



عنوان

## تحلیل سیستم انتخاب واحد دانشگاه

تهیه کننده: علیرضا شیخی نیا

### چکیده

در این پروژه جهت بدست آوردن نیازهای سیستم نرم افزاری ابتدا با کاربران مصاحبه ای انجام گرفته و به تحلیل و بررسی مشکلات و نیازمندی ها پرداخته شده است. به طور کلی سیستم جاری تحلیل شده و نیازمندی ها مشخص شده است، سپس با توجه به نیازمندی ها ، سیستم کامپیوتری تعریف شده است. برای تحلیل سیستم کامپیوتری از نرم افزار Rational Rose استفاده گردیده که از نمودارهای use case، sequence، collaboration و class برای تحلیل بهره گرفته شده است.

### مقدمه

### تحلیل سیستم جاری

نام	X
سمت	مسئول ثبت
عناوین کلیدی مسئولیت ها	تکمیل مشخصات بایگانی دانشجویان، ثبت انتخاب واحد و حذف و اضافه، تکمیل پرونده فراغت از تحصیل، رعایت آیین نامه آموزشی، تهیه کارنامه برای دانشجویان

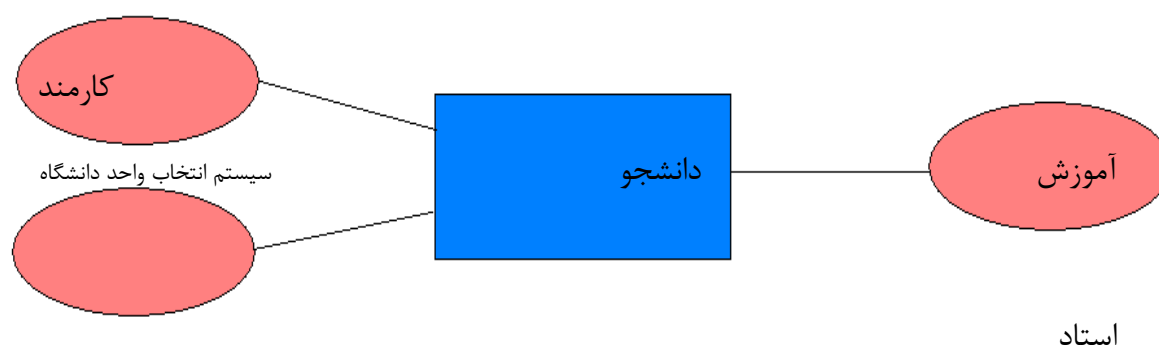
نتیجه کار	ثبت انتخاب واحد
برای	سیستم انتخاب واحد دانشجویان
مسائل دخیل در موفقیت	سیستم نرم افزاری کارآمد با پوشش کامل نیازمندی ها
موارد تسهیل کننده وظائف	<p>سیستم نرم افزاری با کارایی بالا به طوریکه در آن موارد زیر به وضوح مشخص باشد:</p> <p>– ثبت نام برای هر دانشجو بر اساس شماره دانشجویی باشد</p> <p>– ثبت لیست دروس انتخابی دانشجو به راحتی امکان پذیر باشد</p> <p>– رعایت اتوماتیک پیش نیازها و هم نیازها</p> <p>– سیستمی که آیین نامه برای آن تعریف شده باشد</p> <p>– سیستم این قابلیت را داشته باشد که مشروطی ها و اخراجی ها را مشخص کند</p> <p>– ترم های مشروطی مشخص باشد</p> <p>– تمامی اطلاعات دانشجویان مثلا تعداد واحد هایی که پاس کرده اند همراه با اسامی دروس قابل مشاهده و پرینت باشد</p> <p>– تهیه لیست حضور غیاب کلاسی برای اساتید</p> <p>– مشخص بودن اسامی کسانی که انتخاب واحد کرده اند و کسانی که انتخاب واحد نکرده اند</p>
موانع انجام وظایف	امنیت پایین – مشکلات اجرایی آیین نامه – زمانبر بودن انجام کارها

