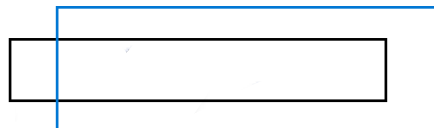


# تجزیه و تحلیل و طراحی سیستم



## مقدمه :

هر انسانی تلاش می کند تا به موفقیت برسد برای این کار بایستی مراحل لازم و ضروری را بگذرانند که عبارتند از :

- ۱- تشخیص نیازهای لازم
- ۲- برنامه ریزی برای رفع نیازها
- ۳- اجرای طرح
- ۴- ارزشیابی کارهای انجام شده

برای توسعه ی سیستم های اطلاعاتی کامپیوتری نیز مراحل نیاز است تا این کار به سرانجام برسد . مراحل آنرا می توان به شکل زیر ارائه کرد :

توسعه ی سیستم → استفاده از سیستم → آزمایش و تست سیستم → ساخت سیستم اطلاعاتی → طراحی و اجرای آن → تشخیص نیازها

## سیستم ها و سیستم های اطلاعاتی

### تعریف سیستم :

سیستم بطور عمومی به یک جعبه سیاه گفته می شود همراه با ورودی هایی که به آن وارد می شوند و خروجی هایی که از آن خارج می شوند .

### - مدل جعبه ی سیاه عمومی یک سیستم :



### - مدل جعبه ی سیاه با بازخورد (شکست):



مدل بازخورد از خروجی های سیستم مراقبت می کند تا آنچه که از سیستم مورد انتظار است تولید شود .

## **تعریف سیستم اطلاعاتی :**

مجموعه ای از افراد، اشیاء، فرایندها، عملیات ها و ... می باشند که راجع به سازمانها و محیط آنها اطلاعاتی تهیه میکنند این اطلاعات بایستی برای اعضاء و مشتریان سازمان مفید باشند .

## **تحلیل ، تحلیل سیستم ها و تحلیل گر سیستم**

### **تعریف تحلیل :**

تحلیل فرایندی است که در این فرایند اشیاء به عناصر یا قسمتهای تشکیل دهنده ی آن تجزیه می شود .

### **تعریف تحلیل سیستم ها :**

قسمتهای مختلف سیستم را به اجزاء تشکیل دهنده ی آن مرتبط می نماید و فرایندهای پیچیده ی سیستم اطلاعاتی را به اجزاء ساده تر تبدیل می کند .

### **هدف از تحلیل سیستم ها :**

هدف از تحلیل سیستم ها این است که قسمتهای پیچیده ی سیستم را به بخشهای ساده تری تقسیم کند بطوریکه قابل فهم باشد و ارتباط بین قسمتها نیز حفظ شود .

### **تعریف تحلیل گر سیستم :**

فردی است که عهده دار تحلیل سیستم می باشد . همچنین در بسیاری از موارد بعنوان طراح سیستم نیز انجام وظیفه میکند و باتوجه به نیازهای سیستم جدید ، آنرا طراحی و آزمایش کند .

### **طراحی سیستم :**

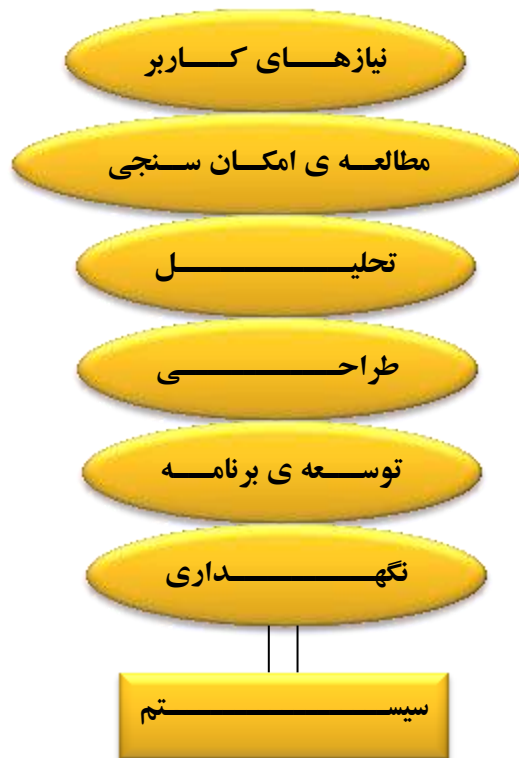
زمانیکه تحلیل گر از نیازهای سیستم جدید مطمئن گردید بایستی طراحی سیستم جدید را آغاز کند. پس از مشخص نمودن نمودار جریان اطلاعات و اطلاعات جاری تهیه ی مدل منطقی سیستم را شروع می کند و با استفاده از این مدل و همچنین تعیین نیازهای اطلاعاتی سیستم جدید و ترکیب آنها مدل منطقی سیستم پیشنهادی را ارائه می کند .

### **افزایش کیفیت و زنجیره های کیفیت :**

با تحلیل و طراحی یک سیستم به ارزش آن سیستم افزوده نمی شود بلکه به کیفیت آن افزوده می شود. افزایش کیفیت یکی از اهداف اصلی ساخت یک محصول است .

در تولید نرم افزارهای کامپیوتری نیز تولید نرم افزار با کیفیت بالا از الزامات یک سیستم کامپیوتری می باشد.

زنجیره ی کیفیت سیستم توسط آقای « ابرین وچستر » ارائه شد که ساختار آن بصورت شکل زیر می باشد :



شکل ۱- زنجیره ی کیفیت سیستم

## چرخه ی حیات سیستم ها و تحلیل سیستم ها

### **چرخه ی حیات یا زندگی یک سیستم :**

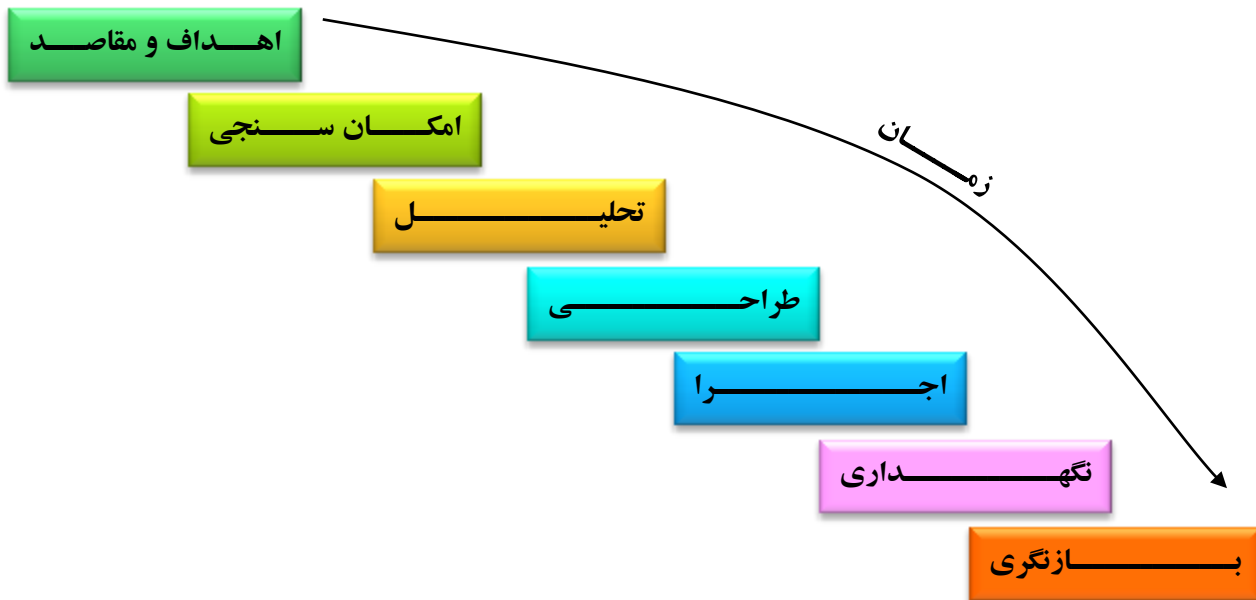
مراحل مختلف توسعه ی یک پروژه از زمان شروع ( فکر ایجاد سیستم ) تا تولید سیستم نهایی ، چرخه ی زندگی یا حیات یک سیستم می باشد .

### **مراحل چرخه ی زندگی در طراحی سیستم های اطلاعاتی (ISD(Information System Design :**

- ۱- اهداف و مقاصد
- ۲- مطالعه ی امکان سنجی
- ۳- تحلیل سیستم ها
- ۴- طراحی سیستم ها
- ۵- اجرا ، کد نویسی ، آزمایش و نصب
- ۶- نگهداری و پشتیبانی
- ۷- توسعه و بازنگری

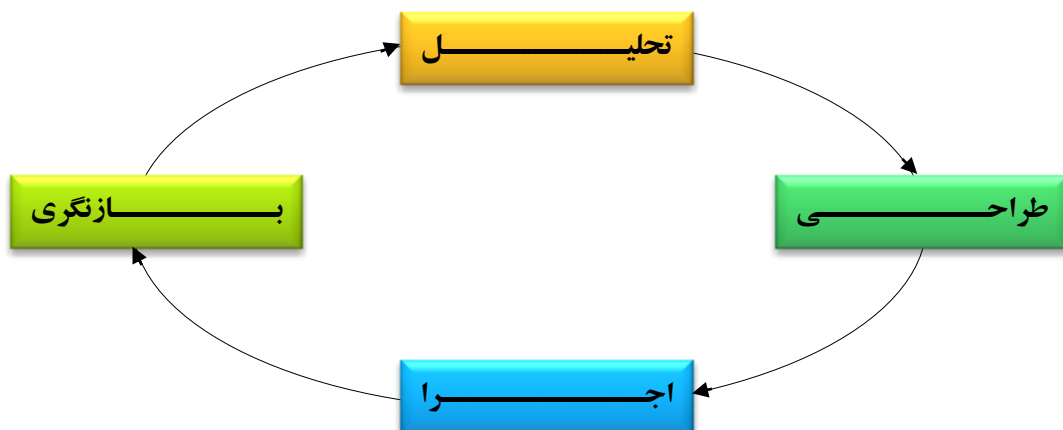
## چرخه ی زندگی آبشاری :

در این چرخه زمانیکه یک مرحله به اتمام می رسد مرحله ی بعدی شروع می شود .



