

## فصل ۱۸: مدلسازی تحلیل شیء گرا

### هدف از مدلسازی تحلیل شیء گرا چیست؟

وقتی قرار است محصول یا سیستم جدیدی ایجاد شود، چگونه آن را به نحوی مشخص کنیم که بتواند به صورت شیء گرا مهندسی شود؟ آیا پرسش‌هایی وجود دارند که باید از مشتریان پرسیده شوند؟ اشیاء چگونه با یکدیگر ارتباط دارند؟ اشیاء در حیطه سیستم چگونه رفتار می‌کنند؟ چگونه مساله‌ای را مشخص یا مدلسازی کنیم تا بتوانیم یک طراحی کارآمد ایجاد کنیم؟

هر یک از این پرسش‌ها در حیطه تحلیل شیء گرا (Object Oriented Analysis- OOA) - نخستین فعالیت تکنیکی که در مهندسی نرم افزار شیء گرا (Object Oriented-OO) اجرا می‌شود - پاسخ داده می‌شود.

OOA به جای بررسی یک مساله با استفاده از مدل جریان اطلاعات کلاسیک، چند مفهوم جدید را معرفی می‌کند. OOA ریشه در مجموعه‌ای از اصول بنیادی دارد که در فصل ۱۱ معرفی شد. برای ساخت یک مدل تحلیل، پنج اصل بنیادی به کار برده می‌شود:

۱. دامنه اطلاعاتی مدلسازی می‌شود؛
۲. عملکرد توصیف می‌شود؛
۳. رفتار نمایش داده می‌شود؛
۴. مدل‌های داده‌ای، عملیاتی و رفتاری افزاری می‌شوند تا جزییات بیشتری در معرض دید قرار گیرند، و
۵. مدل‌های اولیه، بنیاد و ماهیت مساله را نشان می‌دهند، حال آنکه مدل‌های نهایی، جزییات ساده‌ای را نمایش می‌دهند.

این اصول، مبنای روش OOA را تشکیل می‌دهند.

هدف OOA تعریف کلیه کلاس‌هایی است که به نوعی با مساله ارتباط دارند - عملیات و صفات مرتبط با آنها و روابط میان آنها و رفتاری که از خود نشان می‌دهند. برای این منظور، چند کار باید صورت گیرد:

- خواسته‌های اصلی کاربر باید بین مشتری و مهندس تبادل شود.
- کلاس‌ها باید شناسایی شوند.
- سلسله مراتب کلاس‌ها باید مشخص شود.
- روابط شیء با شیء باید نشان داده شود.
- رفتار اشیاء باید مدلسازی شود.

### شهرت OOA

محبوبیت فناوری‌های شیء گرا منجر به ابداع دهها روش OOA در اواخر دهه ۱۹۸۰ و اوایل دهه ۹۰ شد. هر یک از این روش‌ها معرف فرآیندی برای تحلیل یک محصول یا سیستم، یک مجموعه نمودار که از فرآیند به دست می‌آیند، و نشانه‌گذاری‌هایی است که مهندس نرم افزار را قادر به ایجاد مدل تحلیل می‌کند. روش‌های زیر طی دهه اخیر کاربرد بیشتری دارند.

**روش بوچ (Booch) [BOO94]:** این روش شامل یک فرآیند توسعه میکرو و یک فرآیند توسعه ماکرو است. سطح میکرو، مجموعه‌ای از وظایف تحلیل را تعریف می‌کند که برای هر مرحله از فرآیند ماکرو دوباره اجرا می‌شود. از این رو، یک روش تکاملی صورت می‌گیرد. در فرآیند توسعه میکرو بوچ برای OOA، کلاس‌ها، اشیاء و معانی آنها تعیین می‌شود؛ روابط میان کلاس‌ها و اشیاء تعیین می‌گردد و پالایش‌های لازم صورت می‌پذیرد تا مدل تحلیل شناسایی شود.

**روش رومبو و همکاران (Rumbaugh) [RUM91]:** رومبو و همکاران وی تکنیک مدل سازی شیء گرا (OMT) را برای تحلیل، طراحی سیستم و طراحی در سطح اشیاء توسعه دادند. نتیجه فعالیت تحلیل، سه مدل است: مدل اشیاء (نمایشی از اشیاء، کلاس ها، سلسله مراتب و روابط)، مدل پویا (نمایشی از رفتار شیء و سیستم) و مدل عملیاتی (نمایشی سطح بالا و به شکل DFD از جریان اطلاعات در سیستم).

**روش جاکوبسون (Jacobson):** این روش که آن را OOSE (مهندس نرم افزار شیء گرا) نیز می نامند، نسخه ساده ای از یک روش شیء گرای قدیمی تر است که آن را نیز جاکوبسون ابداع کرد. تفاوت این روش با روش های دیگر در تأکید فراوان آن بر موارد کاربرد (Use Case) است- یعنی شرح یا سناریویی که چگونگی تعامل کاربر با محصول یا سیستم را تصویر می کند.

**روش کود و یوردون (Coad & Yourdon) [COA91]:** این روش غالباً به عنوان یک از آسانترین روش های OOA در نظر گرفته می شود. نشانه گذاری مدلسازی نسبتاً ساده است و دستورالعمل های توسعه مدل تحلیل، صریح هستند. فرآیند OOA کود و یوردون را به اختصار در زیر مطرح می کنیم:

- ◆ شناسایی اشیاء با استفاده از ملاک های ((جستجو))
- ◆ تعریف یک ساختار تعمیم- تعیین مشخصات
- ◆ تعریف ساختار کامل مؤلفه
- ◆ شناسایی موضوع ها (نمایشی از مؤلفه های زیرسیستم)
- ◆ تعریف صفات
- ◆ تعریف سرویس ها

**روش ویرف- بروک (Wriifs & Brok) [WIR90]:** ویرف- بروک بین وظایف طراحی و تحلیل، تفاوت چندانی قایل نمی شوند. در عوض، فرآیندی پیوسته را پیشنهاد می کنند که با ارزیابی مشخصات مشتری آغاز می شود و با طراحی پایان می یابد. وظایف مرتبط با تحلیل در این روش به اختصار به شرح زیر می باشد:

- ◆ ارزیابی مشخصات مشتری
- ◆ استخراج کلاس های نامزد از روی مشخصات، از طریق تجزیه گرامری
- ◆ گروه بندی کلاس ها به نیت شناسایی کلاس های پایه
- ◆ تعریف مسوولیت ها برای هر کلاس
- ◆ نسبت دادن مسوولیت ها به هر کلاس
- ◆ تعیین روابط میان کلاس ها
- ◆ تعیین مشارکت میان کلاس ها براساس مسوولیت
- ◆ ساخت نمایش های سلسله مراتبی از کلاس ها
- ◆ تولید گراف مشارکت برای سیستم

گرچه مراحل، اصطلاح شناسی و فرآیندهای هر یک از این روش های OO متفاوت است، ولی فرآیندهای کلی OOA بسیار مشابهند. مهندس نرم افزار برای اجرای تحلیل شیء گرا باید مراحل کلی زیر را دنبال کند:

۱. روشن کردن خواسته های مشتری برای سیستم
۲. شناسایی سناریو یا موارد کاربرد
۳. انتخاب کلاس ها و اشیاء با استفاده از نیازها (خواسته ها) به عنوان یک راهنما
۴. شناسایی صفات و عملیات مربوط به هر یک از اشیای سیستمی
۵. تعریف ساختارها و سلسله مراتبی که کلاس ها را سازماندهی می کنند.

۶. ساخت یک مدل شیء-واسط
  ۷. ساخت یک مدل شیء-رفتار
  ۸. بازیابی مدل تحلیل OO نسبت به موارد کاربرد/ سناریوها
- این مراحل کلی را با جزییات بیشتری در ادامه این فصل ارایه شده است.

چرا از مدل شیء گرا استفاده می کنیم؟

۱. روش یکنواختی برای تحلیل شیء گرا.
۲. استفاده از تحلیل دامنه.
۳. استفاده مجدد.

منابع آگاهی از دامنه

۱. ادبیات فنی
۲. نرم افزارهای کاربردهای موجود
۳. تحقیق از مشتری
۴. وسایل هوشمند و خبره
۵. شناسایی نیازمندی های فعلی و آینده

مدل تحلیل دامنه

۱. شناسایی انواع کلاس ها
۲. داشتن استانداردهای استفاده مجدد
۳. استخراج مدل های عملیاتی
۴. دامنه های مورد نظر

مراحل فرآیند تحلیل دامنه

۱. تعریف و تعیین دامنه ای که باید مورد بررسی قرار گیرد.
  ۲. گروه بندی عناصر استخراج شده از دامنه.
  ۳. جمع آوری نمونه ای از کاربردهای موجود در دامنه که نماینده کل دامنه باشد.
  ۴. تحلیل هر کاربرد در نمونه.
- شناسایی اشیای کاندیدا برای استفاده مجدد.
  - ذکر دلایل برای انتخاب شیء جهت استفاده مجدد.
  - تعریف تطابق هایی برای شیء که ممکن است بعدا استفاده شود.
  - برآورد درصدی از کاربردهای موجود در دامنه که ممکن است از شیء استفاده کنند.
  - شناسایی اشیاء از طریق نام و استفاده از تکنیک های مدیریت پیکربندی برای مدیریت آنها.

مولفه های عمومی در مدل تحلیل شیء گرا

۱. نمای ایستا از کلاس های معنایی
۲. نمای ایستا از صفات
۳. نمای ایستا از روابط

۴. نمای ایستا از رفتارها
۵. نمای پویا از رفتارها
۶. نمای پویا از ارتباطات
۷. نمای پویا از کنترل و زمان

## مدلسازی مسوولیت و مشارکت کلاس ها

### کلاسها

خصوصیات کلاسهای گزینشیء

۱. اطلاعات نگهداری شده
۲. اطلاعات مورد نیاز
۳. صفات چندگانه
۴. صفات متداول
۵. عملیات متداول
۶. خواسته های اساسی

### انواع کلاس ها

۱. کلاس های دستگاهی
۲. کلاس های خواص
۳. کلاس های تعامل
۴. عینی بودن
۵. شمول
۶. دوام
۷. آسیب پذیری

### مسئولیت ها

۱. هوشمندی سیستم باید از توزیعی مناسب برخوردار باشد.
۲. هر مسوولیتی را باید هر چه کلی تر بیان کرد.
۳. اطلاعات و رفتاری که به آن مربوط می شود باید در یک کلاس قرار گی-رد.
۴. اطلاعات مربوط به یک چیز باید در یک کلاس متمرکز شود.
۵. مسوولیت ها باید در صورت امکان بین کلاس ها پخش شود.

### مشارکت کننده ها

- کلاس ... از کلاس ... آگاه است.
- کلاس ... بخشیء از کلاس ... است.
- کلاس ... به کلاس ... وابسته است.

## تعریف ساختارها و سلسله مراتب

### تعریف زیر سیستم

- مدل روابط میان اشیاء
- مدل رفتار اشیاء
- شناسایی رویدادها با موارد کاربرد
- نمایش حالت ها

## مروری بر زبان مدلسازی یکنواخت (UML-Unified Modelling Language)

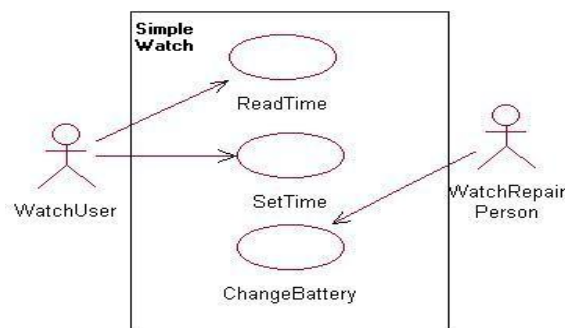
زبان یکنواخت و یکپارچه مدلسازی (UML) زبان مدلسازی گرافیکی است که با ترکیبی از نمادها برای تشریح عناصر اصلی سیستم های نرم افزاری (که در UML به آن محصول گفته می شود) تهیه گردیده است. UML شامل سه مدل به شرح زیر است:

۱. مدل تابعی (Functional Model): عملکرد سیستم را از دید کاربر شرح می دهد. در UML برای این مدلسازی از نمودارهای Use-Case استفاده می شود.
۲. مدل شیء (Object Model): ساختار سیستم را به صورت مجموعه ای از اشیاء، صفات، عملیات و ارتباطات شرح می دهد. در UML برای این مدلسازی از نمودار کلاس (Class Diagram) استفاده می شود.
۳. مدل پویا (Dynamic Model): رفتار داخلی سیستم را شرح می دهد. در UML برای این مدلسازی از نمودارهای Sequence, Statechart, Activity استفاده می شود.