نام	X
سمت	مسئول پشتیبانی انتخاب واحد
عناوین کلیدی مسئولیت ها	ثبت دروسی که باید در هر نیم سال ارائه شود_ثبت ظرفیت کلاس ها_زمان تشکیل کلاسها و زمان امتحانات که توسط مدیر گروه مشخص شده است_بر طرف کردن مشکلات دانشجویان فارغ تحصیل_ثبت نمرات دانشجویان_اصلاح اطلاعات در صورت نیاز
نتیجه کار	ثبت و اصلاح نمرات
برای	سیستم انتخاب واحد دانشجویان
مسائل دخیل در موفقیت	سیستم نرم افزاری کارآمد با پوشش کامل نیازمندی ها
موارد تسهیل کننده وظائف	سیستم نرم افزاری که در آن موارد زیر تعریف شده است _لیست دروسی که دانشجویان پاس کرده اند مشخص باشد _قیمت دروس انتخابی همراه با تمامی مشخصات قابل تعریف باشد _اسامی دانشجویان مشروطی مشخص باشد _به دانشجویان مشروط امکان انتخاب بیش از ۱۴ واحد را ندهد _به دانشجویان ممتاز (معدل A) امکان انتخاب تا ۲۴ واحد را ندهد _بازه زمانی انتخاب واحد(تاریخ شروع و پایان)قابل تعریف باشد _قابلیت پرینت لیست دروس برای اساتید که آنها را تدریس می کنند _جلوگیری از انتخاب دروسی که باهم تداخل دارند _قابلیت تعریف ظرفیت برای هر کدام از دروس _تهیه لیست حضور و غیاب برای هر درس به استاد مربوطه _آیین نامه دانشگاه برای سیستم قابل تعریف باشد _لیست دروس مربوط به هر رشته از قبل تعریف شده باشد _دروس هم نیاز و پیشنیاز برای سیستم تعریف شده باشد _نهایی شدن لیست دروس انتخابی دانشجویان پس از حذف و اضافه
موانع انجام وظایف	امنیت پایین_مشخص نبودن قیمت دروس_زمانبر بودن مراحل انتخاب واحد

مساله	روش حل مساله
مشکلات در اجرای آیین نامه دانشگاه	تعریف آیین نامه دانشگاه برای نرم افزار
زمانبر بودن انجام کارها	نرم افزاری کردن کارها
مشکلات در کنترل دروس انتخابی	تعریف کنترل های لازم در سیستم
عدم وجود پایگاه داده یکپارچه	راه اندازی سیستم اتوماسیون و پایگاه داده یکپارچه
وجود سیستم بایگانی یکپارچه سنتی و عدم امکان دسترسی آسان	راه اندازی سیستم نرم افزاری
امنیت پایین	راه اندازی سیستم نرم افزاری