## چرخه ی زندگی الگوسازی (Prototyping):

این چرخه یک روش تکراری است و ساختار آن به شکل زیر می باشد :  
در این روش اطلاعات قبلی استفاده شده و چرخه آنقدر تکرار می شود تا سیستم تکمیل شود .



در این روش کاربر می تواند تغییراتی را پیشنهاد کند قبل از آنکه توسعه ی سیستم کامل شود .

### چرخه ی زندگی توسعه ی سریع تقاضا (RAD) :

این روش سیستمهای کامپیوتری پیشرفته را بطور سریع و مؤثر کامل می کند.

مراحل چرخه ی زندگی RAD عبارتند از :

۱- برنامه ریزی پیش نیازها

۲- طراحی با کمک کاربر

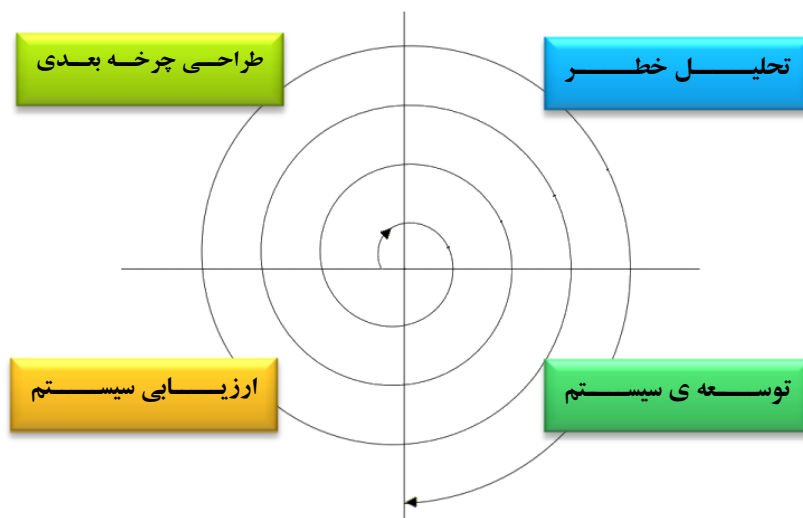
۳- توسعه و ساخت سیستم

در روش RAD از ابزارها و نرم افزارهای مهندسی نرم افزار استفاده می شود .

### چرخه ی زندگی به روش حلزونی :

در این روش در پروژه ی توسعه ی سیستم ها مراحل مختلف « تحلیل ، طراحی و توسعه » مرتباً تکرار می گردد .

مراحل این روش به شکل زیر می باشد :



### روش SSADM (Structured System Analysis and Design Method) :

به معنی روش تحلیل و طراحی سیستمهای ساخت یافته است . روشهای دیگری نیز وجود دارند که عبارتند از : SSA و

SSM از این روشها نیز برای تحلیل و طراحی سیستم استفاده می شود.

هدف از تحلیل و طراحی سیستم موارد زیر می باشد :

الف- مشخص نمودن نوع اطلاعات و فرایندهای مورد نیاز سیستم جدید.

ب- تعیین نیازهای کارکردی و غیر کارکردی سیستم جدید.

ج- مشخص نمودن زبان برنامه نویسی سیستم .

د-تعیین نیازها

## **گزارش نویسی و نمایش آن برای تحلیل سیستم ها :**

گزارش در شکل و اندازه های مختلفی وجود دارد. به گزارش هایی که تحلیل گر سیستم از آن استفاده می کند گزارشهای « رسمی » گفته می شود. این گزارشات نیازهای اساسی و اصلی را در سیستم تهیه می کند و به گزارشهایی که اطلاعات جانبی در سیستم را شامل می شوند گزارشهای « غیررسمی » گفته می شود. بعنوان مثال گزارشاتی که بیان می کند چه کارهایی با این سیستم نمی توان انجام داد جزء گزارشات غیررسمی است .

## **ساختار گزارش :**

صفحه ی عنوان ، محتویات، چکیده ، مقدمه ، بدنه ی اصلی ، نتیجه گیری و پیشنهادات ، مراجع و فهرست کتب ، پیوستها ، بخشهای دیگر .

## **ساختار روش SSM :**

پروفسور چک لند « Check Land » در سال ۱۹۷۰ روش SSM را ارائه داد. این روش شامل ۷ مرحله می باشد که عبارتند از :

مرحله ۱ – تعیین موقعیت سیستم

مرحله ۲ – تحلیل ایده ها و تصاویر دامنه دار راجع به سیستم

مرحله ۳ – تعاریف ریشه ای از سیستمهای ارزیابی شده با روش CATWOE

مرحله ۴ – ایجاد مدل مفهومی سیستم

مرحله ۵ – مذاکره برای امکان تغییرات با مقایسه ی مدل مفهومی و تصاویر دامنه دار

مرحله ۶ – امکان سنجی در مورد تغییراتی که در سیستم بایستی انجام گیرد .

مرحله ۷ – اجرای تغییرات

## **مرحله ۱ – تعیین موقعیت سیستم:**

تحلیلگران در ابتدا باید با هر مشکلی که برخورد می نمایند مقاومت کنند و ساختار و موقعیت سیستم را روشن کنند و زمانیکه درگیر مسأله می شوند آنرا سازماندهی می کنند .

## **مرحله ۲ – استفاده از تصاویر دامنه دار :**

تصاویر دامنه دار روشی است برای خلاصه نمودن اطلاعات و انتقال اطلاعات مهم به روش مؤثر و اقتصادی و متفکر؛ تصاویر دامنه دار حجم بسیار وسیعی از اطلاعات سیستم را در بر می گیرد.

مثال: شکل زیر تصاویر دامنه دار مربوط به مشکلات هتل را نشان می دهد.





### مرحله ۳ – تعاریف ریشه ای از سیستم با روش CATWOE :

CATWOE اصطلاحی است که از حروف زیر تشکیل شده است :

*C(Customers)* → مشتریان

*A(Actors)* → بازیگران

*T(Transformation)* → تغییر اطلاعات

*W(Weltanschauung)* → معنی دار شدن

*O(Owner)* → مالکان

*E(Environment)* → محیط

#### 😊 مشتریان *C(Customers)* :

کسانی هستند که در تغییر اطلاعات سیستم مؤثر باشند . مثلاً در سیستم پذیرش هتل « مهمانان ، حسابدار ، مدیر رستوران و ... » جزء مشتریان هستند .

#### 😊 بازیگران *A(Actors)* :

کسانی هستند که تغییرات را در سیستم انجام می دهند . در سیستم پذیرش هتل « مسئول پذیرش و رئیس پذیرش » جزء بازیگران هستند .

#### 😊 تغییر اطلاعات *T(Transformation)* :

منظور تغییر دادن اطلاعات درون سیستم است . در سیستم پذیرش هتل تغییر اطلاعات « مهمانان » باعث تغییر اطلاعات سیستم می شوند .

#### 😊 معنی دار شدن *W(Weltanschauung)* :

هدف سیستم را مشخص می کند. بعنوان مثال هدف سیستم پذیرش هتل « رفاه مهمانان و جذب مهمان بیشتر » است .

#### 😊 مالکان *O(Owner)* :

کسانی هستند که می توانند هرگونه تغییری را در سیستم ایجاد و یا متوقف کنند. در سیستم پذیرش هتل « رئیس هتل (مالک هتل) می باشد .

#### 😊 محیط *E(Environment)* :

محیطی که سیستم در آن اجرا می شود . مانند سیستم پذیرش هتل که در «هتل» اجرا می شود .

مثال: CATWOE را برای سیستم درمانی بنویسید .

Customers ← بیماران  
Actors ← پزشکان و پرستاران  
Transformation ← بیماران  
Weltanschauung ← درمان بیماران  
Owner ← وزارت بهداشت یا هیئت اُمنّا  
Environment ← بیمارستان

#### مرحله ۴- ایجاد مدل مفهومی سیستم:

مدل های مفهومی سیستم نمودارهایی هستند که فعالیتهای موردنظر مشتریان را نشان می دهند .

مثال: مدل مفهومی پذیرش مهمان در هتل بصورت شکل زیر می باشد :



### مرحله ۵ – مذاکره برای تغییرات در سیستم:

مذاکره برای تغییرات در سیستم را به وسیله ی تصاویر دامنه دار و مدل های مفهومی و سایر نمودارها انجام می دهیم . تفاوت ها و تشابه ها در این نمودارها بررسی می شوند تا مشخص گردد چه تغییراتی در سیستم بایستی صورت گیرد .

### مرحله ۶ – قضاوت در مورد تغییراتی که امکان پذیر باشد .

مذاکره حداقل بایستی بین «مشری ، صاحب سیستم و حل کننده ی سیستم» صورت گیرد .

### مرحله ۷ – اجرای طرح

### نحوه ی ارتباط موجودیت ها (ERD(Entity Relation Diagram) :

۱- ارتباط یک به یک (1:1)



۲- ارتباط یک به چند (1:n)



۳- ارتباط چند به چند (n:m)



### موجودیت ها :

موجودیت ممکن است فیزیکی یا مفهومی باشد مثل یک مشتری ، یک ساختمان ، یک پروژه و ...

