقطه شروع برای پیاده سازی یک سیستم تشخیص نیازمندیهای آن سیستم است.
- نقطه شروع برای استخراج نیازمندیها، استخراج Actorها می باشد.



سناریو

- سناریو متنی است که فعالیتهای UseCase را شرح می دهد.
- باید جزئیات UseCase را شرح دهیم.
- درون UseCase چه کارهایی باید انجام شود تا در جهت نیازهای کاربران بکار رود.
- باید تمام عملیات درون UseCase را مرحله به مرحله شرح دهیم.



واسط کاربر User Interface

- باید دقیقا ذکر شود که واسط کاربر ما چگونه کار می کند.
- واسط کاربر را بصورت Prototype (نمونه آزمایشی) تعریف می کنند و با مشتری بر سر آن به توافق می رسند.



استانداردهای سناریو

۱. Name: نام UseCase
۲. Precondition: شرایطی که باید ایجاد شود تا UseCase فعال گردد را گویند.
۳. Post Condition: شرایطی که بعد از اتمام UseCase انجام می شود را گویند.
۴. Goal: هدف UseCase. (UseCase برای چه منظوری ایجاد شده است).
۵. Description: شرح مختصری از فعالیتهای سیستم را بیان می کند.



...استانداردهای سناریو

۶. Main Flow: جریان اصلی کار Usecase را مورد به مورد و بصورت الگوریتمی بیان می کند.

۷. Alternative Flow: جریان فرعی Usecase را شرح می دهد. یعنی روند غیر روتین، خطاها و exception ها را شرح می دهد.

استانداردهای اصلی شامل: Name, Precondition, Postcondition, Main Flow و Alternative Flow



مثال: سناریو صدور بلیت

• Name

ذکر شده در عنوان

• Precondition

مشتری باید به باجه فروش بلیت مراجعه کند.

• Post condition

باید نام مشتری در بین مسافری درج شود و بلیت برای او صادر شود.

• Goal

صدور بلیت برای مشتری



...مثال

• Description

مشتری با مراجعه به متصدی و پرداخت مبلغ، بلیت را از باجه دریافت می کند.

• Main Flow

۱. مشتری به متصدی فروش بلیت مراجعه می کند.

۲. مشتری ساعت، روز و مقصد خود را به متصدی صدور بلیت اعلام می کند.

۳. متصدی به منوی برنامه حرکت رفته و بر اساس درخواست مسافر برای او یک بلیت صادر می کند.



...مثال

• Alternative Flow

۱. مشتری ساعت و روز خاصی را در نظر نداشته باشد و برنامه حرکت را در یک بازه زمانی درخواست کند. پس متصدی باید بتواند در منوی برنامه حرکت عملیات جستجو را انجام دهد.
۲. مطابق با نظر مشتری جای خالی پیدا نشود، پس متصدی باید بتواند انتخابهای دیگری را در اختیار مشتری قرار دهد.
۳. مشتری یکی از کارمندان شرکت است و تخفیف ویژه شامل حال او می شود، پس متصدی باید بتواند به بعضی افراد تخفیف بدهد.



نتیجه گیری

- برای هر Usecase یک سناریو باید طراحی شود.
- هر چه یک سناریو کامل تر باشد استخراج نیازمندی ها از آن آسانتر صورت می گیرد.



آغاز کار با Rashnal Rose

Usecase Diagram

۱. Usecase.

Name: نام مورد کاربرد

Stereotype: نوع مورد کاربرد

Diagram: انواع دیاگرامهای مربوط به مورد کاربرد را نشان می دهد.