## تحلیل سیستم کامپیوتری با استفاده از نرم افزار Rational Rose

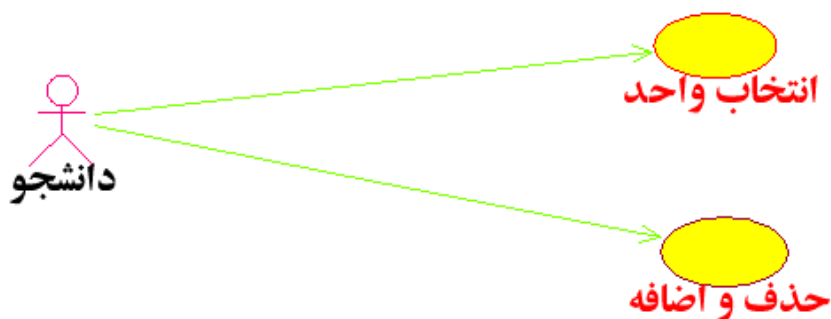
نحوه انتخاب ACTOR ها



با توجه به نمودار فوق موجودیتهای خارجی که با سیستم در ارتباط هستند مشخص می شود. هر کدام از این موجودیتهای خارجی به عنوان کاندیده هایی هستند که می توانند به عنوان actor هایی برای سیستم تشخیص داده شوند. در مجموع در سیستم انتخاب واحد سه actor داریم. پس از شناسایی actor ها ما دیگر می دانیم چه کسانی با سیستم کار می کنند پس به راحتی می توانیم دریابیم که این افراد از سیستم چه می خواهند این که این افراد از سیستم چه می خواهند یعنی usecase. برای هر نیامندی یک usecase در نظر می گیریم. مجموعه این usecase ها همان usecase diagram را تشکیل می دهد، در واقع ما با شناسایی actor ها به استخراج usecase ها میرسیم.

## (usecase diagram) دیاگرام موردهای استفاده

یک usecase رفتار سیستم را توصیف می کند، که شامل تقابل بین سیستم و بازیگران می باشد. عبارت کلی یک usecase یک الگوی رفتاری توانایی های سیستم و یک دنباله تراکنش های وابسته به هم می باشد، که بوسیله ی سیستم و بازیگران انجام می پذیرد و دیاگرام موردهای استفاده یک نمای سطح بالای سیستم از دید بازیگران سیستم به ما می دهد و نحوه ی برخورد آن با دنیای بیرون را مشخص می کنند. این دیاگرام در طول تحلیل سیستم برای بدست آوردن نیازمندی ها و نشان دادن چگونگی کارکرد سیستم بکار می رود. برای رسم یک دیاگرام مورد استفاده ابتدا در نمای usecase view روی آیکون main دو کلیک کنید تا دیاگرام مربوط به آن باز شود، کافی است که از جعبه ابزار سمت چپ، آیکون new use case را انتخاب کرده و بعد از قرار دادن روی دیاگرام با دو کلیک کردن روی آن در فرم specification نام، توضیحات و stereotype مربوط به آن را وارد نمایید. برای ساختن یک وابستگی جدید بین دو عنصر نیز روی آیکون association کلیک کرده و این وابستگی را با کلیک کردن روی عنصر اول و کشیدن تا عنصر دوم بسازید. سپس با دو کلیک کردن روی آن مشخصات آن را وارد نمایید.



جریان اصلی usecase (main flow) و جریان فرعی (alternative flow)

عمل actor	جریان اصلی	جریان فرعی
(۱) دادن رمز عبور برای سیستم		
	(۲) بررسی صحت رمز عبور	
(۳) انتخاب دروس		
	(۴) کنترل دروس انتخابی	(۴) پیام خطا
(۵) انتخاب استاد		
(۶) واریز شهریه		
	(۷) کنترل شهریه	(۸) پیام خطا
(۹) ثبت انتخاب واحد (نهایی)		
	(۱۰) صدور برگه تایید انتخاب واحد	

(۱) شرایطی که باید ایجاد شود تا usecase فعال شود (precondition)

دانشجو جهت انتخاب رشته رمز عبور خود را به سیستم می دهد

(۲) شرایطی که بعد از اتمام کار usecase ایجاد می شود (postcondition)

به دانشجو یک برگه تایید انتخاب واحد صادر می شود

(۳) هدف usecase

ثبت دروس انتخابی دانشجو در سیستم

شرح مختصری از فعالیت های سیستم برای نمودار توالی

دانشجو بعد از ورود به سیستم انتخاب واحد ، دروس مورد نظر خود را با رعایت پیش نیاز ها ، هم نیاز ها و زمان تشکیل کلاس ها و زمان امتحانات انتخاب کرده و در صورتی که یک درس با بیش از یک استاد ارائه شده باشد دانشجو استاد مورد نظر خود را نیز انتخاب می کند. سیستم نیز به صورت اتوماتیک دروس را بررسی می کند تا از تداخل زمانی و مشکلاتی از قبیل تعداد واحد انتخابی و ..... جلوگیری شود. بعد از این دانشجو فیش واریزی خود را ثبت می کند، در انتها برگه تایید انتخاب واحد برای دانشجو صادر می شود.