### صفات :

خصوصیاتی هستند که یک موجودیت می تواند به نفع سیستم اطلاعاتی از آنها بهره ببرد. مثل خصوصیات یک دانشجو که عبارتند از: نام ، نام خانوادگی ، شماره شناسنامه ، کد دانشجویی و ...

### ارزشهای یک صفت :

داده های واقعی هستند که برای هر صفت موجودیت وجود دارند. مثلاً برای نام دانشجو : علی ، رضا و ...

### کلیدها :

عبارتند از یک صفت یکتا یا منحصر به فرد که برای موجودیت در سیستم وجود دارد . مثل کد دانشجو

### انواع ارتباطات منطقی موجودیت ها :

۱- ارتباطات اختیاری

۲- ارتباطات انحصاری

۳- ارتباطات بازگشتی

### ارتباطات اختیاری :

زمانی که کلمه ی «باید» وجود دارد ارتباط اجباری می باشد و زمانی که کلمات « اگر ، ممکن است و شاید» وجود داشته باشد روابط اختیاری است . مثلاً یک منزل ممکن است توسط یک مالک اجاره داده شود و یا یک مالک باید یک یا چند منزل را اجاره دهد که ارتباط ERD آن به شکل زیر نشان داده می شود .



رابطه ی اختیاری ← ممکن است اجاره داده شود

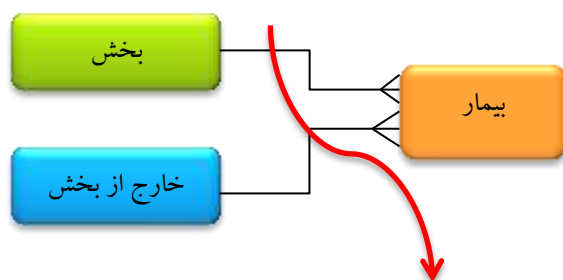
رابطه ی اجباری ← باید اجاره دهد

رابطه ی اجباری ← وام گیرنده باید با یک یا چند وام در ارتباط باشد

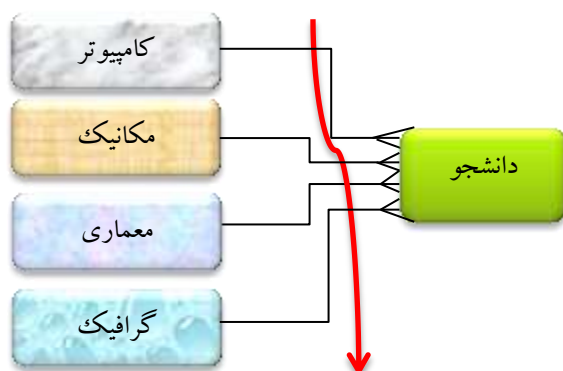
رابطه ی اختیاری ← وام گیرنده ممکن است با یک یا چند وام در ارتباط باشد

## ارتباطات انحصاری :

این نوع ارتباط زمانی به وجود می آید که یک ارتباط مانع از وجود دیگری شود .  
بعنوان مثال : در سیستم بیمارستان بیمار می تواند در بخش یا خارج از بخش باشد . که هر دو ارتباط ERD آن به صورت زیر نمایش داده می شود .



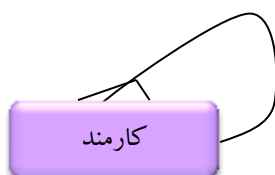
یک بیمار نمی تواند در یک زمان هم در بخش و هم در خارج از بخش باشد



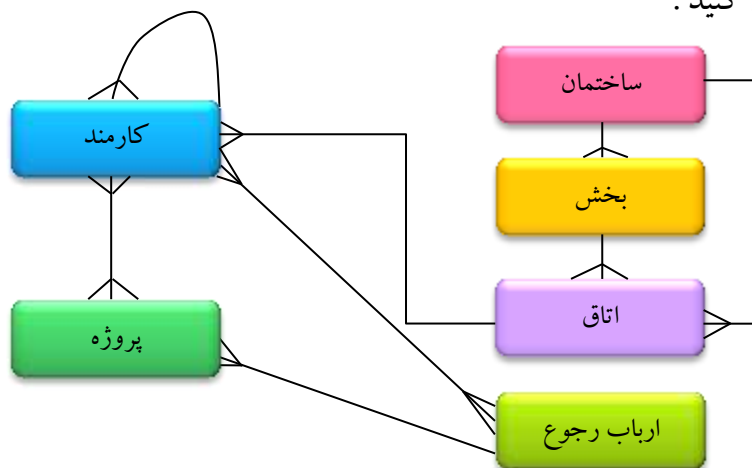
یک دانشجو نمی تواند چند رشته را بخواند

## ارتباطات بازگشتی :

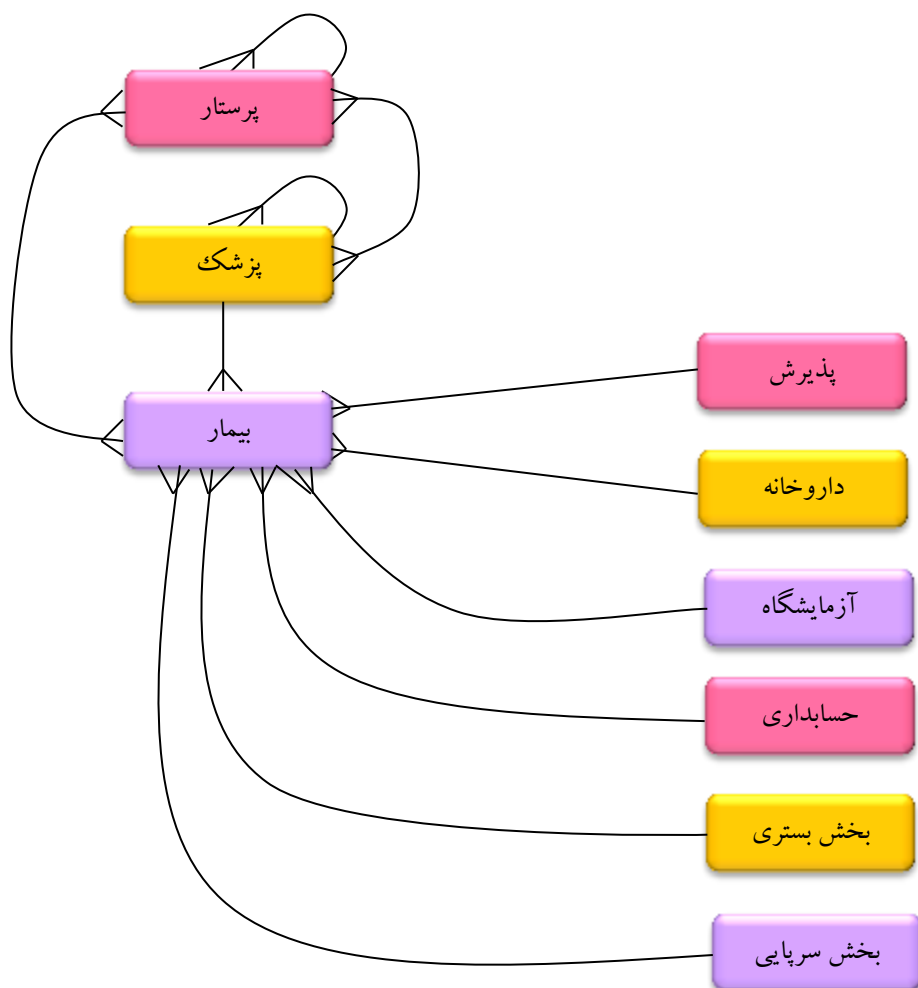
این ارتباط بیانگر ارتباط یک موجودیت با چند موجودیت دیگر از همان نوع می باشد . مثل ارتباط کارمند با کارمند



**مثال :** در پایگاه داده های یک مؤسسه موجودیت های (کارمند - ساختمان اتاق - بخش - پروژه و ارباب رجوع) وجود دارد نمودار ERD آنها را رسم کنید .



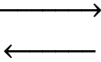
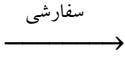

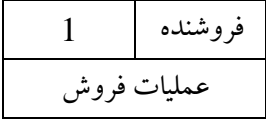
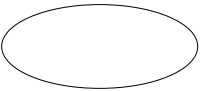


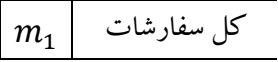
**مثال :** در پایگاه داده های یک بیمارستان موجودیت های بیمار ، پزشک ، پرستار ، پذیرش ، داروخانه ، آزمایشگاه ، حسابداری ، بخش بستری ، بخش سرپایی و ... وجود دارد . نمودار ERD آنها را رسم کنید.



## نمودار جریان داده ها (DFD(Data Flow Diagram) :

نمودارهای جریان داده ها یک روش مدل سازی سیستم می باشد . این نمودارها شامل پردازش و فعالیتهای سیستم ، اطلاعات در حال جریان به سیستم و یا اطلاعات در حال خروج از سیستم می باشد.

اجزای نمودار جریان داده ها یا DFD عبارتند از :

Component (شی)	Symbol (شکل)	Example (مثال)
Data Flow		
Process (پردازش یا فعالیت)		
External Entity (موجودیت خارجی)		
Data Store (پایگاه داده ها)		

## سطوح نمودار جریان داده ها :

در سیستم اطلاعاتی پیچیده ی منطقی این امکان وجود ندارد که تمامی اطلاعاتی مربوط به سیستم را در یک صفحه نشان دهیم ؛ بدین منظور نمودار جریان داده ها را در DFD به سطوح دیگری تقسیم می کنیم .

سطح صفر (نمودار زمینه ای)

سطح یک

سطح دو

.

