

مرحله طراحی سیستم

تحلیل و طراحی سیستمها

دکتر قاسمی

دانشگاه آزاد تهران شمال

- ❖ بعد از تحلیل نیازمندی ها نوبت به طراحی سیستم می رسد
- ❖ منظور از طراحی ، تبدیل نیازمندیها و مستند های تحلیل به نمودار ها است
- ❖ تبدیل سناریو ها به نمودار Use-Case Diagram
- ❖ هر سناریو متنی از نیازمندی ها به یک یا چند مورد کاربرد تبدیل می شود

تکنیک‌هایی برای شناسایی موارد کاربرد Use Case

سه روش برای شناسایی موارد کاربرد از روی مستندات تحلیل سیستم وجود دارد

❖ تکنیک هدف کاربر User Goal و شناسایی elementary business process EBP

❖ تکنیک تجزیه و تحلیل CRUD (ایجاد، خواندن، به روز رسانی، حذف). create, read, update, delete

❖ تکنیک تجزیه رویداد Event decomposition

تکنیک هدف کاربر User Goal

❖ تکنیک هدف کاربر User Goal

- **تعریف EBP**: هر هدف در سطح فرآیند یک elementary business process یا کسب و کار ابتدایی یا EBP است
- EBP وظیفه ای که توسط یک کاربر، در یک مکان در پاسخ به یک رویداد تجاری انجام می شود نامیده می شود، که ارزش تجاری قابل اندازه گیری را اضافه می کند و سیستم و داده ها را در حالت ثابت باقی می گذارد.
- تکنیک: هر EBP معادل یک مورد کاربرد است

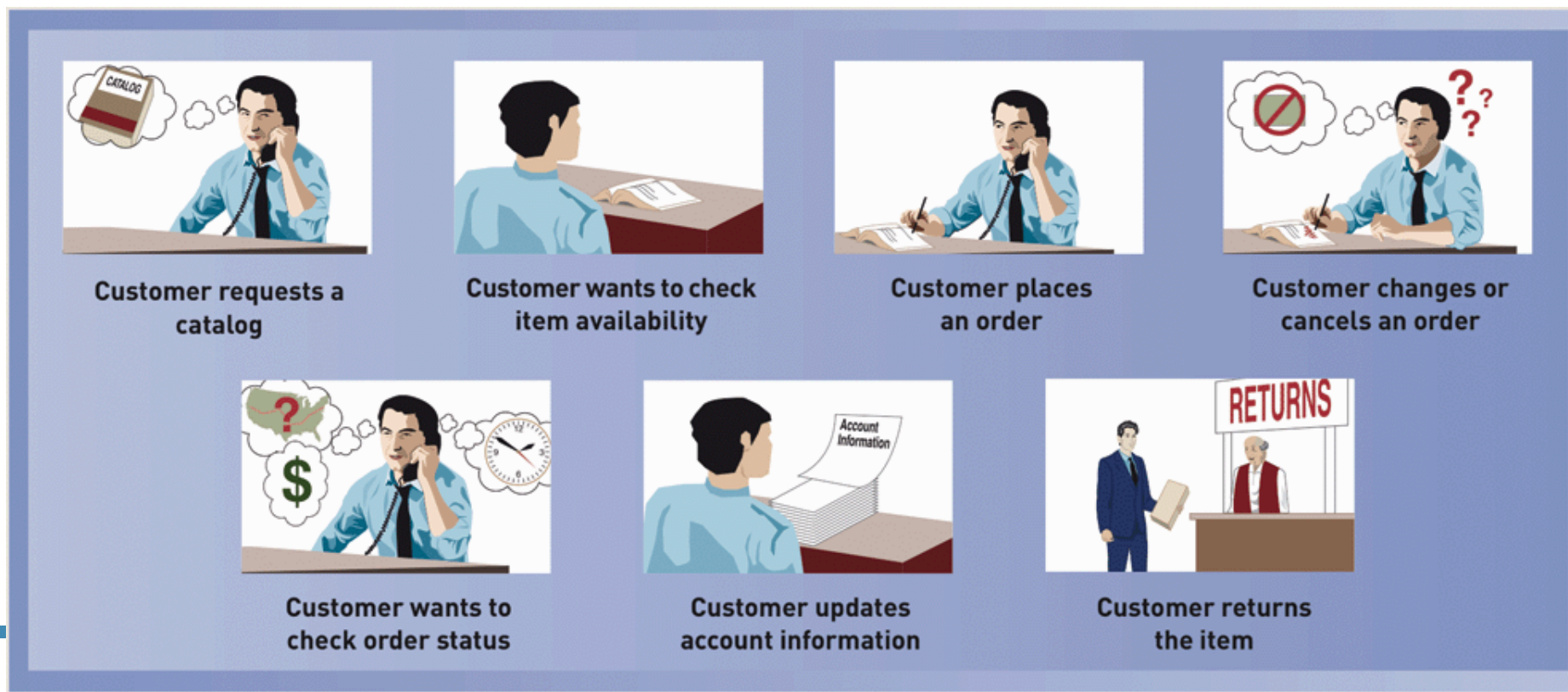
کاربر	هدف / علاقه / دوست دارد که
کاربر باجه رستوران	• ثبت سفارش جدید • ویرایش سفارش موجود
پیک	• دریافت لیست سفارشات بیرون بر • ارسال یک سفارش • اطلاع از سفارش های ارسال شده امروز