## دیگرام توالی (sequence diagram)

دیگرام توالی بصورت منظم و در یک توالی زمانی پشت سر هم ارتباطات متقابل اشیا را به ما نشان می دهد. این دیگرام برای انجام عمل خاصی در یک usecase مشخص مراحل انجام کار را مرحله به مرحله به شما نشان می دهد، یعنی دنباله ای از رویدادها را برای انجام یک عمل مشخص می سازد.

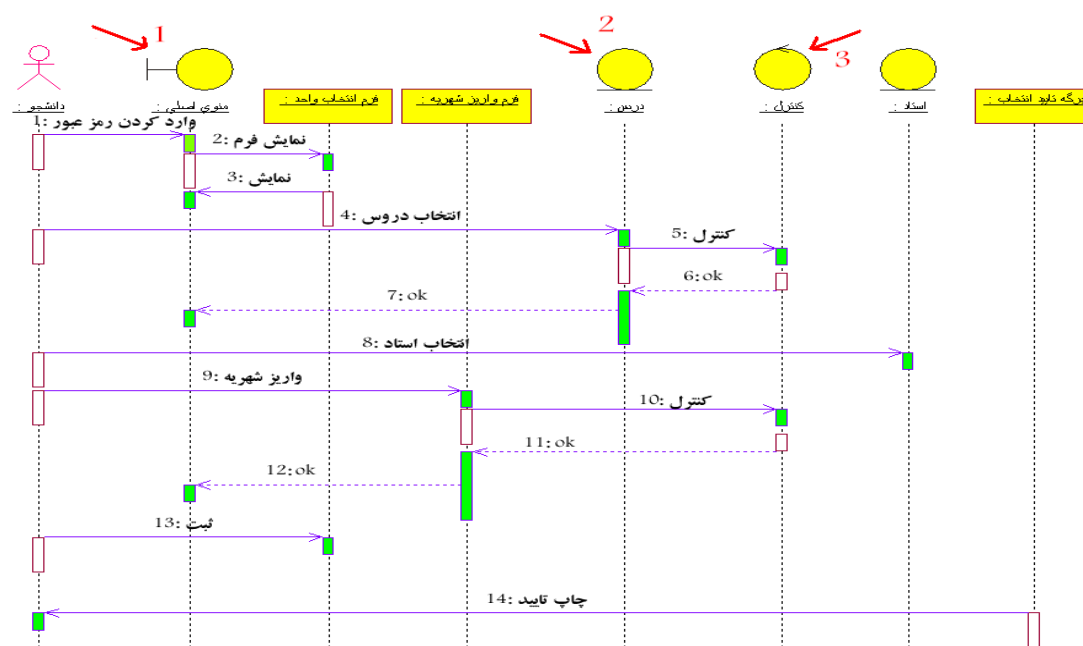
در مراحل تحلیل و طراحی برای فهم نحوه ی عملکرد سیستم از این دیگرام استفاده می شود. جهت ساخت یک دیگرام توالی در قسمت browser رشنال رز در قسمت usecase view روی نود مورد استفاده ی مورد نظر راست کلیک کرده و new و سپس sequence diagram را انتخاب نمایید. روی نود جدید ایجاد شده دو کلیک کنید تا پنجره ی دیگرام مربوط به آن باز شود.

۱) **Boundary**: اجزای لازم برای برقراری ارتباط سیستم با یک بازیگر را در خود دارند. این کلاسها می توانند پنجره ها، سنسورها، ترمینالها یا ... باشند. مثلاً پنجره ی گرفتن اسم رمز جهت ورود به برنامه، یک boundary می باشد. این نوع کلاسها معمولاً جهت استفاده، با یک کلاس از نوع کنترلی در ارتباط هستند. در این سیستم فرم اصلی یک boundary است.