.

.

## نمودار سطح صفر DFD :

بالاترین سطح نمودار DFD می باشد. این نمودار نشان می دهد که سیستم چه اثری بر محیط خود دارد .  
این نمودار تشکیل شده از : سیستم ، موجودیت های خارجی و جریان داده ها بین سیستم و موجودیت های خارجی.  
**بعنوان مثال نمودار زمینه ای (سطح صفر) یک فروشگاه ویدئو به شکل زیر است :**



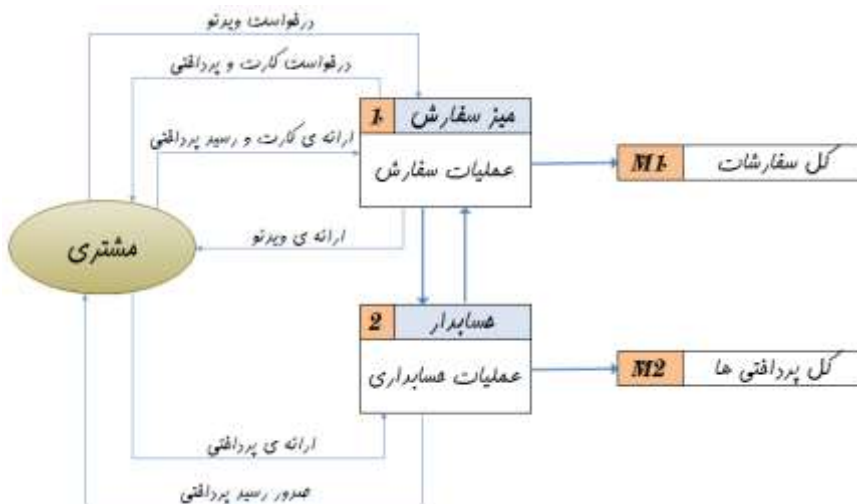
**نمودار زمینه ای (سطح صفر) بیمارستان :**



## نمودار سطح یک DFD :

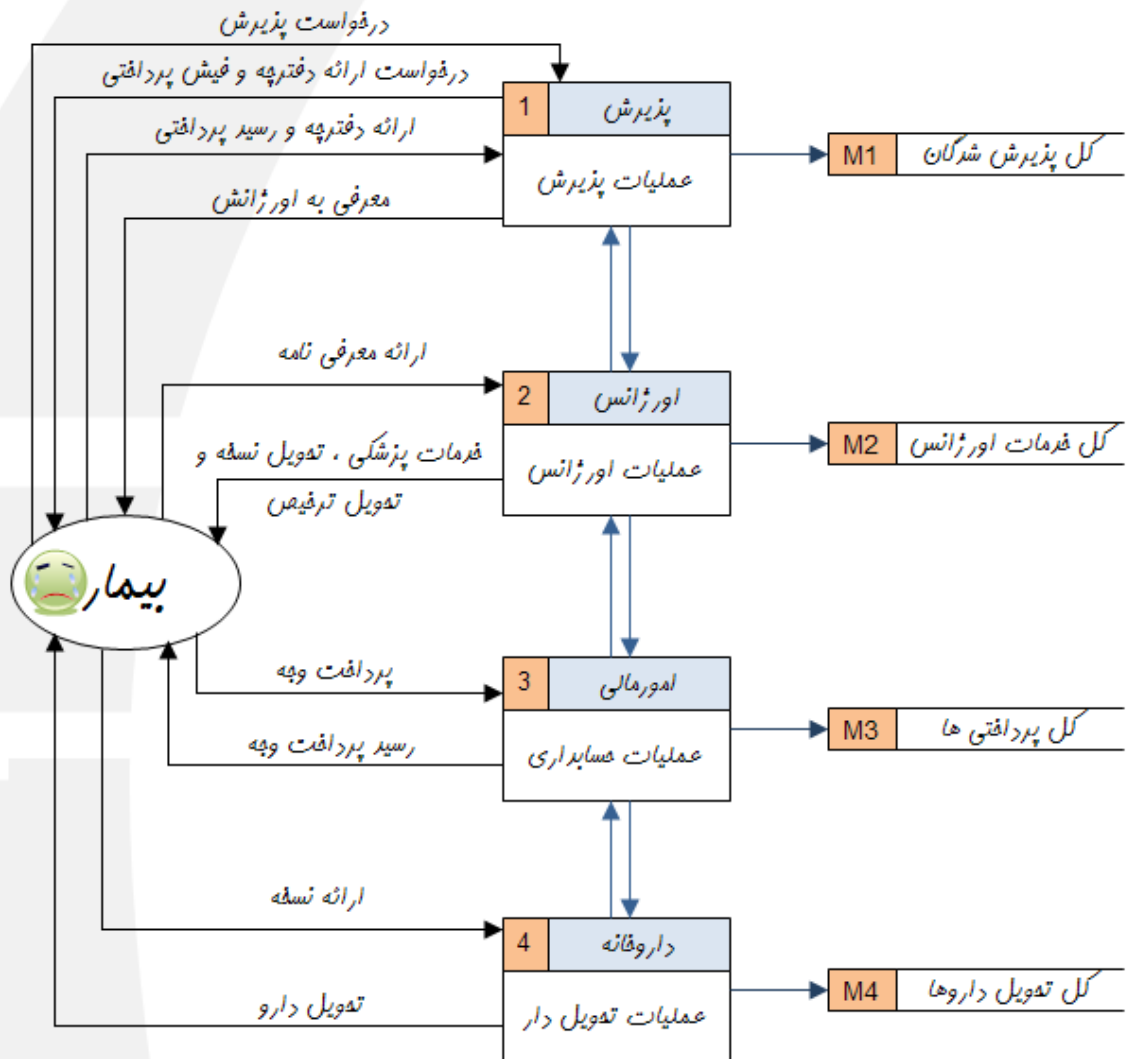
این نمودار یک درک عمومی از یک سیستم اطلاعاتی را نشان می دهد در این نمودار فعالیتهای اصلی سیستم ، پایگاه داده ها و جریان بین داده ها نشان داده می شود.

**بعنوان مثال نمودار DFD سطح یک فروشگاه ویدئو به صورت زیر است :**





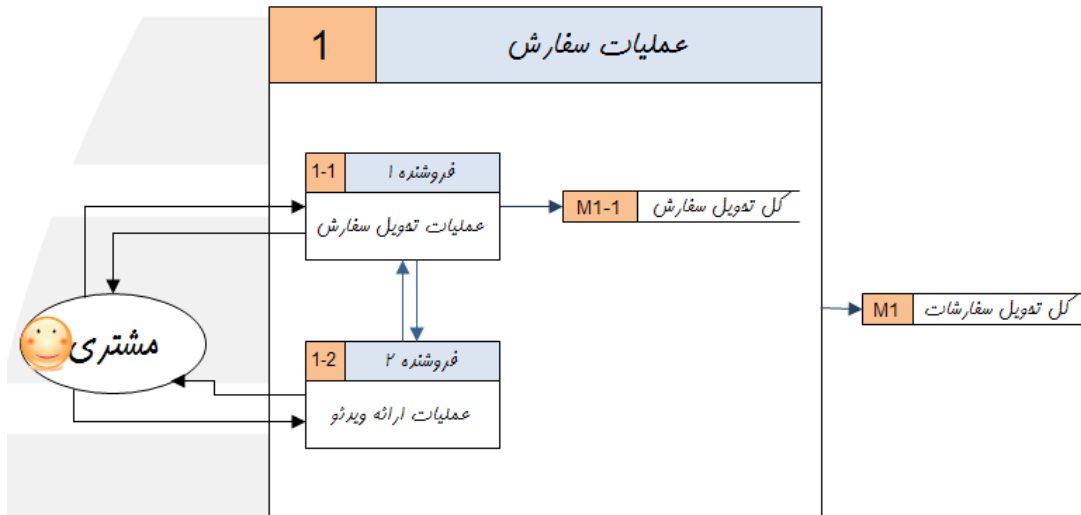
نمودار سطح یک بیمارستان (بخش اورژانس):



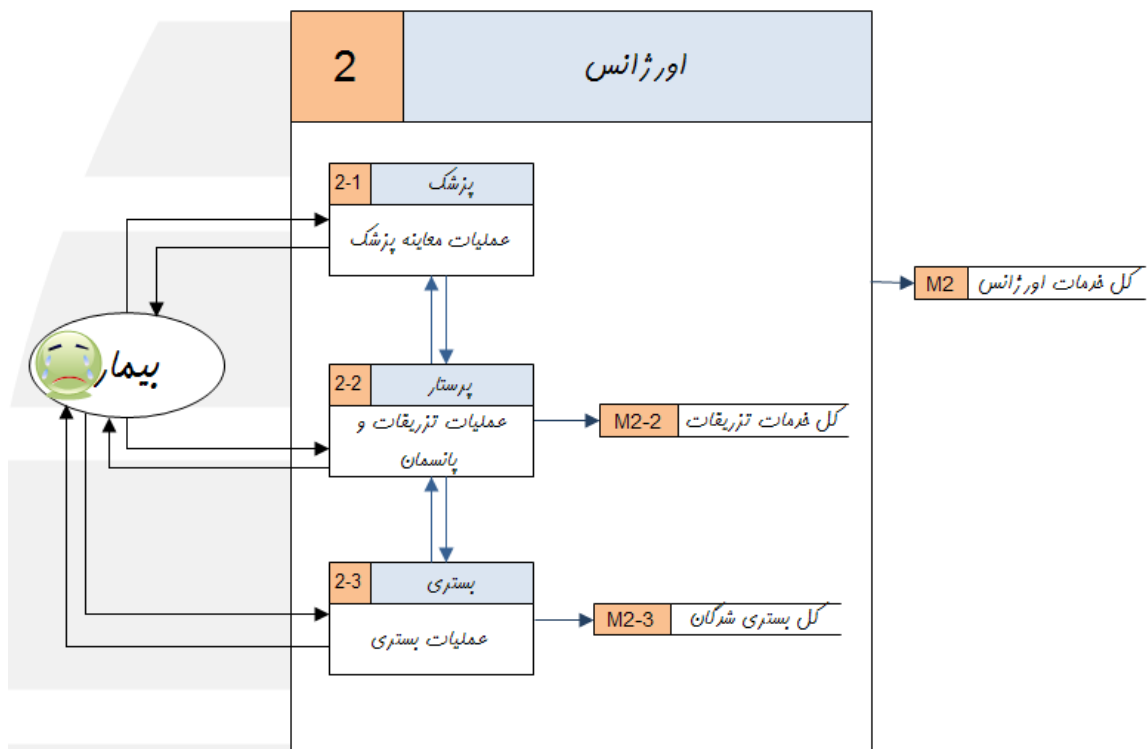
## نمودار سطح دو (DFD) :

سطح بعدی DFD سیستم را با اطلاعات وسیعتر نشان می دهد . به همین صورت نمودار سطح ۳ و ۴ و ... را داریم .

