



# مرحله تحلیل نیازمندی ها USDP

---

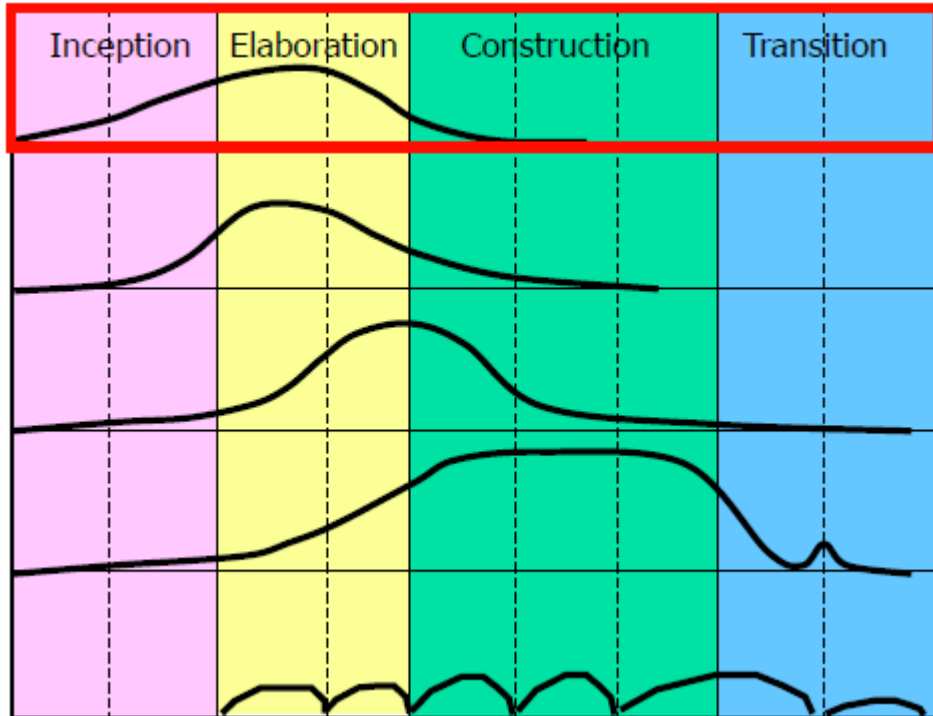
طراحی شی گرای سیستمها

دکتر قاسمی

دانشگاه آزاد تهران شمال

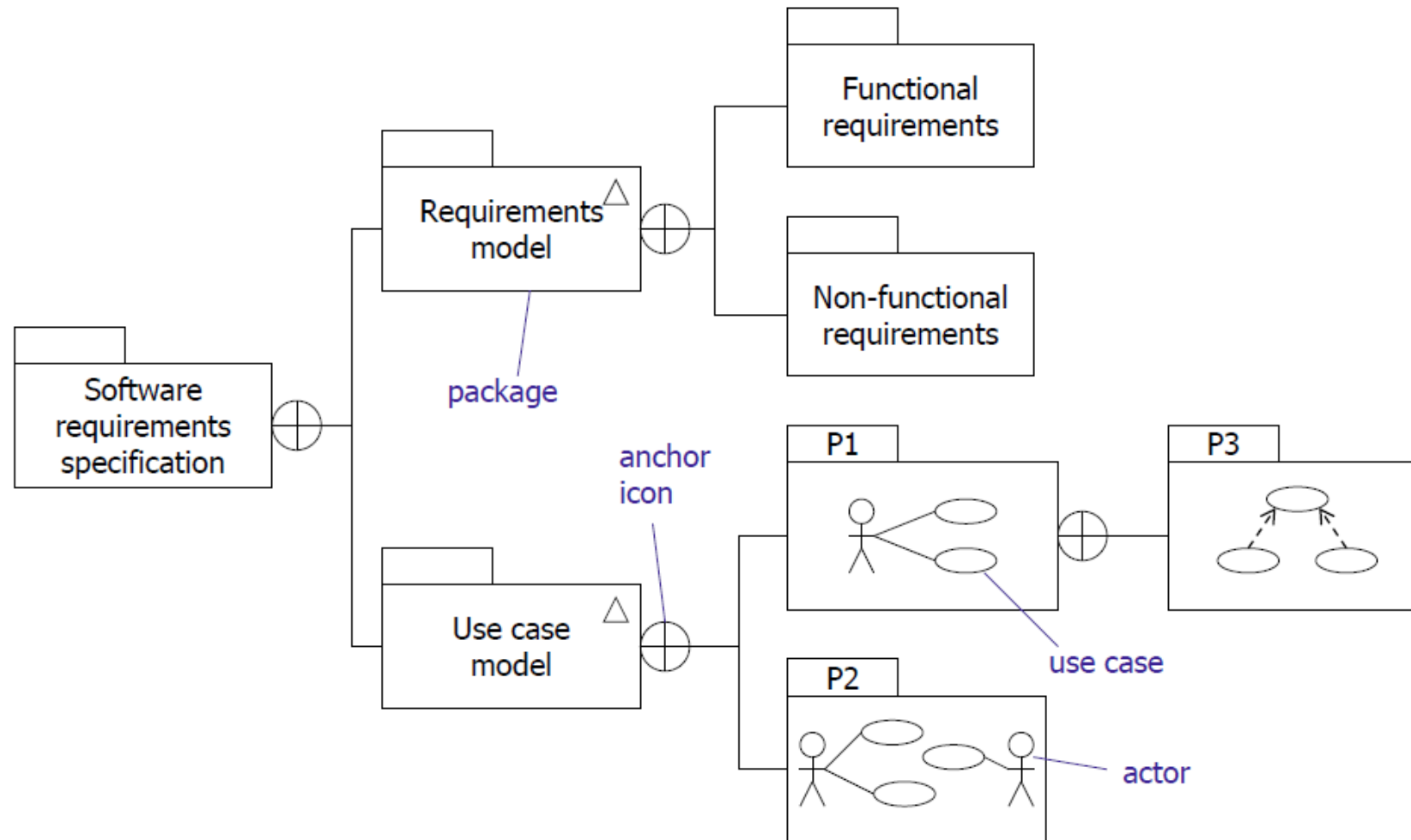
# Requirements

❖ هدف گردش کار