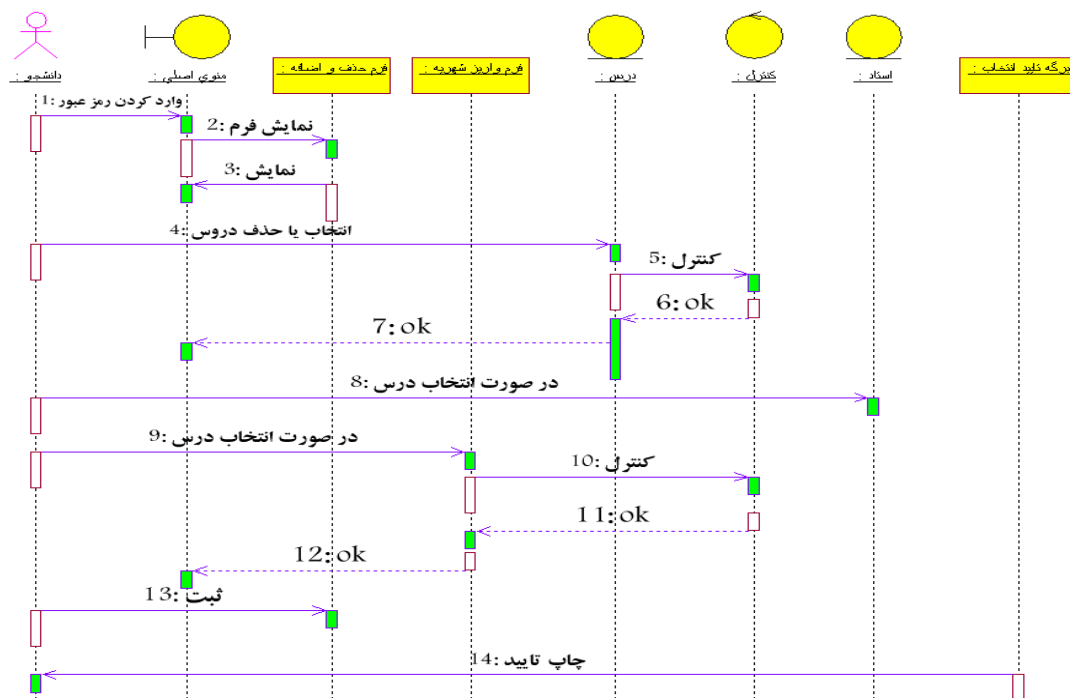
۲) **Control**: این کلاسها معمولاً اشیا دیگر و رفتارهای تعبیه شده در یک usecase را کنترل می کنند. در این سیستم نیز بخش کنترلی وجود دارد که صحت را بررسی می کند.

۳) **Entity**: این کلاسها اطلاعاتی را که باید توسط سیستم ذخیره گردند را در خود نگهداری می کنند. محل نگهداری ساختمان داده های منطقی سیستم، این کلاسها می باشند. در این سیستم درس entity می باشد. استاد، فرم شهریه و انتخاب و برگه تایید object هستند که از ابزار های نمودار توالی است.