### نمودار سطح دو عملیات سفارش ویدئو :



### نمودار سطح دو سیستم اورژانس :



## نمودار تاریخچه ی زندگی موجودیتها (ELH(Entity Life History):

نمودار جریان داده ها یا DFD مشخص نمی کند که یک سیستم چه موقع شروع و چه موقع تمام می شود و یا به چه ترتیبی فرایندها انجام می شود . برای رفع این مشکل از نمودار ELH استفاده می کنیم.

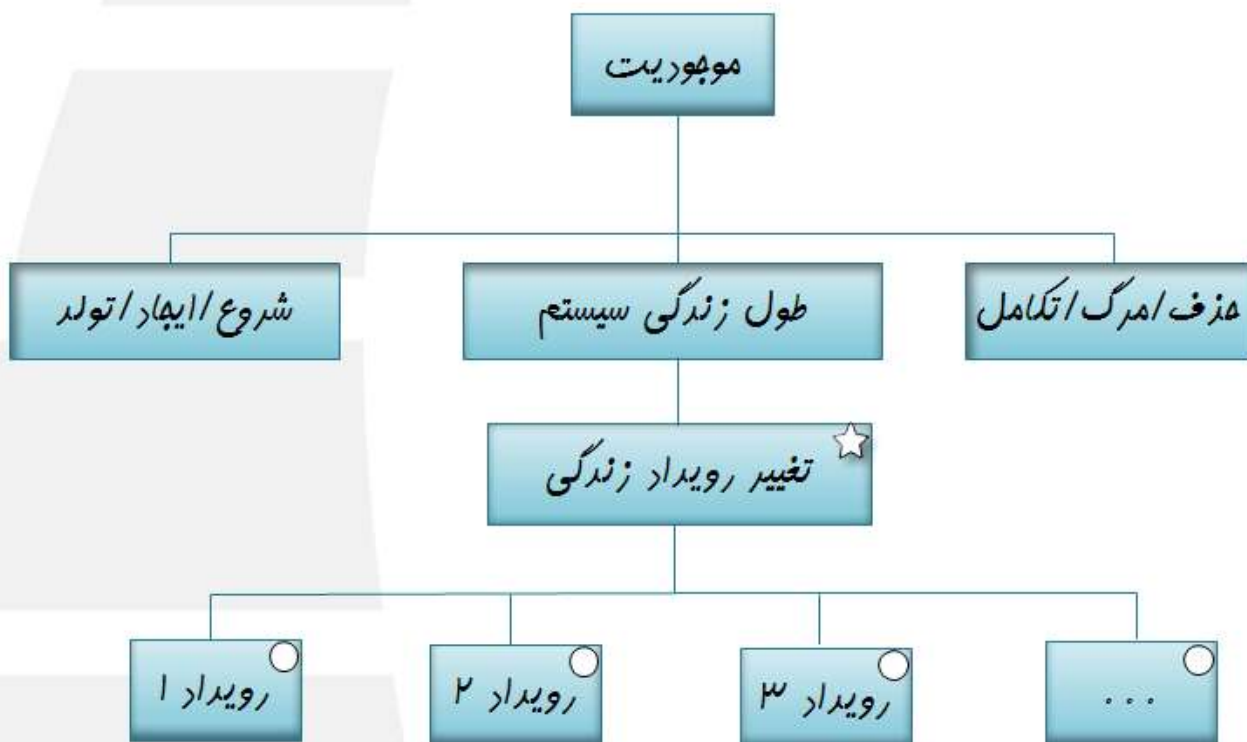
اجزای نمودار ELH عبارتند از :

- توالی
- تکرار
- انتخاب

### توالی:

عبارت است از عملیات پی در پی که به دنبال هم بر حسب زمان انجام می شود . سلسله مراتب از چپ به راست نشان داده می شود. موجودیت موردنظر ابتدا ایجاد شده سپس یک دوره ی زندگی را در سیستم سپری می کند و در انتها از سیستم حذف می شود .

نمودار عمومی ELH به صورت زیر است :



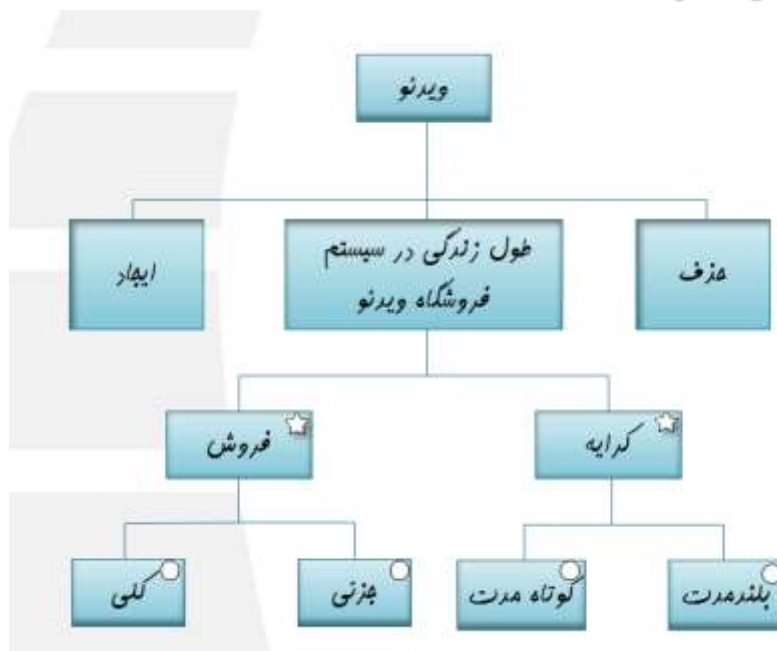
### تکرار:

عبارت است از مجموعه ای از عملیات تکراری که مرتباً تداوم دارد .

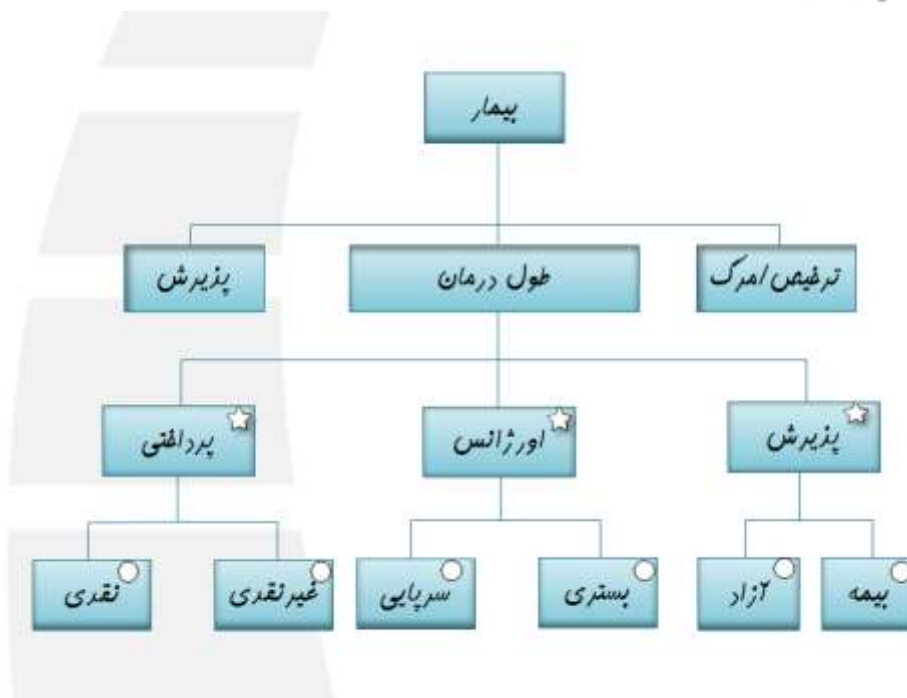
### انتخاب:

انتخاب رویدادها بیانگر تعداد اختیاراتی را مشخص می کند که موجودیت می تواند انجام دهد و فقط یکی از این اختیارات در یک زمان قابل انتخاب است .

مثال - نمودار ELH سیستم فروش ویدئو:



مثال - نمودار ELH موجودیت بیمار:



## سه نظریه ی سیستم های اطلاعاتی :

### ۱-نظریه ی فرایندی :

در این نظریه نمودارهای DFD نشان می دهند که چگونه اطلاعات در گرداگرد سیستم و در اطراف سیستم جریان مینماید.