## نمودار مورد کاربرد (Use Case Diagram)

نمودار Use Case در طول استخراج نیازها و تحلیل سیستم برای مشخص کردن عملکرد برنامه به کار می رود. Use Case ها روی رفتار سیستم از یک دیدگاه خارج از سیستم تمرکز می کنند. یک Use Case در واقع عملیاتی را شرح می دهد که توسط سیستم تهیه شده و نتیجه ای مشخص برای یک بازیگر (Actor) دارد.

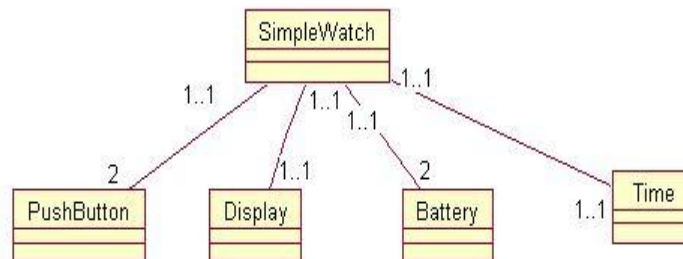
بازیگر یک موجودیت است که با سیستم در حال تعامل می باشد. (کاربر، یک سیستم دیگر و ...) نمونه ای از نمودار مورد کاربرد در شکل ۱ نشان داده شده است. در این مورد کاربرد Watch User و WatchRepair Person دو بازیگر هستند که با سیستم در تعامل می باشند.



شکل ۱: نمودار یک مورد کاربرد

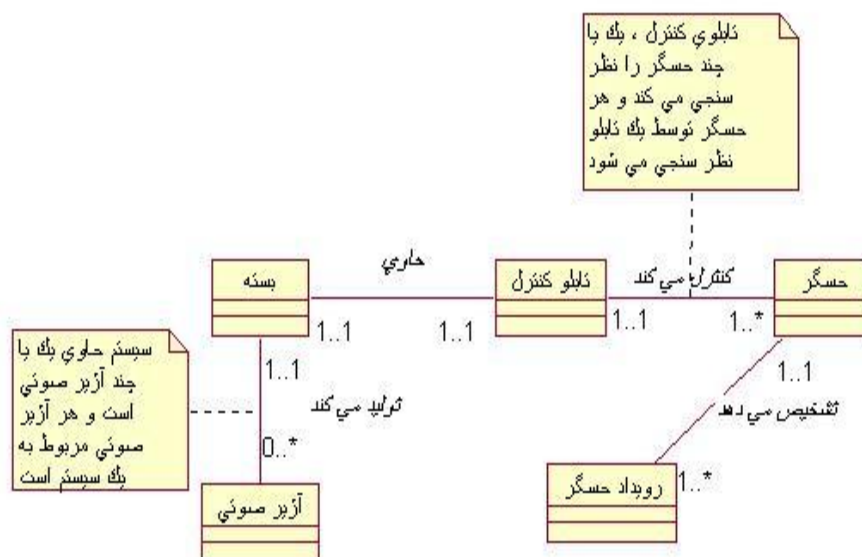
## نمودار کلاس (Class Diagram)

از نمودار کلاس برای نشان دادن ساختار سیستم استفاده می شود. کلاس ها انتزاعی از ساختارهای مشترک و رفتار مجموعه ای از اشیا می باشند. اشیا نمونه هایی از کلاس ها می باشند که در حین اجرای سیستم ساخته می شوند، تغییر می کنند و غالباً بعد از پایان اجرای سیستم از بین می روند. شکل ۲ نیز نمونه ای از نمودار کلاس را برای مورد کاربرد نشان داده شده در شکل ۱ نشان می دهد.



شکل ۲: نمودار سلسله مراتبی کلاس ها

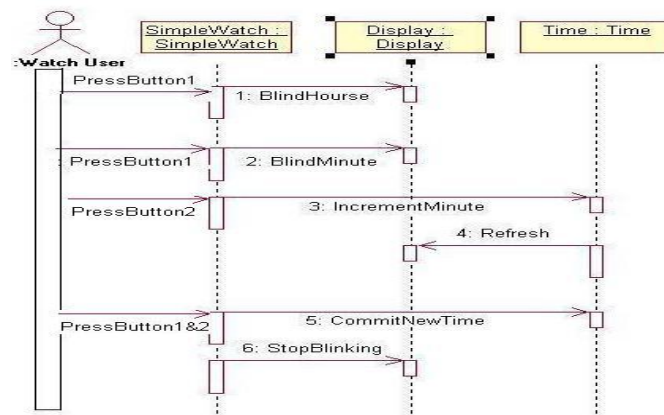
مجموعه ای از کلاس های مرتبط با ارتباطات نامگذاری شده و چندی های مشخص



شکل ۳: نمودار کلاس ها به همراه چندی بین آنها

## نمودار توالی (Sequence Diagram)

نمودار توالی برای فرموله کردن رفتار سیستم و همچنین واضح کردن ارتباطات بین اشیا به کار می رود. نمودار توالی برای نشان دادن اشیایی که در یک Use Case سهیم هستند مفید است. به همین دلیل به اشیایی که در یک نمودار توالی نشان داده می شوند اشیا شرکت کننده (Participating Object) می گویند. در شکل ۴ نمونه ای از نمودار توالی رسم شده است.