نیازمندی ها ایجاد مشخصات سطح بالایی از آنچه باید اجرا شود است



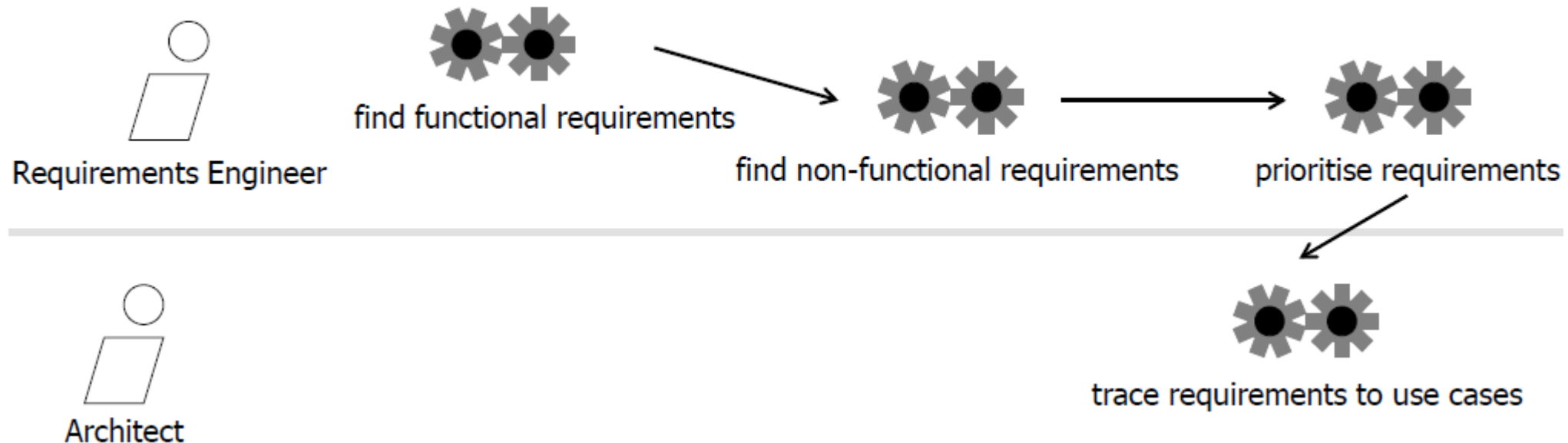
❖ ما با ذینفعان مصاحبه می کنیم تا دریابیم که آنها به چه چیزی نیاز دارند که سیستم برای آنها انجام دهد - الزامات آنها