### ۲- نظریه ی داده ای :

نمودارهای ERD در این نظریه ارتباط داده ها و چگونگی ذخیره ی آنها را نشان می دهد .

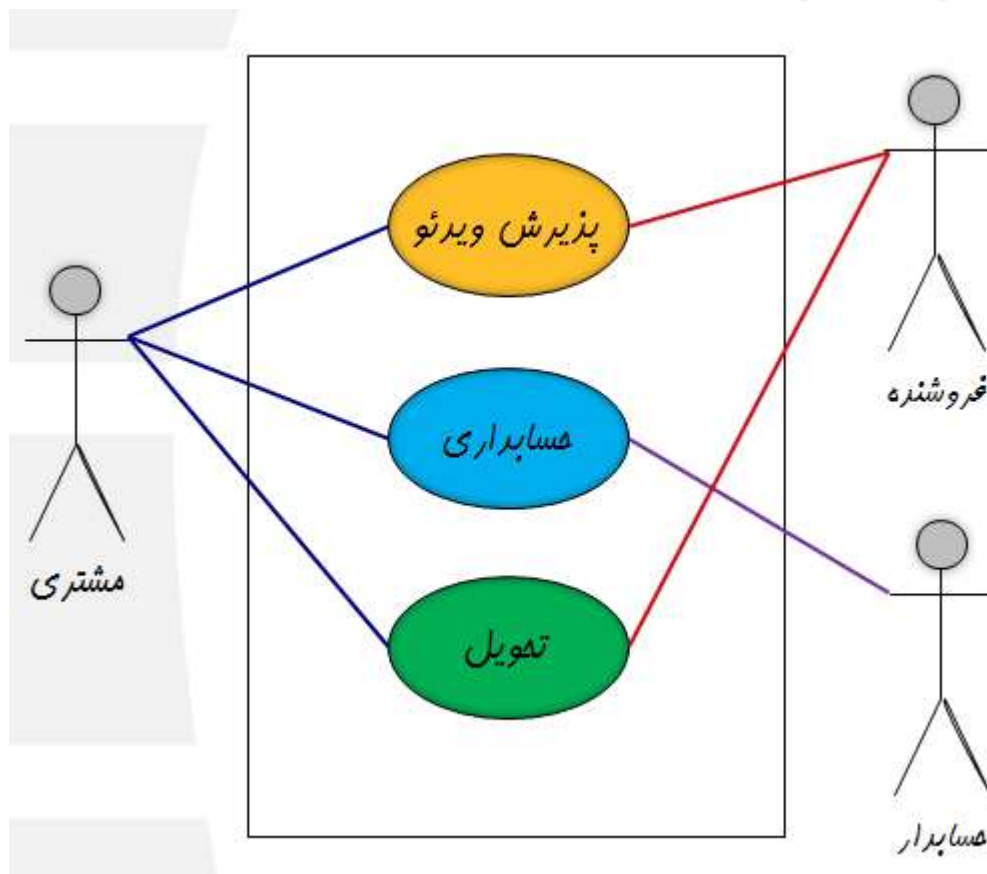
### ۳- نظریه ی دینامیکی :

در این نظریه نمودارهای ELH نشان می دهند که چگونه اطلاعات در سیستم تغییر پیدا می کند.

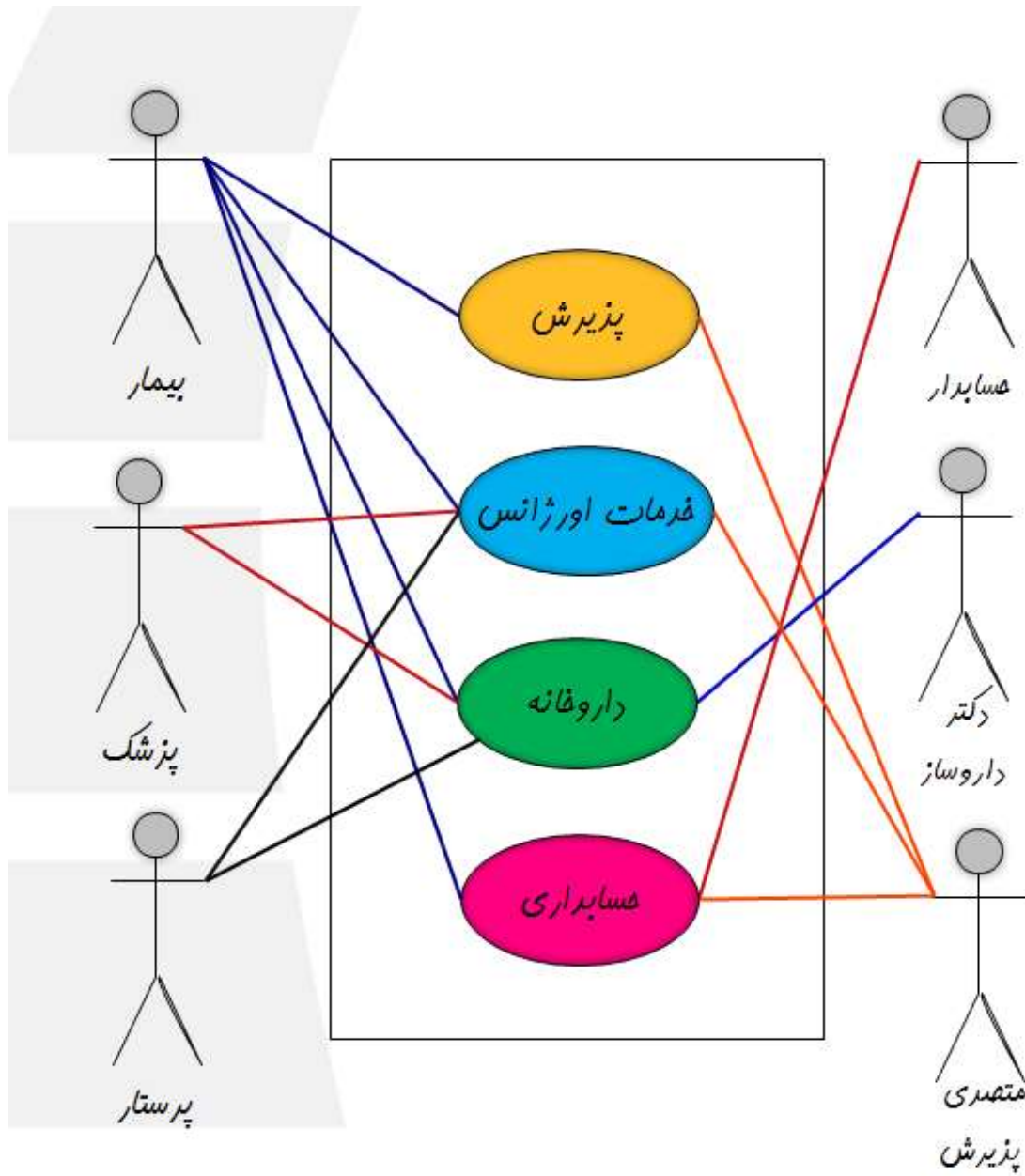
## نمودار وضعیت :

در این نمودار بازیگران و مشتریان بعنوان موجودیت بیرونی DFD می باشند و نشان داده می شود که آنها چه تأثیری بر سیستم دارند.

مثال - نمودار وضعیت فروشگاه ویدئو :



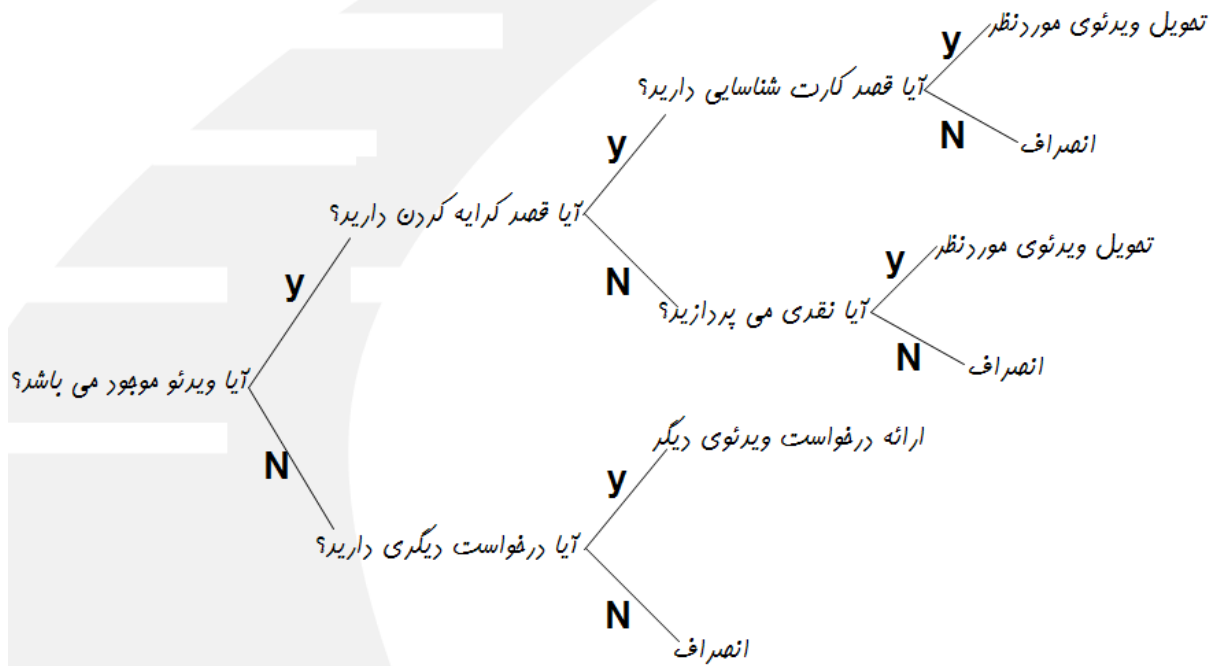
مثال - نمودار وضعیت سیستم اورژانس بیمارستان :