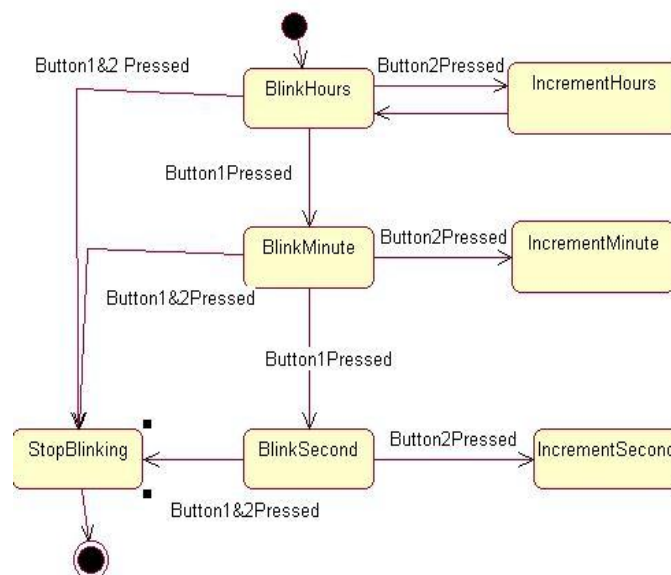


شکل ۴: نمودار کلاس ها به همراه چندی بین آنها

### نمودار حالت (StateChart Diagram)

نمودار حالت رفتار یک شیء مشخص را با تعدادی حالت (State) و حرکت بین این حالات شرح می دهد. یک شیء در یک حالت، مقادیر مشخصی برای ویژگی های خودش دارد. منظور از حرکت یعنی رفتن شیء از یک حالت به حالت دیگری با روی دادن یک رویداد خاص می باشد.

در نمودار حالت تمرکز روی پیغام هایی است که تحت تاثیر یک رویداد بوجود آمده توسط بازیگر بین اشیا مختلف مبادله می شود اما در نمودار حالت تمرکز بر انتقال بین حالت ها می باشد که این انتقال ناشی از روی دادن یک رویداد برای آن شیء خاص می باشد. شکل ۵ را ببینید.



شکل ۵: نمودار کلاس ها به همراه چندی بین آنها

اکنون در ادامه بحث فرایند تحلیل شیء گرای را با مثال سیستم خانه امن یکبار دیگر ادامه می دهیم.

## فرایند تحلیل شیء گرا (OOA Process)

فرایند تحلیل شیء گرا در آغاز کاری با خود اشیا ندارد و ابتدا با درک شیوه استفاده از سیستم توسط کاربران (انسان، ماشین و یا برنامه های دیگر) آغاز می شود .  
تکنیک هایی برای جمع آوری خواسته های مشتری و سپس تعریف یک مدل تحلیل برای یک سیستم شیء گرا به شرح زیر وجود دارد:

### Use-Case ها

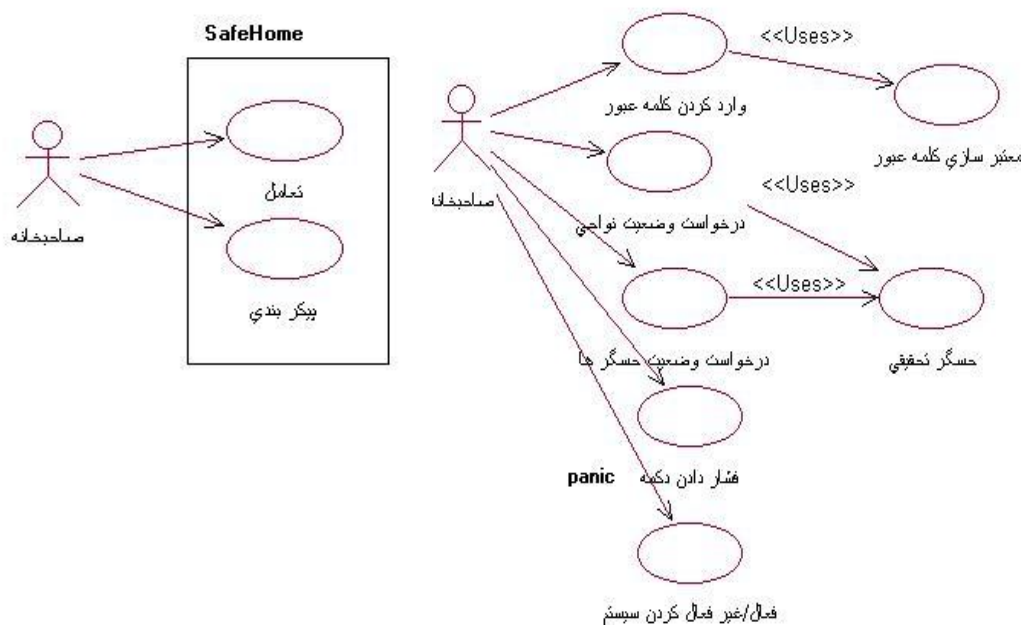
Modeling of Responsibility and Collaboration (Class-Responsibility-Collaborator Modelling)

Defining Structures and Hierarchies (Defining Structures and Hierarchies)

Defining Subjects and Subsystems (Defining Subjects and Subsystems)

## موارد کاربرد (Use-Case)

- مورد کاربرد یکی از نمودارهای UML می باشد که در واقع شرحی از مجموعه تعاملات بین کاربران و سیستم است. Use-Case سیستم را از دیدگاه کاربر نهایی مدلسازی می کند. Use-Case ها باید اهداف زیر را برآورده سازند:
- تعریف خواسته های عملیاتی سیستم با تعریف یک سناریوی کاربرد که مشتری و تیم مهندسی نرم افزار در مورد آن به توافق رسیده اند.
- فراهم آوردن توصیفی واضح و عاری از ابهام از چگونگی تعامل سیستم و کاربر نهایی با یکدیگر
- فراهم آوردن مبنایی برای انجام آزمون های اعتبار سنجی



شکل ۶: نمودار مورد کاربرد سیستم خانه امن

## مدلسازی مسوولیت و مشارکت کلاس ها

مدلسازی مسوولیت و مشارکت کلاس ها (CRC) وسیله ای ساده برای شناسایی و سازماندهی کلاس های مرتبط با خواسته های سیستم فراهم می آورد.