❖ در روش CRUD از موجودیت اصلی سیستم سوال می پرسیم که درباره CRUD آیا نیاز به سیستم دارد یا نه؟

موجودیت	CRUD	آیا در سیستم پیاده شود؟
اطلاعات غذاهای رستوران	CREATE	• بله: غذای جدید
	READ	• بله: لیست منو
	UPDATE	• بله: اعمال قیمت جدید
	DELETE	• خیر: به علت وجود داده های مرتبط سفارشات قبلی امکان حذف نداریم

روش تحلیل رخداد Event

- ❖ رویداد - اتفاقی است که در زمان و مکان خاص رخ می دهد و باید به خاطر سپرده شود
- ❖ رویدادهای تجاری فرآیندهای کسب و کار ابتدایی EBP رافراخوانی یا راه اندازی می کنند
- ❖ رویدادهای تجاری را شناسایی کنید تا سیستم را به موارد کاربرد تجزیه کنید



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

انطباق نیازمندا با Use Case ها

با توجه به اینکه می‌توانیم نیازمندی‌های عملکردی را در یک مدل نیازمندی‌ها 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 می‌دهد
- یک نیاز کاربردی ممکن است با استفاده از موارد متعدد محقق شود

❖ می‌توانیم یک ماتریس Requirements Traceability ایجاد کنیم

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

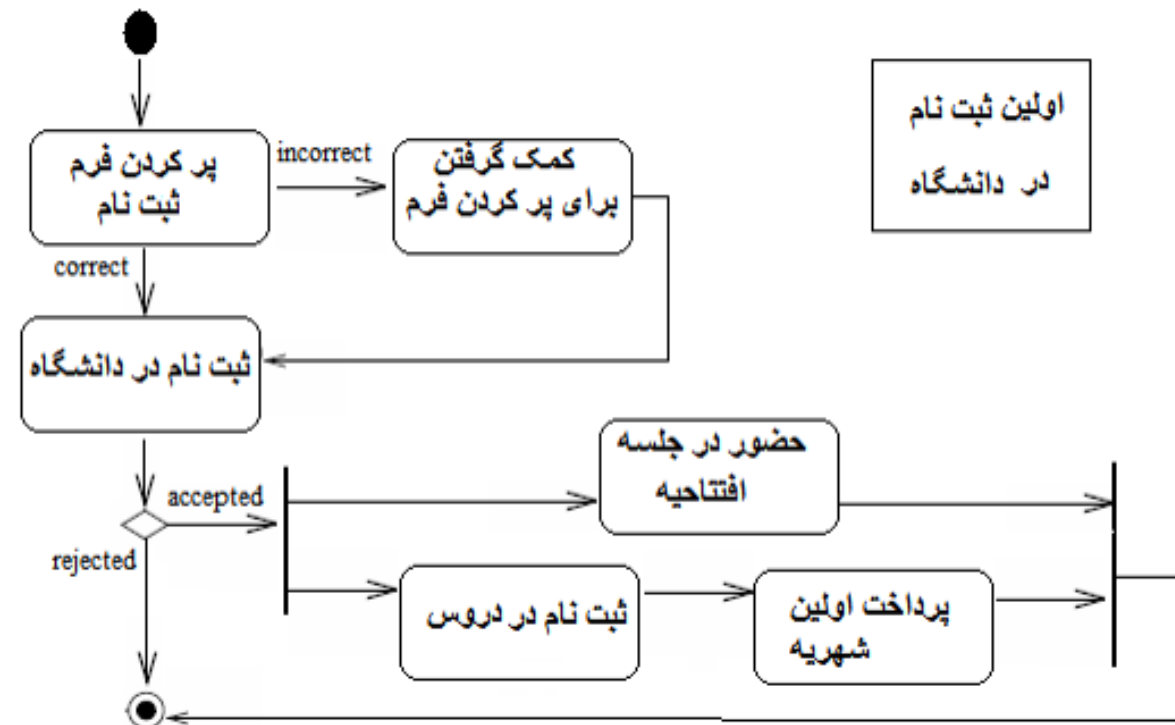
Activity Diagram

- ❖ نمودار فعالیت Activity diagram نمایش گرافیکی از گردش کار در فعالیت‌ها و اقدامات در یک سناریوی مشخص می‌باشد.