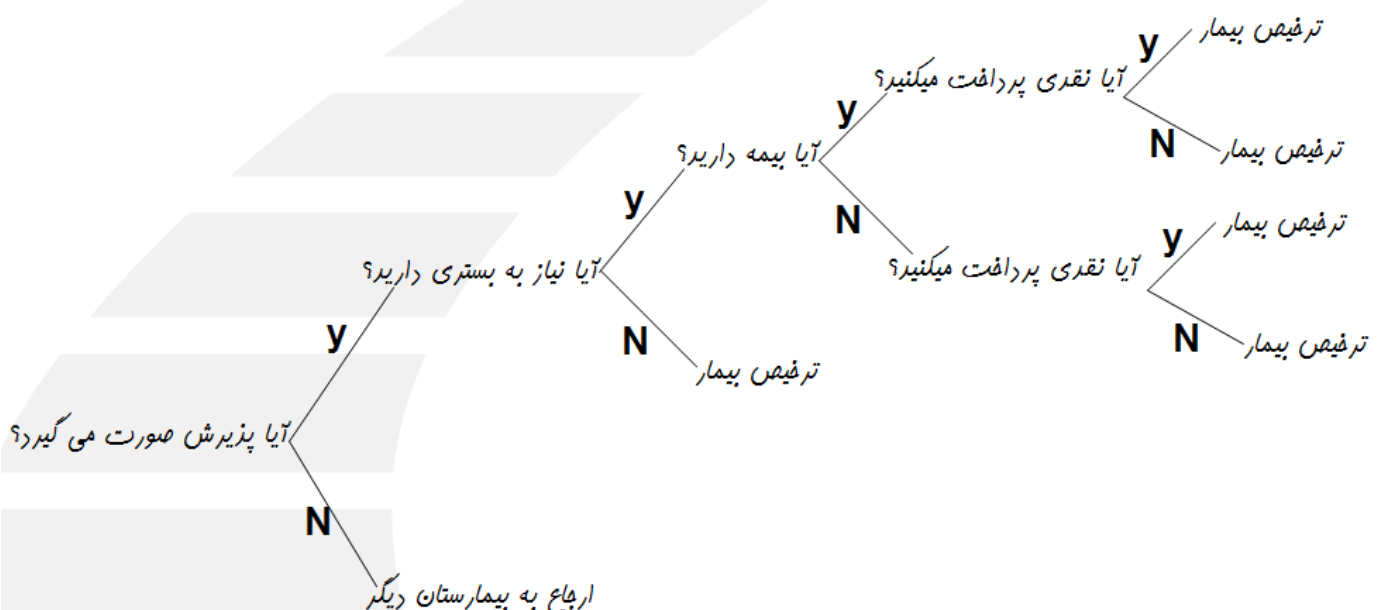
## درخت تصمیم گیری :

یکی از روشهای معروف نمایش شرایط عملیات می باشد. حقایقی هستند که به سادگی قابل فهم باشند .

### مثال - درخت تصمیم گیری فروشگاه ویدئو :



### مثال : درخت تصمیم گیری سیستم اورژانس :





# CATWOE

پروژه تجزیه و تحلیل سیستم نمایندگی فروش اینترنت پر سرعت (ADSL)

Customers مشتریان : مراجعین (ادارات - شرکت ها - اشخاص و ...)

Actors بازیگران : مدیرعامل - حسابدار - مسئول خدمات - پشتیبانی و ...

Transformation تغییر اطلاعات : اطلاعات ثبت نام های ADSL

Weltanschauung معنی دار شدن : ارائه ی اشتراک، خدمات و پشتیبانی (ADSL)

Owner مالکان : ۱۰۰٪ بخش خصوصی (هیئت مؤسس شرکت خدمات اینترنت)