### مراحل کار:

- شناسایی کلاس‌ها
- شناسایی مسوولیت‌ها (صفات و عملیات)
- شناسایی مشارکت کنندگان

### شناسایی کلاس‌ها

شش خصوصیت برای شناسایی کلاس‌های بالقوه تعریف می‌شود:

- ۱- اطلاعات نگهداری شده
- ۲- اطلاعات مورد نیاز
- ۳- صفات چندگانه
- ۴- صفات مشترک
- ۵- عملیات مشترک
- ۶- خواسته‌های اساسی

### انواع کلاس‌ها

- ۱- کلاس‌های دستگاهی (Device Classes)
- ۲- کلاس‌های خواص (Property Classes)
- ۳- کلاس‌های تعامل (Interaction Classes)

### ویژگیهای یک کلاس

- ۱- عینی یا انتزاعی
- ۲- اتمی یا مجتمع
- ۳- ترتیبی یا غیر ترتیبی
- ۴- گذرا، موقت یا دائمی
- ۵- جامعیت کلاس

### شناسایی مسوولیت‌ها

پنج دستور العمل برای تخصیص مناسب مسوولیت‌ها به کلاس‌ها به شرح زیر می‌باشد:

- ۱- توزیع مناسب هوشمندی سیستم
- ۲- بیان کلی مسوولیت‌ها
- ۳- قرار گرفتن اطلاعات و رفتار مربوط به هر کلاس در همان کلاس
- ۴- متمرکز شدن اطلاعات مربوط به هم در یک کلاس
- ۵- به اشتراک گذاشتن مسوولیت‌ها بین کلاس‌های مرتبط

### شناسایی مشارکت کنندگان

برای شناسایی مشارکت کنندگان سه رابطه کلی میان کلاس‌ها بررسی می‌شود:

- ۱- رابطه بخشی از .... است.
- ۲- رابطه از ..... آگاه است.
- ۳- رابطه وابسته به ... است.

## کارت شاخص

کارت‌هایی که برای نشان دادن کلاس‌ها به کار می‌روند و دارای سه بخش هستند:

- ۱- نام کلاس در بالای کارت
  - ۲- لیستی از مسوولیت‌های کلاس در طرف چپ
  - ۳- لیستی از مشارکت‌کننده‌ها در طرف راست
- نمونه‌ای از یک کارت شاخص در شکل نشان داده شده است.

نام کلاس:	
نوع کلاس (مثلا دستگاہی، خواص، نقش، رویداد)	
خصوصیت کلاس (مثلا ملموس، اتمی، همزمان و ..)	
مسئولیتها:	مشارکت کننده ها:

شکل ۷: نمونه‌ای از یک کارت CRC

## بررسی مدل (CRC)

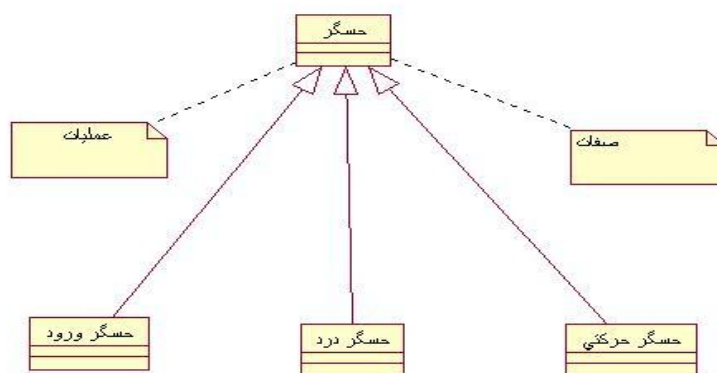
مراحل کار برای تولید مدل CRC به صورت زیر خواهد بود:

- ۱- به همه شرکت کنندگان در بررسی، زیر مجموعه‌ای از کارت‌های شاخص مدل CRC داده می‌شود.
- ۲- همه سناریوهای Use-Case باید گروه بندی شوند.
- ۳- شخص مسوول بررسی، Use-Case ها را به دقت می‌خواند و همین که به نام یک کلاس رسید نوبت را به کسی می‌دهد که کارت شاخص کلاس مربوطه در دست اوست.
- ۴- هنگام تحویل کارت‌ها از دارنده کارت خواسته می‌شود تا مسوولیت‌های ذکر شده در کارت را شرح دهد و گروه بررسی می‌کند که آیا تمام خواسته‌های Use-Case برآورده می‌شود یا نه؟
- ۵- اگر مسوولیت‌ها و مشارکت‌های ذکر شده روی کارت‌های شاخص نتوانند جوابگوی نیازهای موارد کاربردها (Use-Cases) باشند اصلاحات در کارت‌ها ضروری است و این کار انجام می‌گیرد.

## تعریف ساختارها و سلسله مراتب کلاس‌ها

### ۱- ساختار کلاس تعمیم تخصص (Is A)

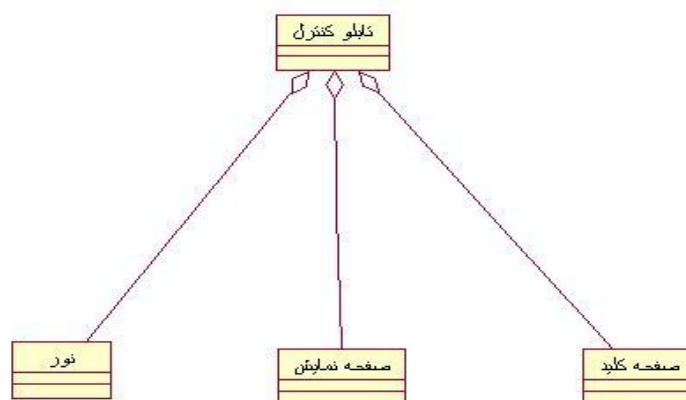
در این ساختار یک کلاس به عنوان زیر کلاس کلاس دیگر مطرح می‌شود و تمام صفات و عملیات‌های آن را به ارث می‌برد. مثال:



شکل ۸: نمودار کلاس‌ها برای تعمیم- تخصیص

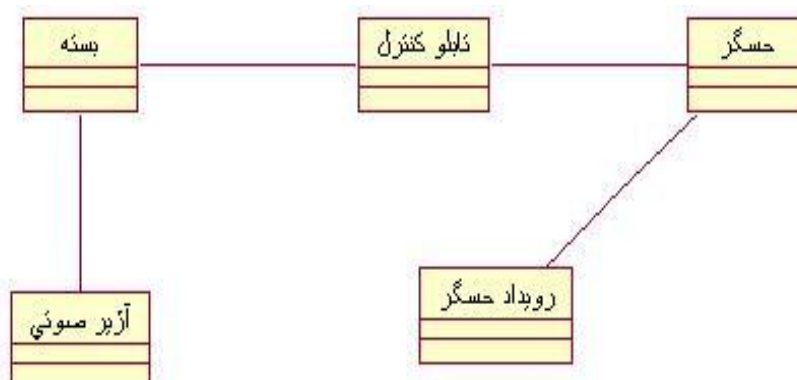
## ۲- ساختار کلاس مجتمع مرکب (Is A Part Of)

در این ساختار یک شیء از چند مولفه تشکیل شده است که این مولفه‌ها را می‌توان به عنوان یک شیء تعریف کرد. مثال:



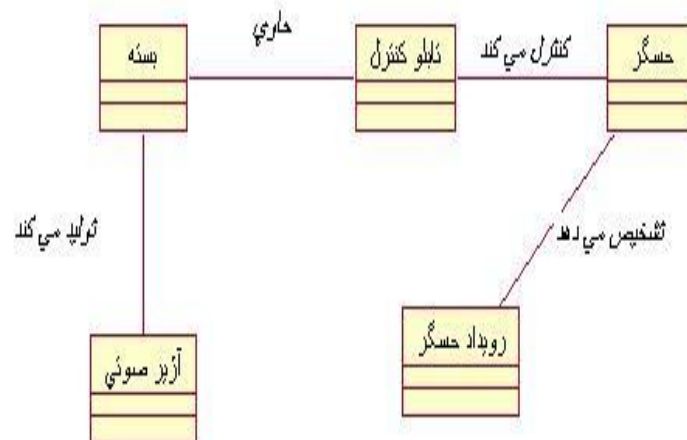
شکل ۹: نمودار کلاس‌ها برای کلاس‌های مجتمع مرکب

مجموعه ای از کلاس‌های مرتبط



شکل ۱۰: نمودار کلاس‌های مرتبط

مجموعه ای از کلاس های مرتبط با ارتباطات نامگذاری شده



شکل ۱۱: نمودار کلاس های مرتبط با تعریف ارتباطات

### تعریف زیر سیستم (SubSystem)

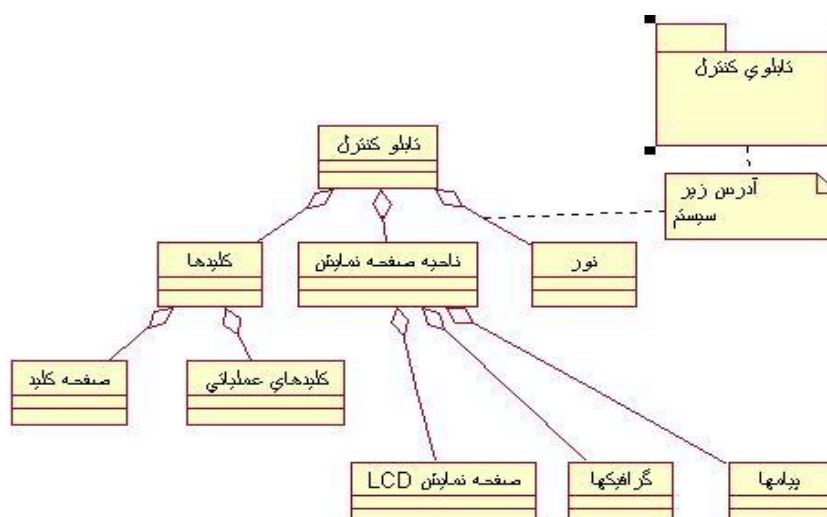
هرگاه گروهی از کلاس ها با یکدیگر مشارکت کنند تا مجموعه ای از مسوولیت های منسجم را بوجود آورند، می توان آنها را به عنوان یک زیر سیستم در نظر گرفت.

### کارت شاخص زیر سیستم

شامل نام زیر سیستم ، توافقنامه های مربوطه و کلاس ها و یا زیر سیستم های دیگری که باید توافق را پشتیبانی کنند.

### بسته (Package)

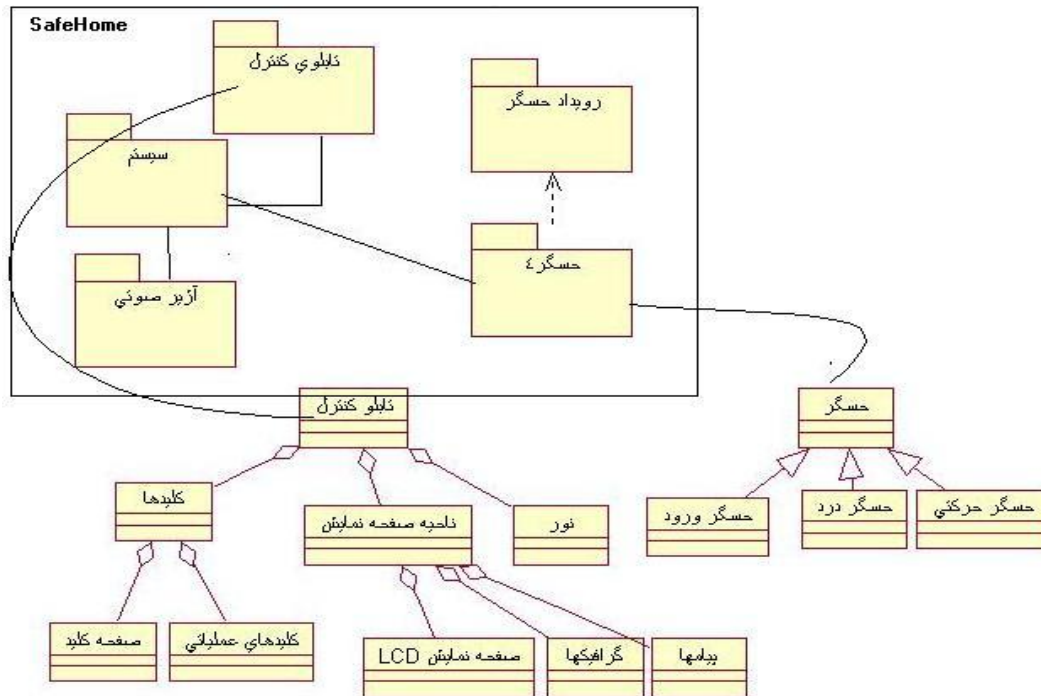
نمایش گرافیکی زیر سیستم در UML است.