Relations: انواع ارتباطات مورد کاربرد با سایر عناصر

Files: انواع فایل‌های مورد کاربرد مانند سناریوها

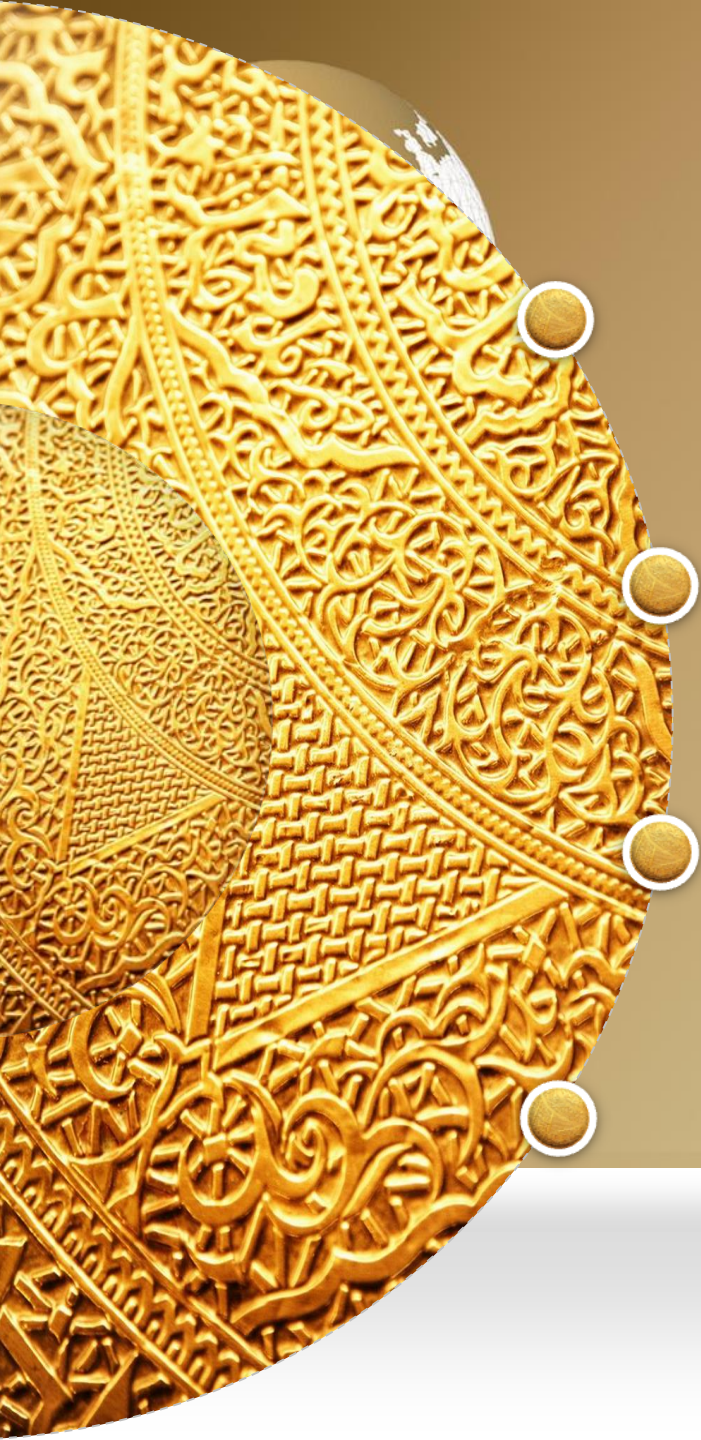


...آغاز کار با Rashnal Rose

۲. Actor

Name: نام

Stereotype: نوع Actor را مشخص می کند.



Thank You
End of Session2



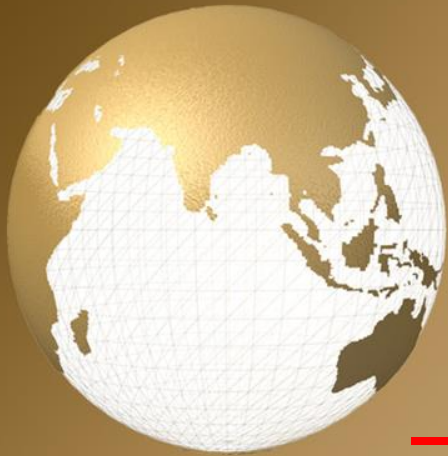
انواع روابط

→ Association ۱.