دیگرام توالی انتخاب واحد



دیگرام توالی حذف و اضافه

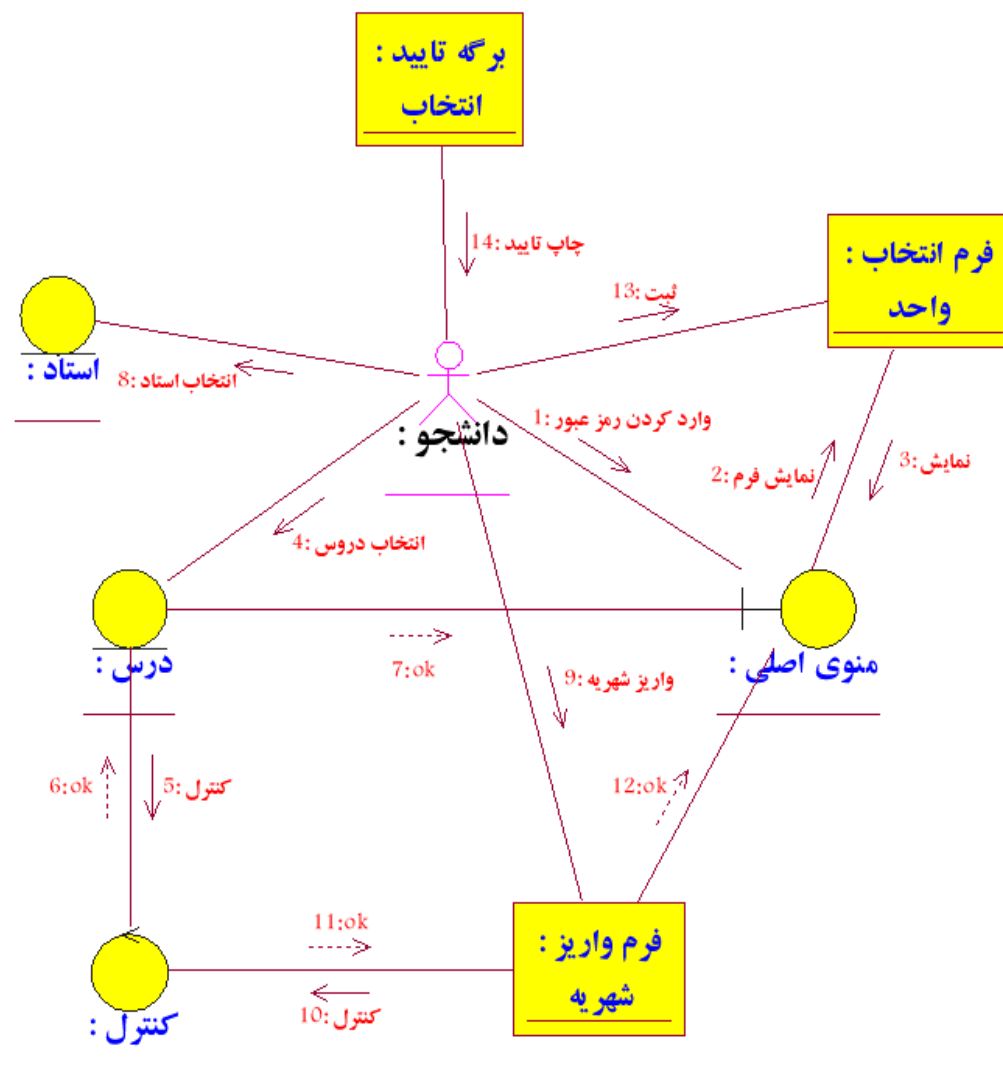


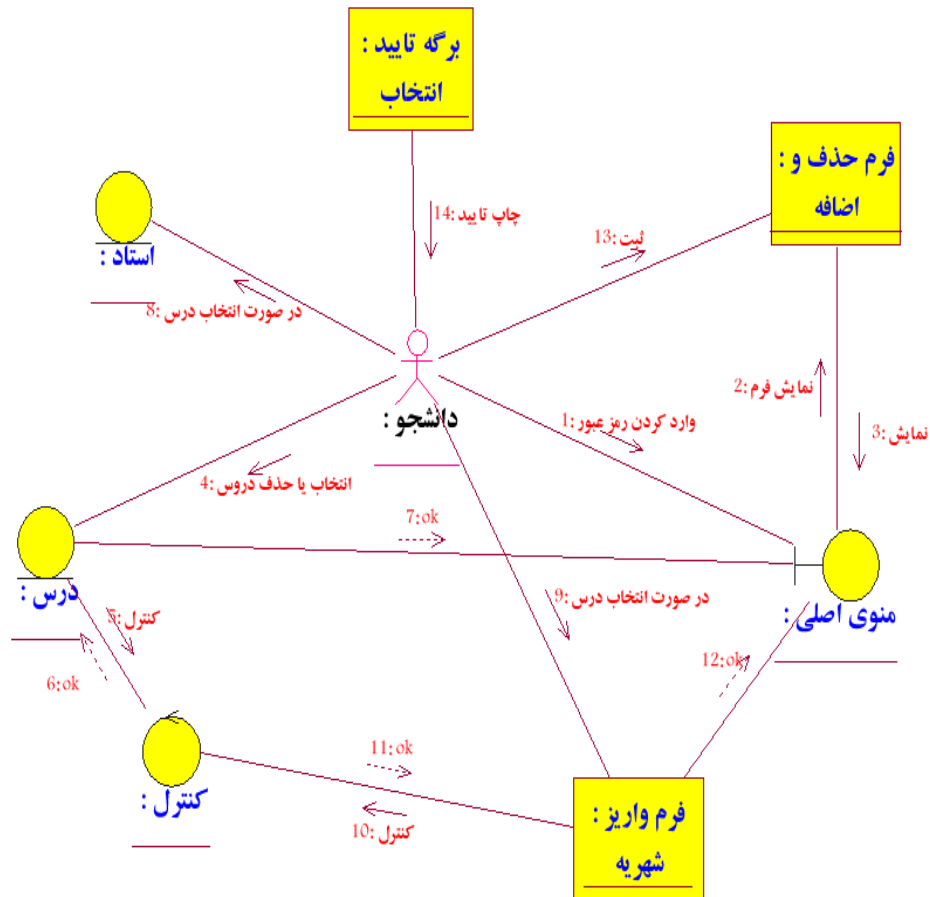
### دیگرام همکاری (collaboration diagram)