- ❖ نمودارهای فعالیت در نظر گرفته شده‌است تا هر دو فرایند محاسباتی و سازمانی (به عنوان مثال، گردش کار) و همچنین فرایندهای داده شده متقابل با فعالیت‌های مرتبط، مدل شوند
- ❖ نمودارهای فعالیت عمدتاً جریان کلی کنترل را نشان می‌دهند، آنها همچنین می‌توانند جریان داده‌ها را بین فعالیت‌ها از طریق یک یا چند داده ذخیره شده نشان دهند

Activity Diagram

❖ نمودارهای فعالیت از تعداد محدودی از اشکال ساخته شده که با فلش به هم متصل می‌شوند. مهم‌ترین اجزای نمودار فعالیت:

- بیضی نشان دهنده دستورات؛
- لوزی نشان دهنده تصمیم‌گیری؛
- میله‌ها (خطوط پررنگ افقی یا عمودی) نشان دهنده شروع (تقسیم) یا پایان (ادغام) فعالیت‌های همزمان؛
- یک دایره سیاه نشان دهنده شروع (گره اولیه) گردش کار؛
- یک دایره سیاه داخل دایره دیگر نشان دهنده پایان (گره نهایی).
- جهت فلش‌ها از شروع به سمت پایان رسم شده و نشان دهنده ترتیب اجرای فعالیت‌ها هستند.



❖ نمودار کلاس اصلی ترین جزء سازنده در مدلسازی شی گراست. این نمودار هم برای مدلسازی مفهومی کلی برنامه و هم برای مدلسازی طراحی جزئیات برای ترجمه به کد برنامه نویسی به کار می رود. یک کلاس در این نمودار به صورت جعبه مستطیلی نمایش داده می شود

❖ یک کلاس نمایشگر یک موجودیت از یک سیستم معین است که پیاده سازی کپسوله شده **encapsulated** ای از یک کارکرد موجودیت داده شده را ارائه می دهد.