- رابطه بین Usecase و Actor است.
- ترجیحا از این رابطه بین Usecase ها استفاده نشود زیرا فهم مسئله را دشوار می کند.

- - - - - → Dependency ۲.

- Include
- Extended



...انواع روابط

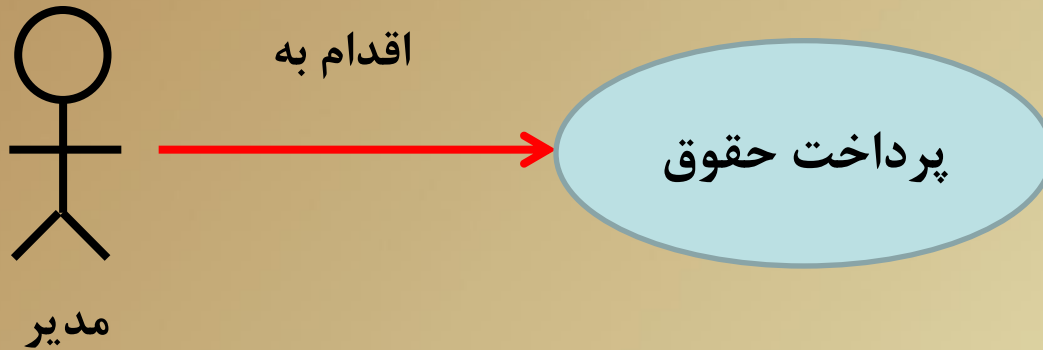
۳. Generalization یا ارث بری

یک Usecase می تواند تمامی خواص خود را از Usecase دیگر به ارث ببرد و در عین حال خواص خود را نیز داشته باشد.