یک نما از ارتباطات ساختاری بین اشیا در مدل فعلی را بما می دهد . تاکید این دیگرام بر ارتباط بین اشیا است در حالیکه تاکید دیگرامهای توالی بر روی دنباله ای از رویدادها بود . این نوع دیگرام شامل اشیا، لینک ها، وپیغام ها می باشد.

برای درک وفهم چگونگی رفتار سیستم واتخاذ هر گونه تصمیمی در این مورد از این دیگرام استفاده کنید. برای ساخت این دیگرام کافی است روی دیگرام توالی ایستاده و کلید f5 را بزنید. همچنین اگر ابتدا این دیگرام را ساخته سپس روی آن ایستاده f5 را بزنید دیگرام توالی آن ساخته می شود.







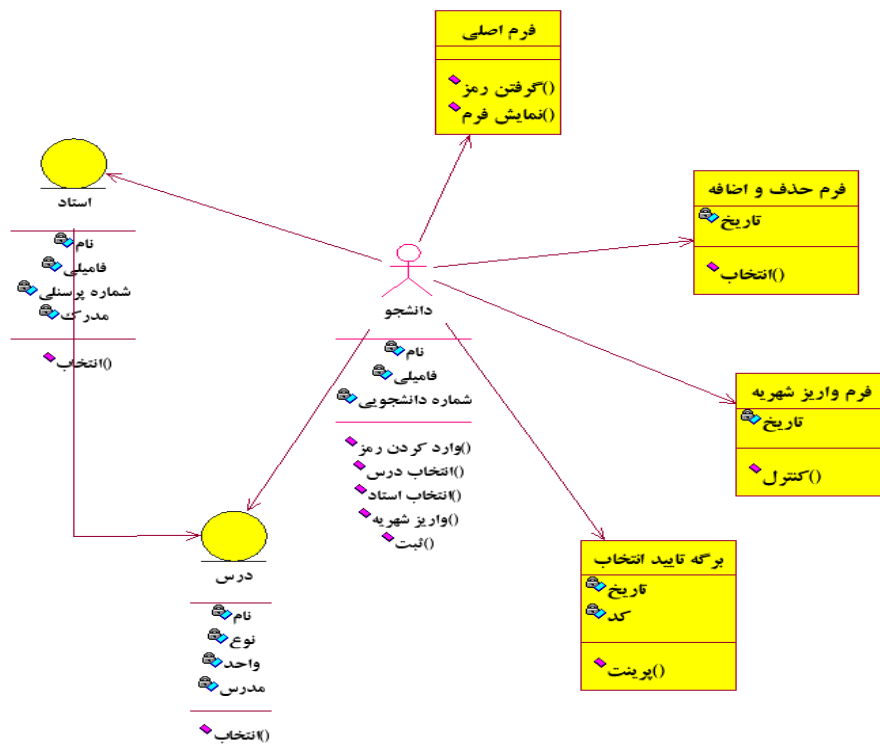
### دیاگرام کلاس (class diagram)