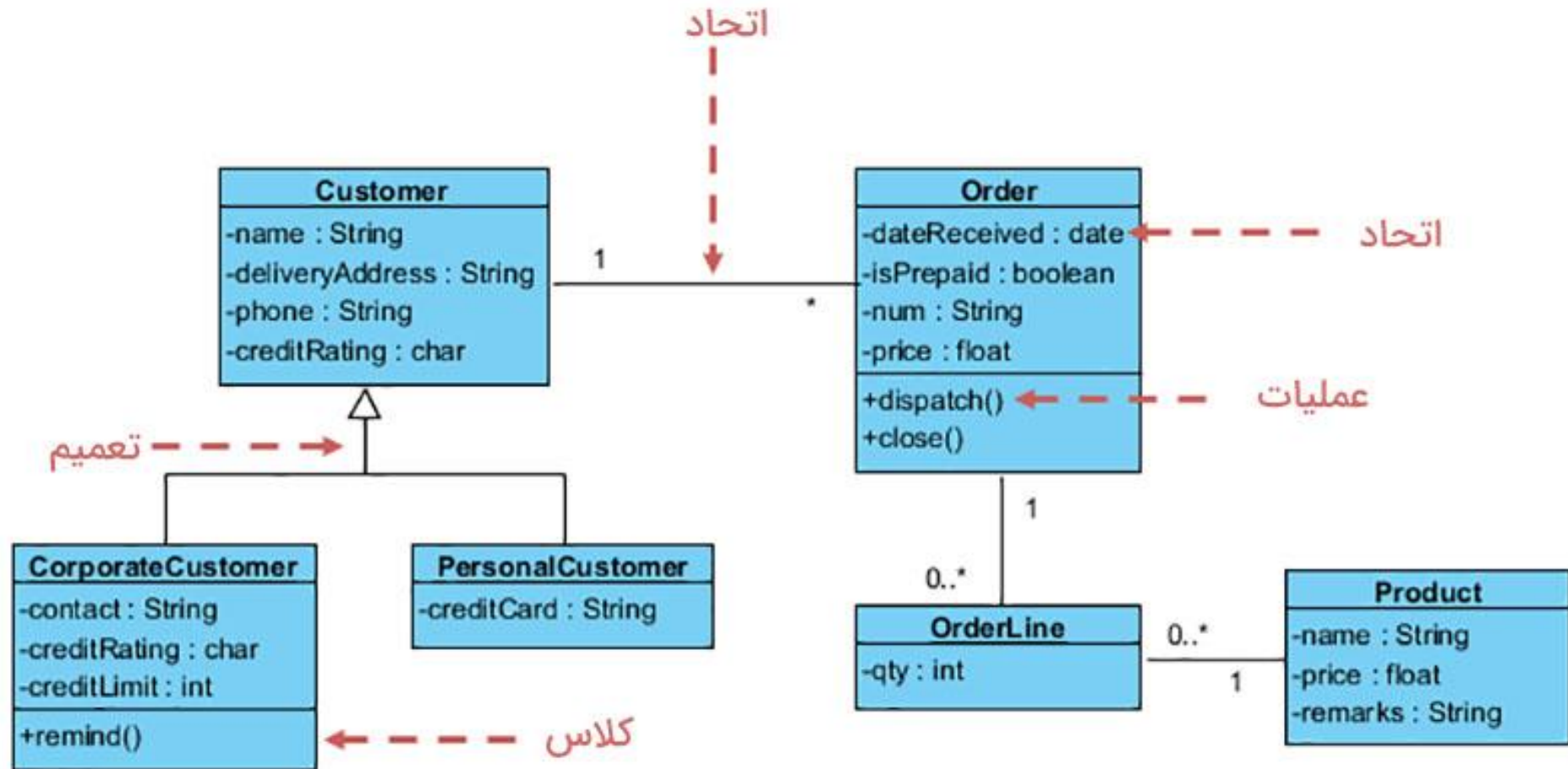
- این کارکردها از طریق متدهای یک کلاس عرضه می شوند.
- کلاس همچنین دارای ویژگیهایی است که خصایص یکتای کلاس را تعریف می کنند.

BankAccount

owner : String
balance : Dollars = 0

deposit (amount : Dollars)
withdrawal (amount : Dollars)

نمودار کلاس



یافتن کلاسها

❖ انجام تحلیل اسم/فعل روی اسناد:

- اسم ها احتمالا همان کلاس های کاندید هستند
- افعال احتمالا همان مسئولیت ها و رفتار کلاسهای کاندید هستند

❖ مراقب کلاس های جعلی باشید:

- به دنبال مترادف ها باشید - کلمات مختلف که معنی یکسان دارند
- به دنبال کلمات متجانس باشید - همان کلمه به معنای چیزهای مختلف

❖ به دنبال کلاس های "پنهان" باشید!

- کلاس هایی که به صورت اسم یا کارت ظاهر نمی شوند

❖ از همه مستندات فازهای قبلی استفاده کنید : مستند نیازمندیها، موارد کاربرد و هر مستند دیگری

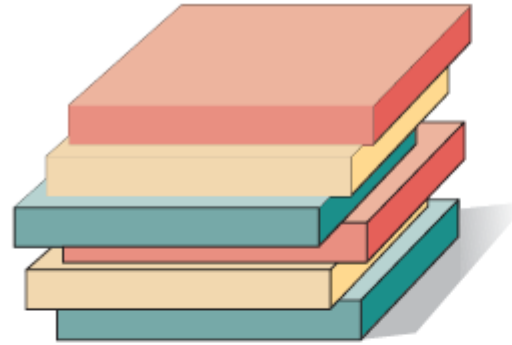
مرحله دوم از پروژه کلاسی

- ❖ برای پروژه خود یک نمودار Use Case Diagram ایجاد کنید
- ❖ برای پروژه خود یک نمودار Activity Diagram ایجاد کنید
- ❖ سه فایل را داخل یک فایل زیپ قرارداده و به ایمیل adftalent@gmail.com ارسال کنید
اسم فایل گویای فرد (نام ؛ شماره دانشجویی)