Association

• مثال

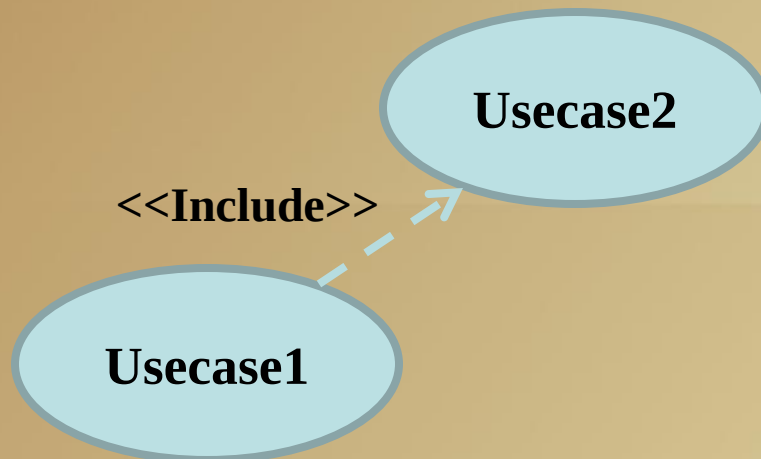




Dependency

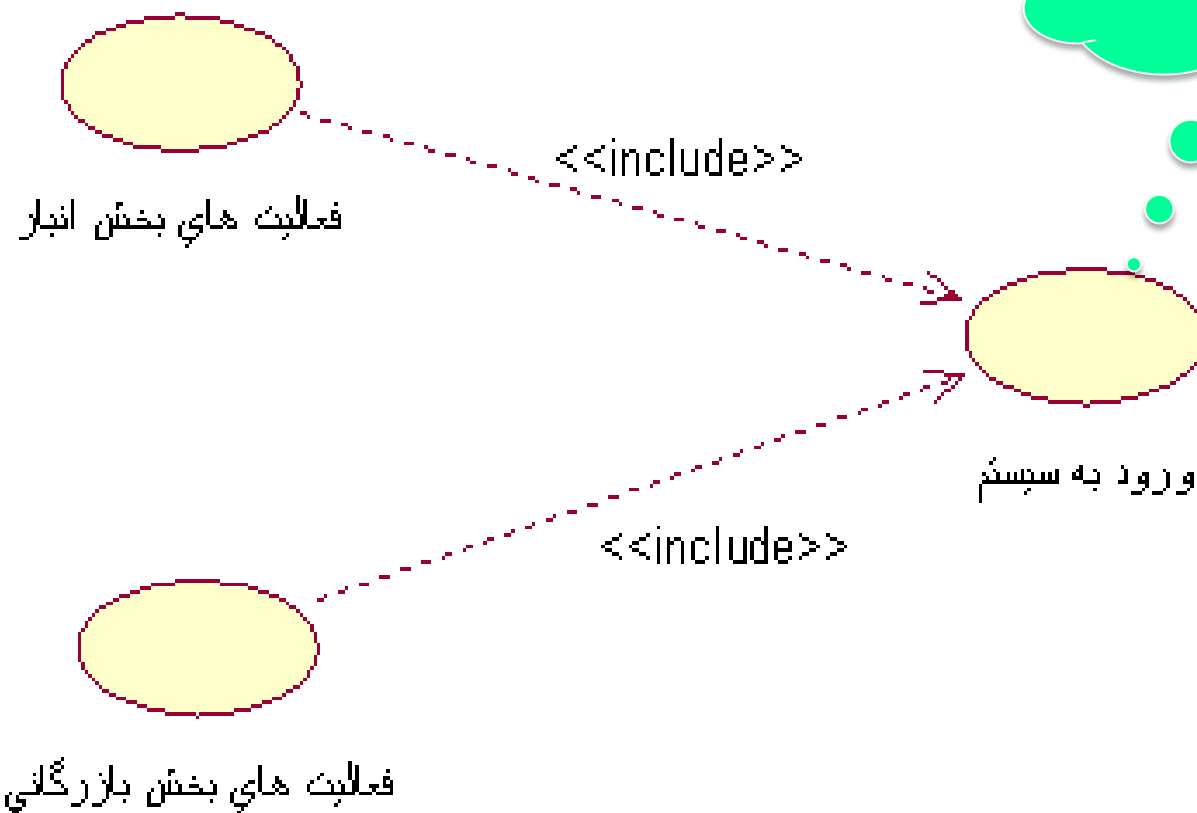
• Include

- Usecase1 از عملیاتی که در Usecase2 صورت می گیرد، استفاده می کند تا کار خود را پیش ببرد.
- یک Usecase سرویسی را ایجاد می کند که سایر Usecase ها می توانند از آن استفاده کنند. این نوع ارتباط نیاز همیشگی نیست و دلخواه است. مثلاً مدیر می تواند به صورت دلخواه گزارشهای هفتگی را چاپ کند.



مثال

نام کاربری و رمز
عبور

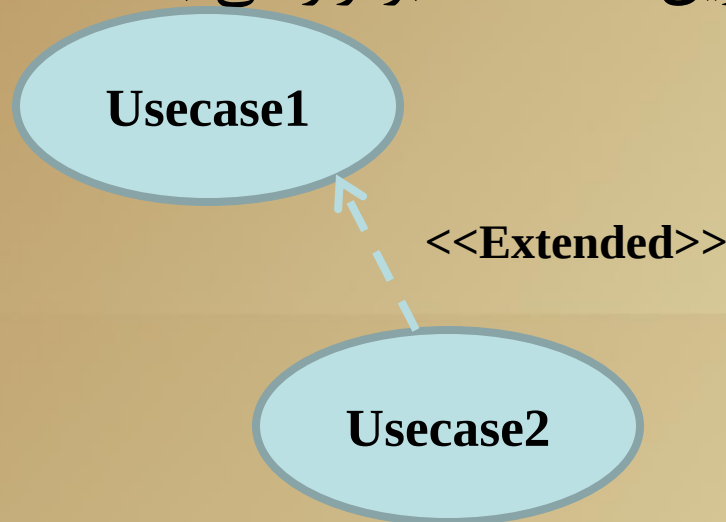




Dependency...