شکل ۱۲: نمودار زیر سیستم و بسته در سیستم خانه امن

## مدل OOA در انتزاعی ترین سطح

مدل OOA در بالاترین سطح انتزاع فقط شامل بسته‌ها است. هریک از آدرس‌ها به یک ساختار بسط می‌یابد. شکل را ببینید.

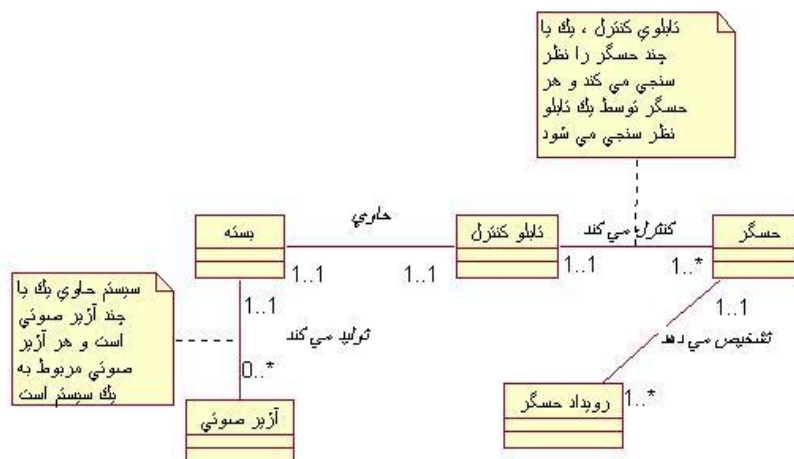


شکل ۱۲: نمودار زیر سیستم و بسته در سیستم خانه امن

## مدل روابط بین اشیاء

مدل روابط بین اشیاء در سه مرحله به دست می‌آید:

- ۱- با استفاده از کارت‌های شاخص CRC مجموعه‌ای از اشیاء مشارکت کننده رسم می‌شود.
- ۲- با بازبینی کارت‌های شاخص مدل CRC مسوولیت‌ها و مشارکت‌کننده‌ها مورد ارزیابی قرار گرفته و ارتباطات نامگذاری می‌شوند.
- ۳- هنگامی که روابط با نام مشخص شد نحوه مشارکت کلاس‌ها در ارتباط ارزیابی می‌شود. شکل زیر را ببینید.



شکل ۱۳: نمودار روابط بین اشیاء

## مدل رفتار اشیا ( Object – Behavior Model )

مدل رفتار اشیا نشان می دهد که سیستم شی گرا چگونه به رویداد ها و محرک های خارجی پاسخ خواهد داد.

### مراحل ایجاد مدل

- ۱- ارزیابی کلیه Use-Case ها برای درک کامل ترتیب تعامل ها در سیستم
- ۲- شناسایی رویدادهایی که ترتیب تعامل ها را هدایت می کنند و درک چگونگی ارتباط این رویداد ها با اشیا خاص
- ۳- ایجاد یک پیگرد رویداد برای هر Use-Case
- ۴- ساختن یک نمودار گذار حالت برای سیستم
- ۵- بازبینی مدل رفتار اشیا به منظور اعتبار سنجی، صحت و سازگاری

### مثال SafeHome

- ۱- صاحبخانه به تابلوی کنترل نگاه می کند تا تعیین کند که آیا سیستم آماده دریافت ورودی هست یا خیر .  
اگر سیستم آماده نباشد صاحبخانه باید از نظر فیزیکی درها یا پنجره ها را ببندد تا نشانگر آمادگی ارائه شود.
- ۲- صاحبخانه با استفاده از صفحه کلید ، کلمه عبور چهار رقمی را وارد می کند . این کلمه با کلمه عبور موجود در سیستم مقایسه می شود. اگر کلمه عبور نادرست باشد ، تابلوی کنترل بوق می زند و آماده ورودی بعدی خواهد شد . اگر کلمه عبور درست باشد ، منتظر فعالیتهای دیگر می ماند.
- ۳- صاحبخانه با استفاده از صفحه کلید ، Stay یا Away را وارد می کند تا سیستم فعال شود.
- Stay فقط حسگر های محیطی را فعال می کند . Away تمام حسگر ها را فعال می کند.
- ۴- پس از فعال سازی صاحبخانه می تواند تمام حسگرها را فعال کند.