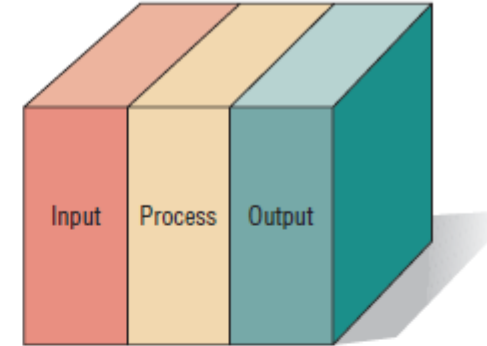
❖ تعریف: یکی از روش های طراحی قبل از شروع فاز تولید ، ایجاد نمونه کوچک از سیستم یا Prototype است که از طریق آن واکنش کاربران به نمونه اولیه را به تحلیلگر سیستمی نشان می دهد. واکنش ها از طریق مشاهده، مصاحبه و برگه های بازخورد (احتمالاً پرسشنامه) جمع آوری می شوند که برای جلب نظر هر فرد در مورد نمونه اولیه هنگام تعامل با آن طراحی شده اند.

❖ انواع نمونه های اولیه

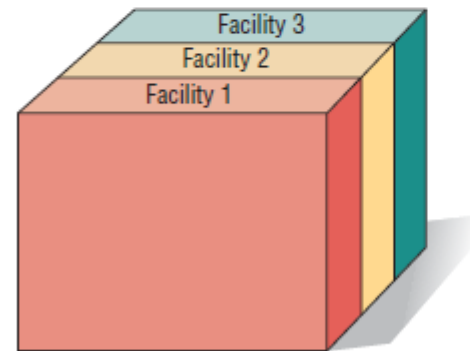
- نمونه اولیه وصله شده. اولین نوع نمونه سازی مربوط به ساختن سیستمی است که کار می کند اما از وصله کردن زیرسیستمهای دیگران ایجاد شده شده است.
- نمونه ای اولیه ناکارآمد : در سیستم های اطلاعاتی، مدل کاری است که تمام ویژگی های لازم را دارد اما ناکارآمد است. در این نمونه از نمونه سازی، کاربران می توانند با سیستم تعامل داشته باشند و به رابط و انواع خروجی های موجود عادت کنند. با این حال، بازیابی و ذخیره سازی اطلاعات ممکن است ناکارآمد باشد، زیرا برنامه ها به سرعت با هدف قابل اجرا بودن به جای کارآمد نوشته شده اند.
- نمونه اولیه غیر عملیاتی ، مدلی در مقیاس غیر کاربردی است که برای آزمایش جنبه های خاصی از طراحی تنظیم شده است. نمونه ای از این رویکرد یک مدل کامل از یک خودرو است که در آزمایشات تونل باد استفاده می شود. اندازه و شکل خودرو دقیق است، اما ماشین عملیاتی نیست.
- نمونه اولیه ویژگی های انتخاب شده. شامل برخی، اما نه همه، ویژگی هایی است که سیستم نهایی خواهد داشت. یک قیاس می تواند یک مرکز خرید خرده فروشی جدید باشد که قبل از تکمیل ساخت همه مغازه ها افتتاح می شود.



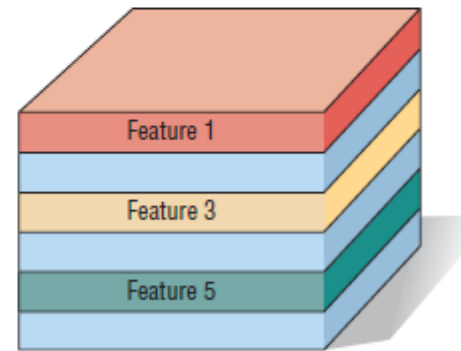
Patched-Up Prototype



Nonoperational Prototype



First-of-a-Series Prototype



Selected Features Prototype