این دیاگرام به شما کمک می کند تا نمای ساختاری سیستم تان را بصورت بصری (visual) در آورید . این دیاگرام از معمول ترین دیاگرامهای UML می باشد. این دیاگرامها جزئیات هر کلاس و ارتباطات بین آنها را نشان می دهد و پایه و اساس دیاگرامهای اجزا و پیاده سازی می باشد.

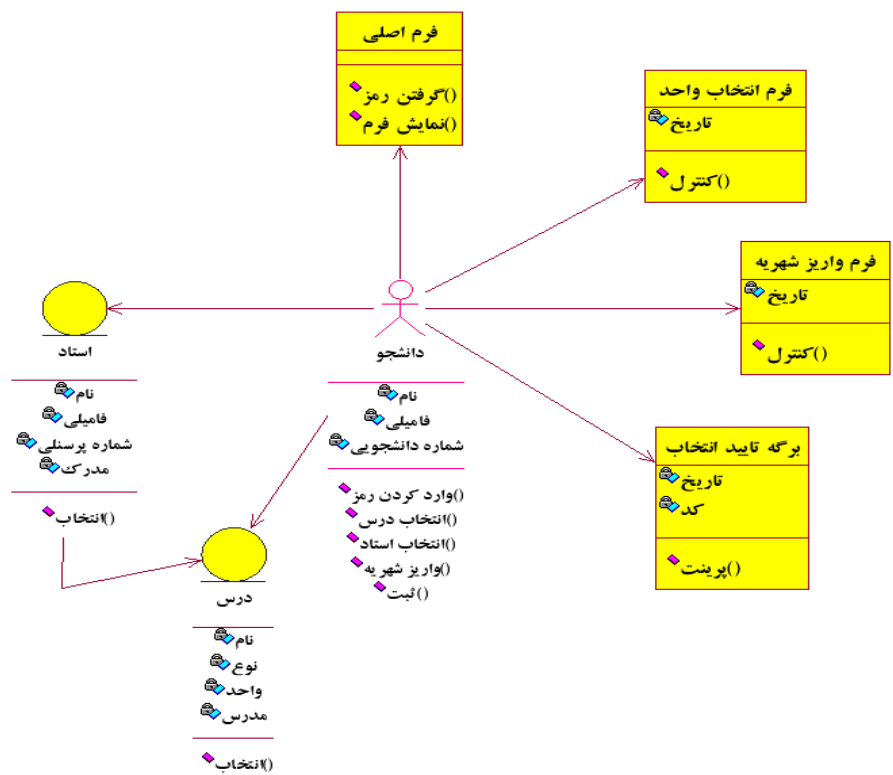
برای add کردن یک متد به یک کلاس کافی است، در browser روی آن راست کلیک کرده و `new > operation` را انتخاب نماییم، سپس در کادر فعال شده مشخصات متد مورد نظر را وارد کنیم. در این سیستم اعمالی که هر اشیا باید انجام دهند یک متد است.

حال باید این کلاسهای تولیدی و کلیه ی attribute های مربوط به کلاسهای entity را وارد نماییم. وارد کردن یک attribute جدید به یک کلاس، با کلیک راست کردن روی آن و انتخاب گزینه ی `new > attribute` صورت می پذیرد در این سیستم هر اطلاعاتی که اشیا در خود دارند یک خصوصیت است. .

## دیاگرام کلاس انتخاب واحد



## دیاگرام کلاس حذف و اضافہ



## نتیجه گیری

رعایت آیین نامه دانشگاه به صورت اتوماتیک\_سرعت بخشیدن به روند انجام کارها\_صرفه جویی در زمان و هزینه\_بالا رفتن امنیت اطلاعات و سهولت دسترسی به اطلاعات