# Requirements -metamodel



# جریان کاری Requirements

- ❖ به منظور اتخاذ یک رویکرد دقیق به نیازمندی‌ها، باید گردش کار پایه UP را با استخراج نیازمندی‌های کاربردی و غیرعملکردی و قابلیت ردیابی نیازمندی‌ها گسترش دهیم.
- ❖ نیازمندی‌های کارکردی : قابلیت‌های کارکردی سیستم چیست ؟ مثال : سیستم باید امکان آپلود عکس داشته باشد
- ❖ نیازمندی‌های غیر کارکردی : سیستم باید چه ویژگی‌های کیفی داشته باشد: مثال سیستم باید امن باشد



# نیازمندیها چیستند

❖ الزامات Requirements - "مشخصات آنچه باید اجرا شود":

- سیستم چه رفتاری باید ارائه دهد
- یک ویژگی خاص از سیستم
- محدودیت در سیستم

❖ در UP ما سند یک مشخصات مورد نیاز نرم افزار Software Requirements Specification SRS ایجاد می کنیم.

- بیانیه ای از الزامات سیستم برای همه ذینفعان است
- الزامات مربوطه را به بخش ها سازماندهی می کند
- مدل الزامات شامل الزامات عملکردی و غیر عملکردی است
- مدل مورد کاربرد Use Case شامل بازیگران و موارد استفاده

## نحوه نوشتن نیازمندی ها

<کد یکتا> : <سیستم> باید <عملکرد داشته باشد>  
۳۲: سیستم ATM باید ۴ رقم کلمه عبور دریافت کند

❖ الزامات نوشتن

❖ هیچ روش استاندارد UML برای نوشتن الزامات وجود ندارد!

❖ ما ساختار جمله یکسان را در بالا توصیه می کنیم

❖ الزامات عملکردی - آنچه که سیستم باید انجام دهد

• "سیستم خودپرداز باید تسهیلاتی را برای احراز هویت یک کاربر سیستم فراهم کند."

❖ الزامات غیر عملکردی - یک محدودیت در نحوه اجرای الزامات عملکردی

• "سیستم ATM باید یک مشتری را در چهار ثانیه یا کمتر احراز هویت کند"

# مرحله ۱ از پروژه

با استفاده از Power Designer برای پروژه خود موارد زیر را انجام دهید:

❖ لیست نیازمندی ها را تولید کنید

- Functional requirements

- Non-functional requirements

❖ فایل خروجی گزارش را بسازید

❖ به ایمیل [adftalent@gmail.com](mailto:adftalent@gmail.com) ارسال کنید فرمت فایل نام و درس و شماره تمرین