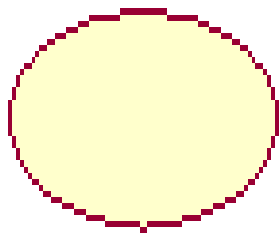
Extended •

- یک Usecase در شرایطی خاص از عملیات Usecase دیگر استفاده می کند.
- این نوع رابطه از طریق if-then برقرار می باشد.

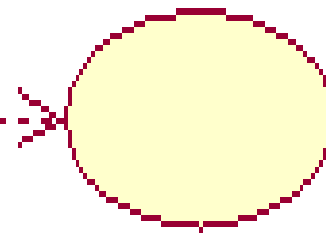


مثال

<<extend>>



دریافت مقاله

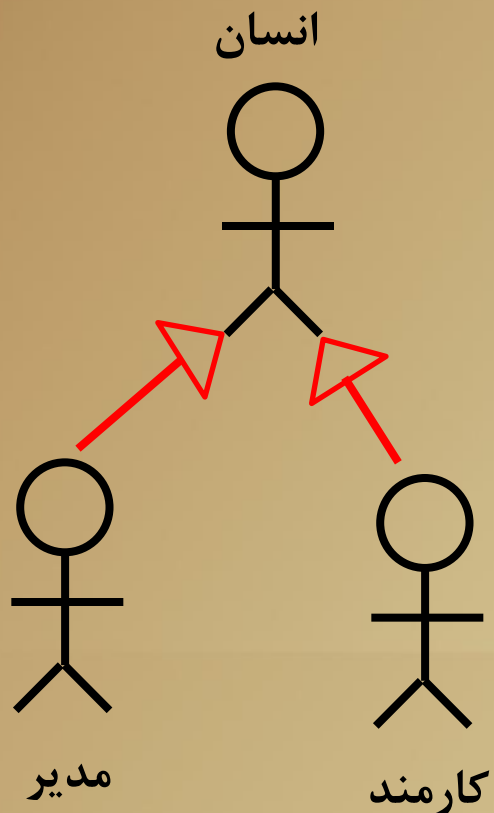


دریافت مقاله از طریق Email



Generalization یا ارث بری