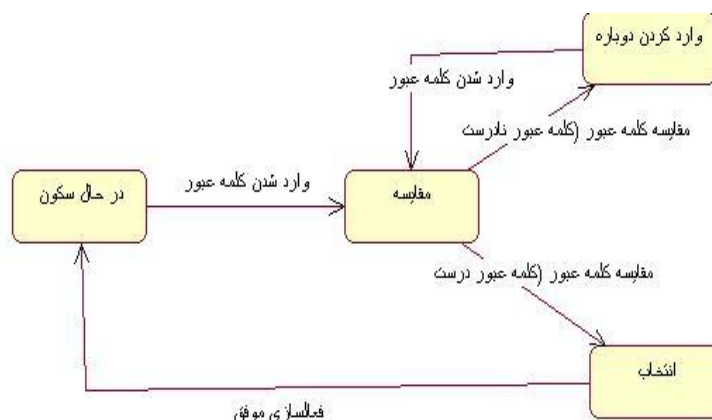
## نمودار نمایش حالت ( State Diagram )

برای هر سیستم شی گرا حالت سیستم توسط دو مورد زیر مشخص می شود:

- ۱- حالت هر شیء در هنگامی که سیستم در حال اجرای عملکرد خود می باشد. هر شیء خود دارای دو گونه حالت می باشد:

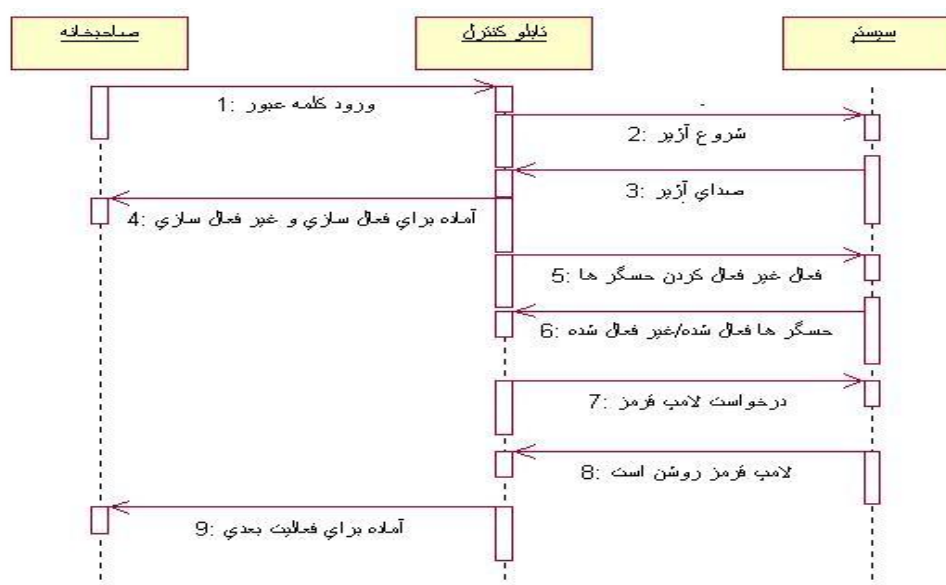
۱- حالت انفعالی

۲- حالت فعال



شکل ۱۴: نمودار انتقال حالت های سیستم

۲- حالت هر سیستم از دیدگاه خارجی و در حالی که سیستم عملکرد خود را اجرا می کند. نمودار ترتیب (Sequence Diagram)



## تست های فصل ۱۸: مدلسازی تحلیل شی گرا

- ۱- کدام یک از اصول بنیادی برای تحلیل توسط کود و یوردون بکار برده می شوند.  
(الف) توصیف عملکرد (ب) مدلسازی دامنه اطلاعات (ج) دیده شدن جزئیات بیشتر (د) تمام موارد
- ۲- کدام یک از گزینه های زیر نادرست است؟  
(الف) هیچ توافق جهانی بر سر مفاهیم مبنایی OOA وجود ندارد.  
(ب) اتصالات اشیا در OOA باید نشان داده شود.  
(ج) نظریه کود و یوردون بعنوان مبنای OOA است.  
(د) رفتار اشیا در OOA باید مدلسازی شود.
- ۳- چه کسی OOA را انجام می دهد؟  
(الف) مهندس نرم افزار (ب) طراح (ج) کاربر (د) ۲ و ۳
- ۴- کدام یک از گزینه های زیر در مورد تحلیل شی گرا نادرست است؟  
(الف) روش آن نسبت به مهندسی اطلاعات اندکی تفاوت دارد.  
(ب) روش آن تغییر بنیادی نسبت به تحلیل ساخت یافته نشان می دهد.  
(ج) محصول آن یک مدل شی گرا است.  
(د) فرآیند توسعه میکرو روش رومبو است.
- ۵- کدامیک از گزینه های زیر از وظایف مهندس نرم افزار برای اجرای تحلیل شی گرا نیست؟  
(الف) ساخت یک مدل شی - رفتار (ب) شناسایی موارد کاربرد  
(ج) ارزیابی مشخصات مشتری (د) ساخت یک مدل شی - واسط
- ۶- فرآیند OO در کار خود به اشیا نیاز ندارد.  
(الف) درست (ب) نادرست
- ۷- فرآیند OO در آغاز کار خود توسط ... (اگر در کنترل فرآیند بکار گرفته شود) شروع می شود.  
(الف) انسان (ب) ماشینها (ج) بعضی برنامه ها (د) هر ۳ مورد
- ۸- در اثنای تحلیل خواسته ها باید به اطلاعات اصلی تاکید شود.  
(الف) درست (ب) نادرست
- ۹- اطلاعات مربوط به یک چیز باید در میان چند کلاس توزیع شوند.  
(الف) درست (ب) نادرست
- ۱۰- نمایشی ساختار افراز، CRC را در اختیار تحلیلگر قرار می دهد.  
(الف) درست (ب) نادرست
- ۱۱- پکیج ....  
(الف) از لحاظ هدف مانند زیرسیستم است. (ب) در UML نشان داده نمی شود.  
(ج) از لحاظ محتویات مانند سیستم است. (د) همان سیستم است.
- ۱۲- حسگر به وضعیت ... وابسته است.  
(الف) پکیج (ب) سیستم (ج) حسگرها (د) الف و ج
- ۱۳- ساختن یک نمودار گذرا برای سیستم وظیفه کیست.  
(الف) مهندس نرم افزار (ب) طراح (ج) تحلیل گر (د) کاربر
- ۱۴- UML ....  
(الف) برای نشان دادن برخی از رفتار پویای اشیا و کلاس هایی است.  
(ب) تلفیقی از نمودارهای حالت، مشارکت، ترتیب و فعالیت است.  
(ج) زبانی برای مدلسازی است.  
(د) هر ۳ مورد
- ۱۵- مدل رفتار اشیا رفتار کلی سیستم OO است.  
(الف) درست (ب) نادرست