❖ Aliahmadi\_oop\_hw1.zip

# تبدیل نیازمندی ها به Use Case

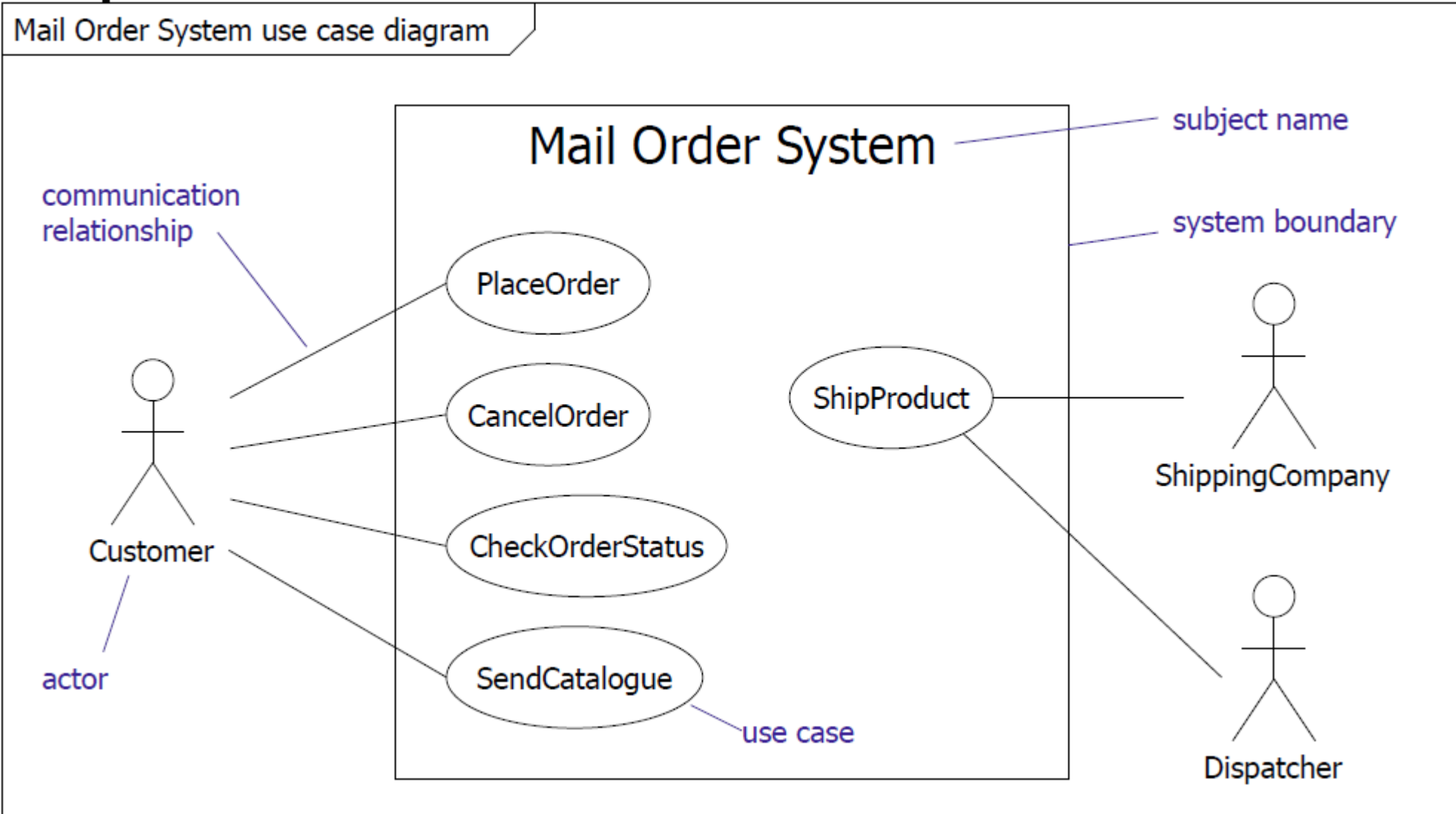
❖ مدل سازی مورد استفاده شکلی از مهندسی نیازمندی ها است

❖ مدل سازی مورد استفاده به شرح زیر است:

- مرز سیستم را پیدا کنید system boundary
- بازیگران را پیدا کنید actors چه کسی این کار را می کند
- موارد کاربرد را پیدا کنید use cases = همان جمله نیازمندی است فقط از حالت امری به حالت جمله مضارع
- برای هر Use Case ها مشخصات و سناریوها را بنویسید

❖ این به ما امکان می دهد مرز سیستم را شناسایی کنیم، چه کسی یا چه چیزی از سیستم استفاده می کند و چه عملکردهایی باید ارائه دهد

# مثال از Use Case Diagram



## Project Glossary

### Term1

Definition  
Synonyms  
Homonyms

### Term2

Definition  
Synonyms  
Homonyms

### Term3

Definition  
Synonyms  
Homonyms

...

❖ در هر حوزه کاری همیشه مقدار مشخصی از اصطلاحات خاص وجود دارد. مهم است که زبان دامنه را در واژه نامه پروژه ثبت کنید

❖ هدف از واژه نامه تعریف اصطلاحات کلیدی و حل مترادف ها و همنام ها است

❖ شما در حال ساختن واژگانی هستید که می توانید از آن برای بحث در مورد سیستم با ذینفعان استفاده کنید

# Actor کیست؟

- ❖ هنگام شناسایی بازیگران باید پرسید :
- چه کسی یا چه چیزی از سیستم استفاده می کند؟
  - آنها چه نقشی در تعامل دارند؟
  - چه کسی سیستم را نصب می کند؟
  - چه کسی سیستم را راه اندازی و خاموش می کند؟
  - چه کسی سیستم را حفظ می کند؟
  - چه سیستم های دیگری از این سیستم استفاده می کنند؟
  - چه کسی اطلاعات را دریافت کرده و در اختیار سیستم قرار می دهد؟
  - آیا چیزی در یک زمان مشخص اتفاق می افتد؟

## Use Case ها کدامند

- ❖ مورد کاربرد : کاری است که یک بازیگر به سیستم نیاز دارد تا انجام دهد. این یک "مورد استفاده" از سیستم توسط یک بازیگر خاص است
- موارد استفاده همیشه توسط یک بازیگر شروع می شود
  - بازیگر اصلی مورد کاربر UseCase را تحریک می کند (فراخوانی می کند یا اجرا می کند)
  - موارد کاربرد همیشه از دیدگاه بازیگران نوشته می شود ( جمله ما = کاری که بازیگر Actor انجام می دهد)