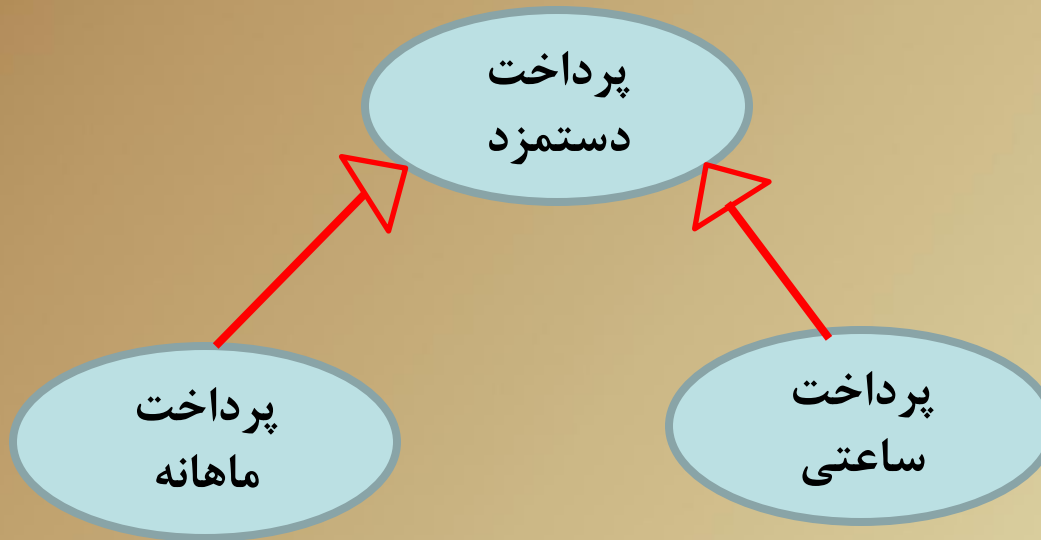
• Generalization بین Actor ها



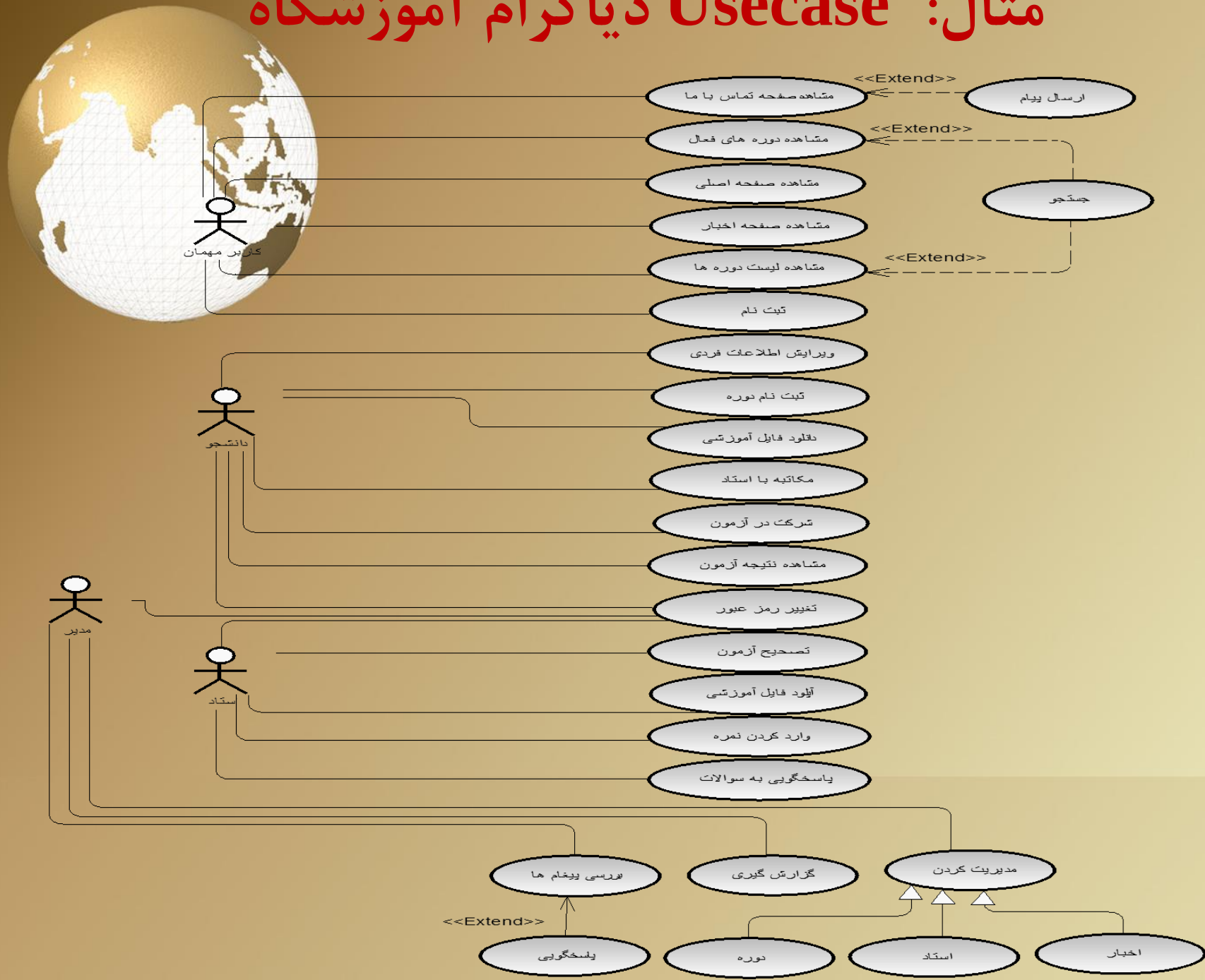
Generalization... یا ارث بری



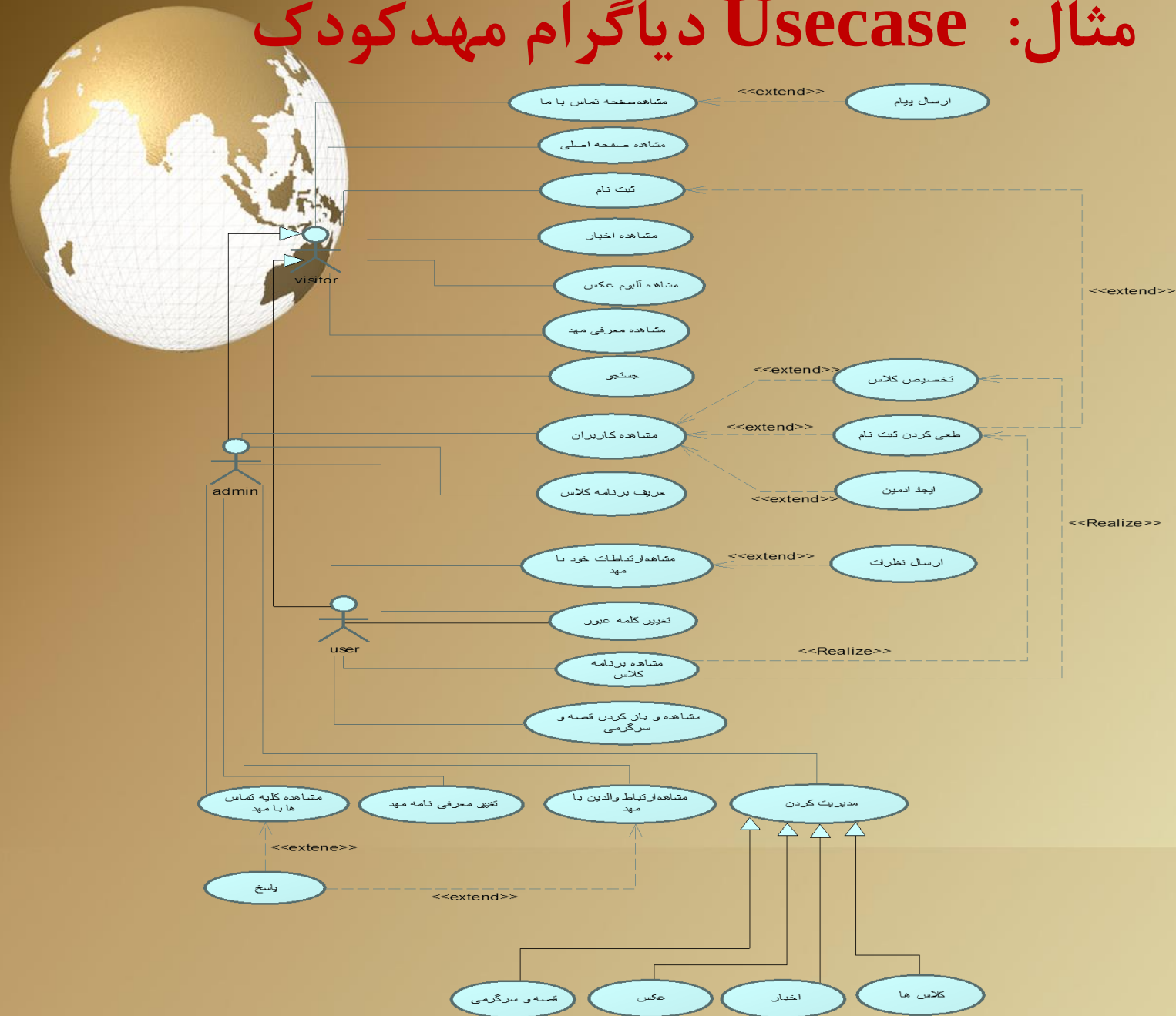
• Generalization بین Usecase ها

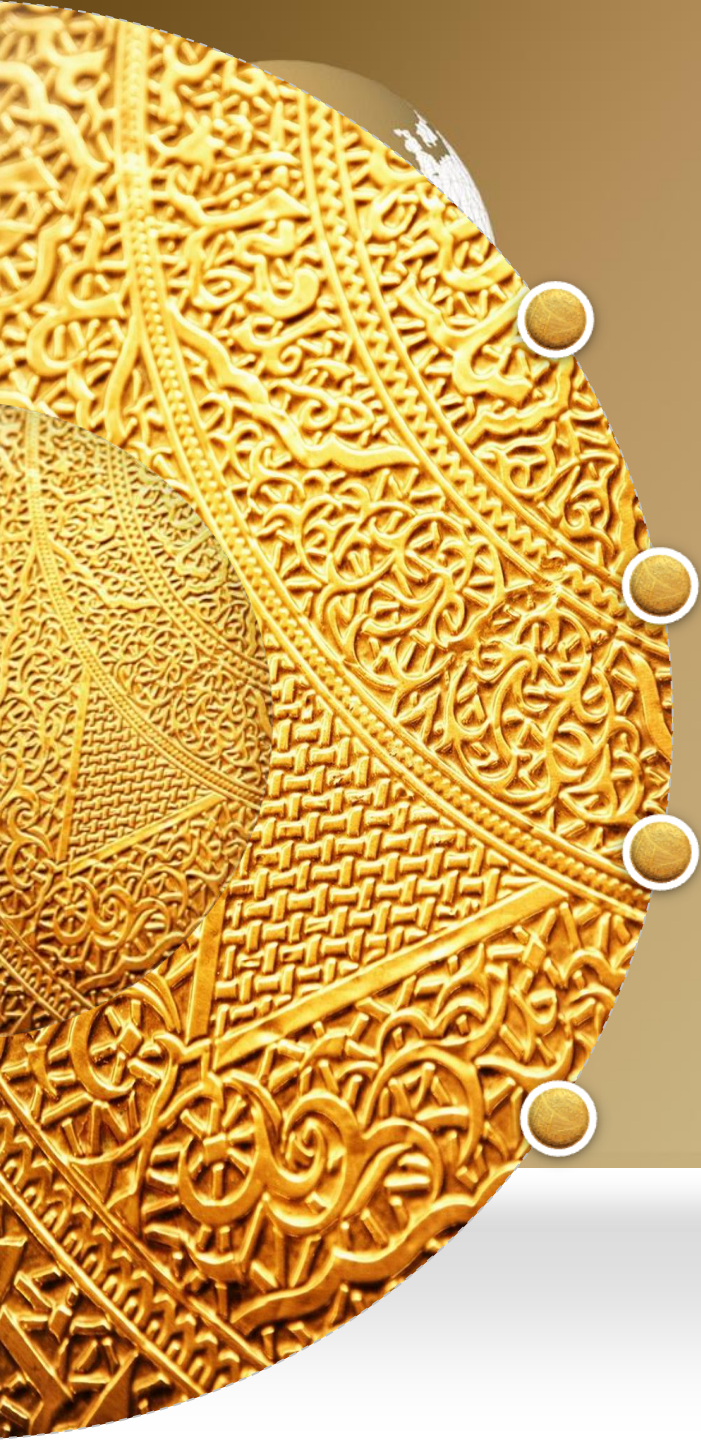


مثال: Usecase دیاگرام آموزشگاه



مثال: Usecase دیاگرام مہد کودک





Thank You
End of Session3