## مورد کاربرد: جریان کاری

- ❖ مراحل باید در یک مورد کاربرد در جریان رویدادها فهرست شوند که به زبان انسان توضیح داده شده است.
- ❖ هر مورد کاربرد یک جریان اصلی دارد (سناریو اولیه)، که مراحل یک مورد کاربرد را فهرست می کند که موقعیتی را نشان می دهد که در آن همه چیز طبق انتظار و دلخواه پیش می رود.
- ❖ جریان های جایگزین (سناریوهای ثانویه) انحراف از جریان اصلی هستند و می توانند خطاها، انشعابات و وقفه ها را در جریان اصلی ثبت کنند.
- ❖ جریان اصلی همیشه با انجام کاری برای شروع استفاده از بازیگر اصلی شروع می شود. زمان می تواند یک بازیگر باشد، بنابراین مورد استفاده نیز ممکن است با بیان زمانی به جای بازیگر شروع شود.

# یافتن جریان های جایگزین

با بازرسی جریان اصلی، جریان های جایگزین را شناسایی کنید. در هر مرحله از جریان اصلی، به دنبال موارد زیر باشید:

- ❖ جایگزین های ممکن برای جریان اصلی.
- ❖ خطاهایی که ممکن است در جریان اصلی ایجاد شوند.
- ❖ وقفه هایی که ممکن است در نقطه خاصی از جریان اصلی رخ دهد.
- ❖ وقفه که ممکن است در هر نقطه از جریان اصلی رخ دهد

# جزئیات یک Use Case

|                                                 |                                                                                                                                                                                                                                                               |
|-------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| use case name                                   | Use case: PaySalesTax                                                                                                                                                                                                                                         |
| use case identifier                             | ID: 1                                                                                                                                                                                                                                                         |
| brief description                               | Brief description:<br>Pay Sales Tax to the Tax Authority at the end of the business quarter.                                                                                                                                                                  |
| the actors involved in the use case             | Primary actors:<br>Time                                                                                                                                                                                                                                       |
|                                                 | Secondary actors:<br>TaxAuthority                                                                                                                                                                                                                             |
| the system state before the use case can begin  | Preconditions:<br>1. It is the end of the business quarter.                                                                                                                                                                                                   |
| the actual steps of the use case                | Main flow: <i>implicit time actor</i><br>1. The use case starts when it is the end of the business quarter.<br>2. The system determines the amount of Sales Tax owed to the Tax Authority.<br>3. The system sends an electronic payment to the Tax Authority. |
| the system state when the use case has finished | Postconditions:<br>1. The Tax Authority receives the correct amount of Sales Tax.                                                                                                                                                                             |
| alternative flows                               | Alternative flows:<br>None.                                                                                                                                                                                                                                   |

## پروژه : بخش دوم

برای پروژه خود موارد زیر را انجام دهید:

❖ نمودار موارد کاربرد را تولید کنید

❖ برای هر مورد کاربر جریان های اصلی و فرعی را بنویسید

❖ فایل خروجی گزارش را بسازید

❖ به ایمیل [adftalent@gmail.com](mailto:adftalent@gmail.com) ارسال کنید فرمت فایل نام و درس و شماره تمرین

❖ Aliahmadi\_oop\_hw2.zip

# انطباق نیازمندا با Use Case ها

با توجه به اینکه می توانیم نیازمندی های عملکردی را در یک مدل نیازمندی ها و یک مدل موردی استفاده کنیم، به روشی برای ارتباط این دو نیاز داریم.

❖ بین الزامات و موارد استفاده رابطه چند-به-چند وجود دارد:

- یک مورد استفاده بسیاری از نیازهای عملکردی فردی را پوشش می دهد
- یک نیاز کاربردی ممکن است با استفاده از موارد متعدد محقق شود

❖ می توانیم یک ماتریس Requirements Traceability ایجاد کنیم

| Requirements | Use cases |    |    |    |    |
|--------------|-----------|----|----|----|----|
|              |           | U1 | U2 | U3 | U4 |
|              | R1        |    |    |    |    |
|              | R2        |    |    |    |    |
|              | R3        |    |    |    |    |
|              | R4        |    |    |    |    |
|              | R5        |    |    |    |    |