Environment محیط : شرکت ارائه دهنده ی خدمات ADSL2+

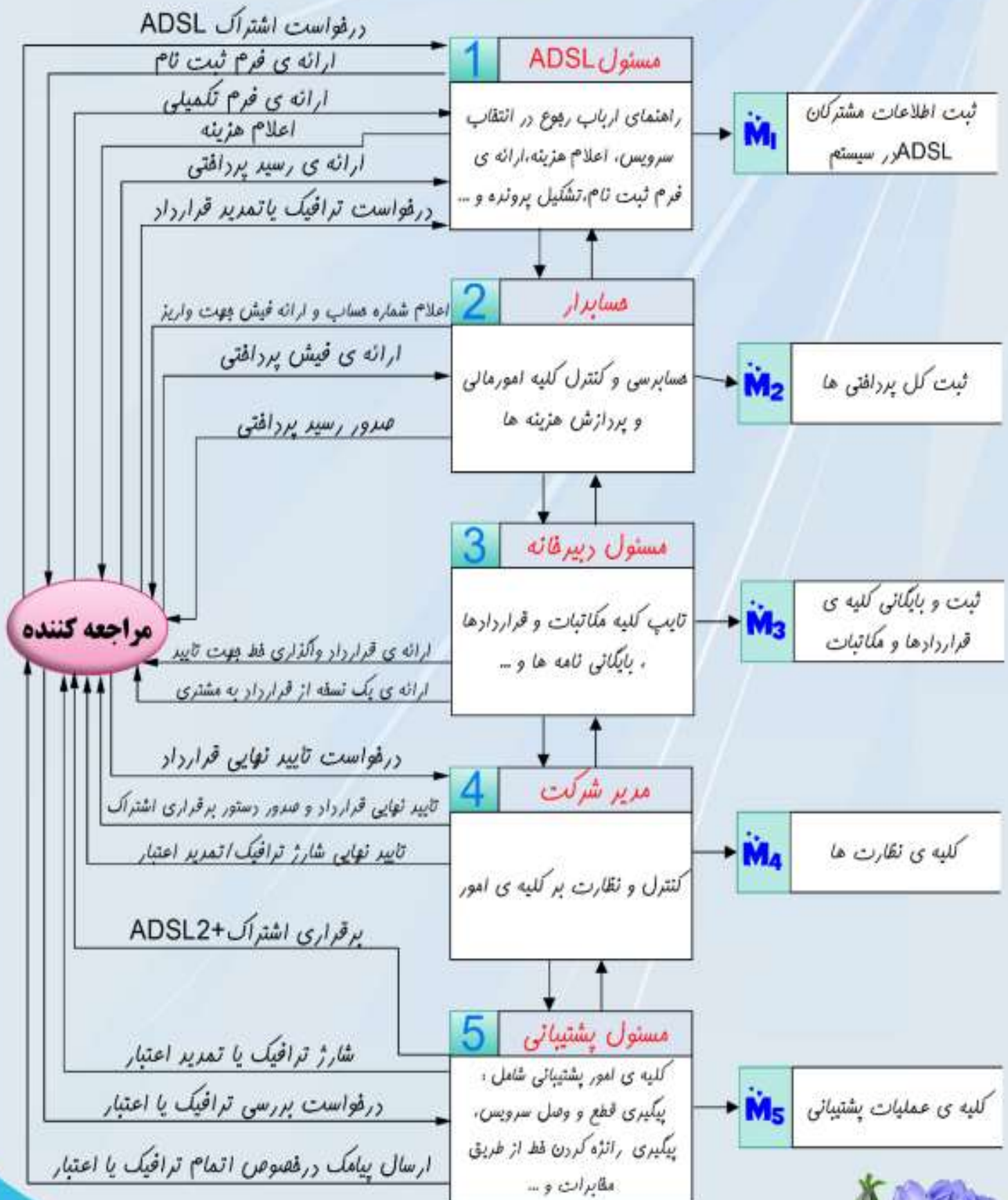




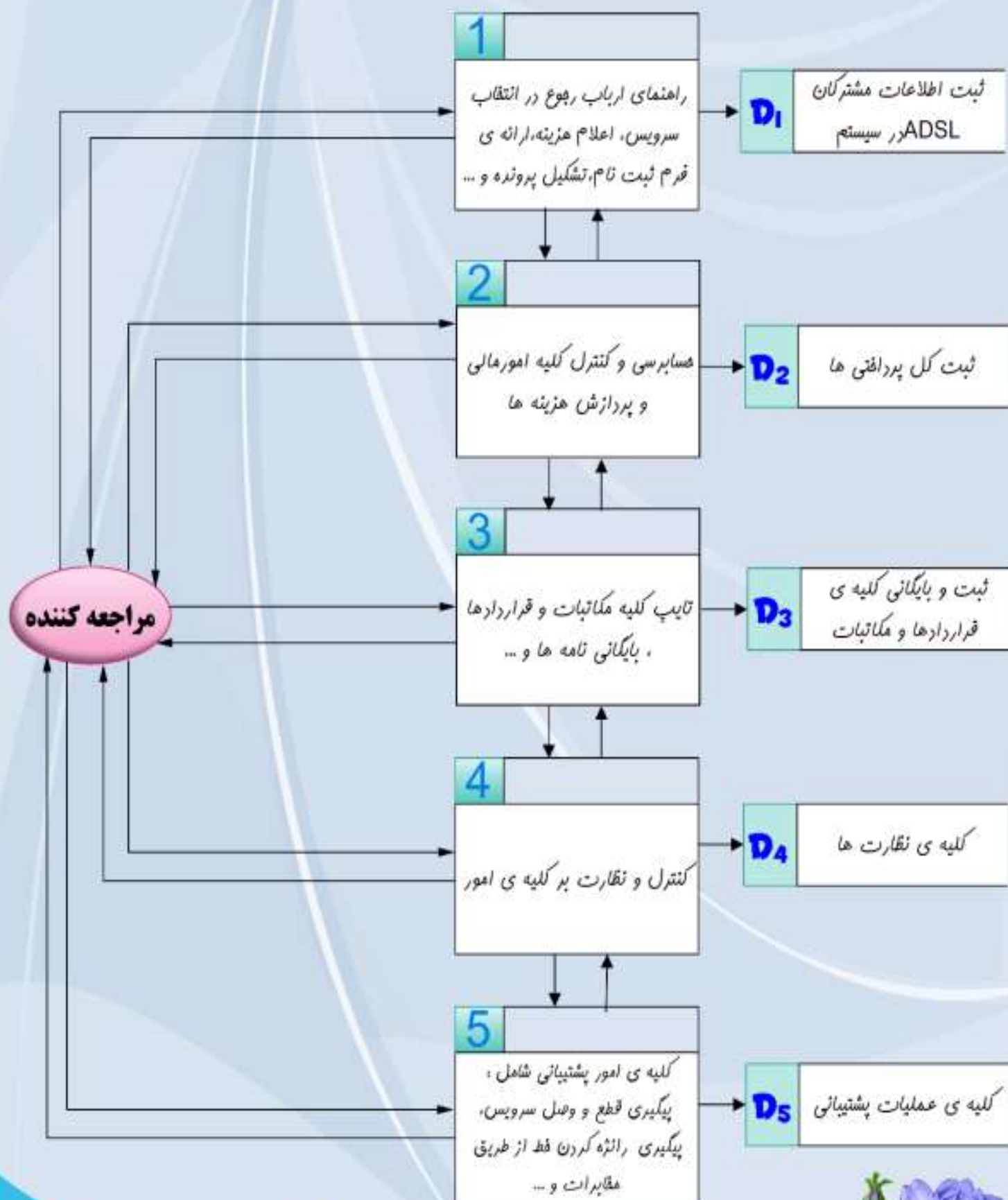
## نمودار سطح صفر (زمینه ای)



# نمودار سطح یک DFD

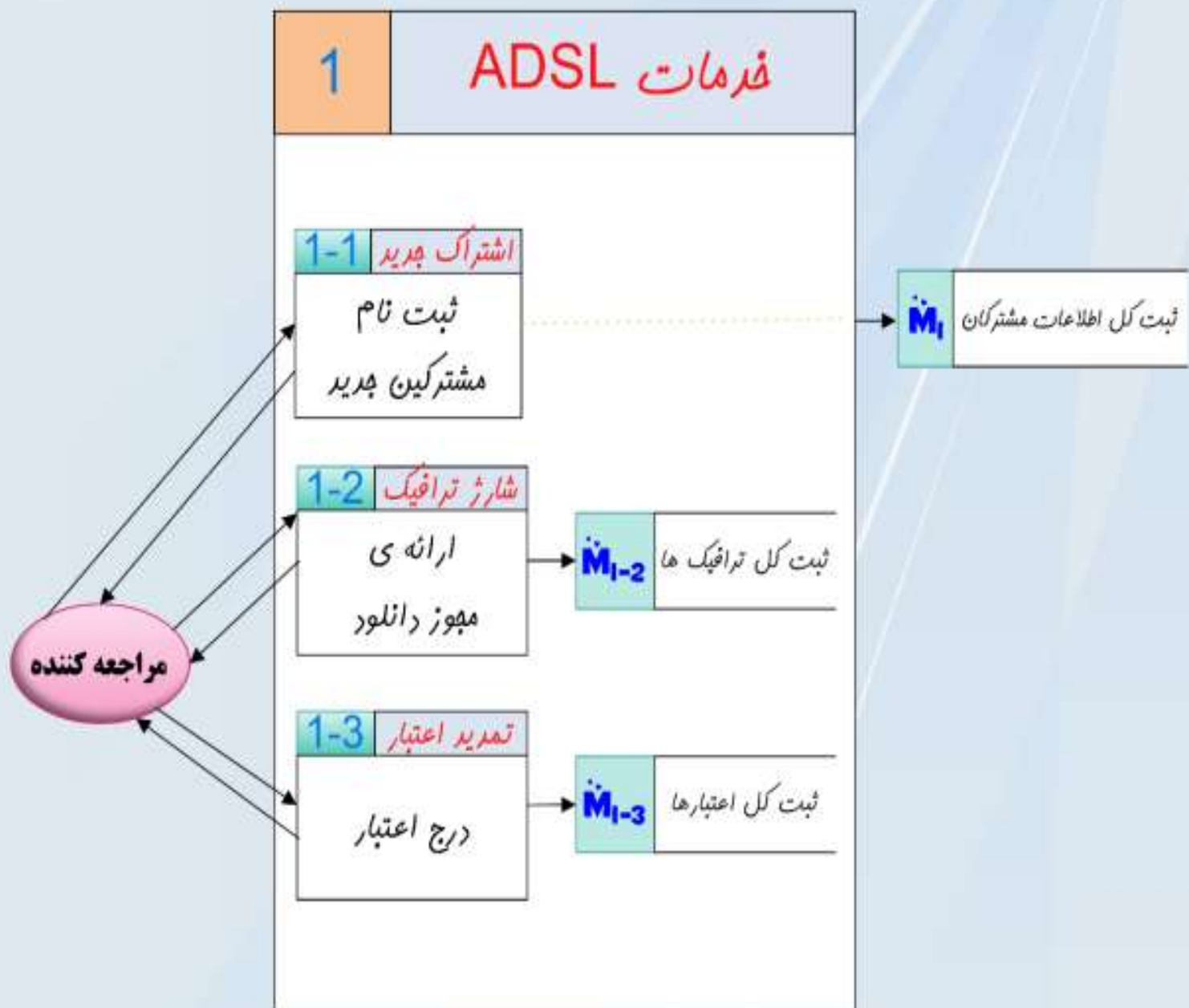


# نمودار منطقی





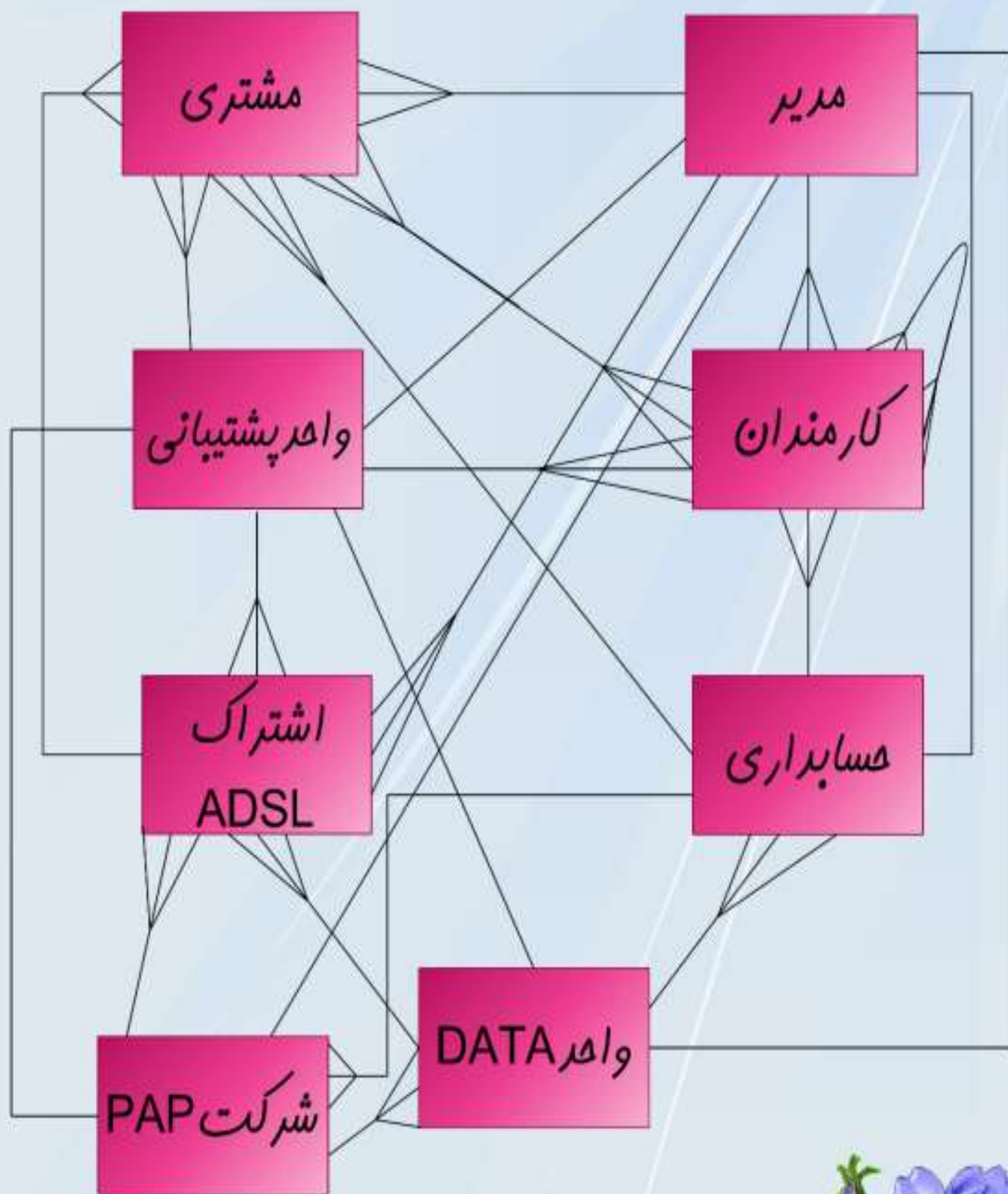
# نمودار سطح دوم DFD



# نمودار ELH

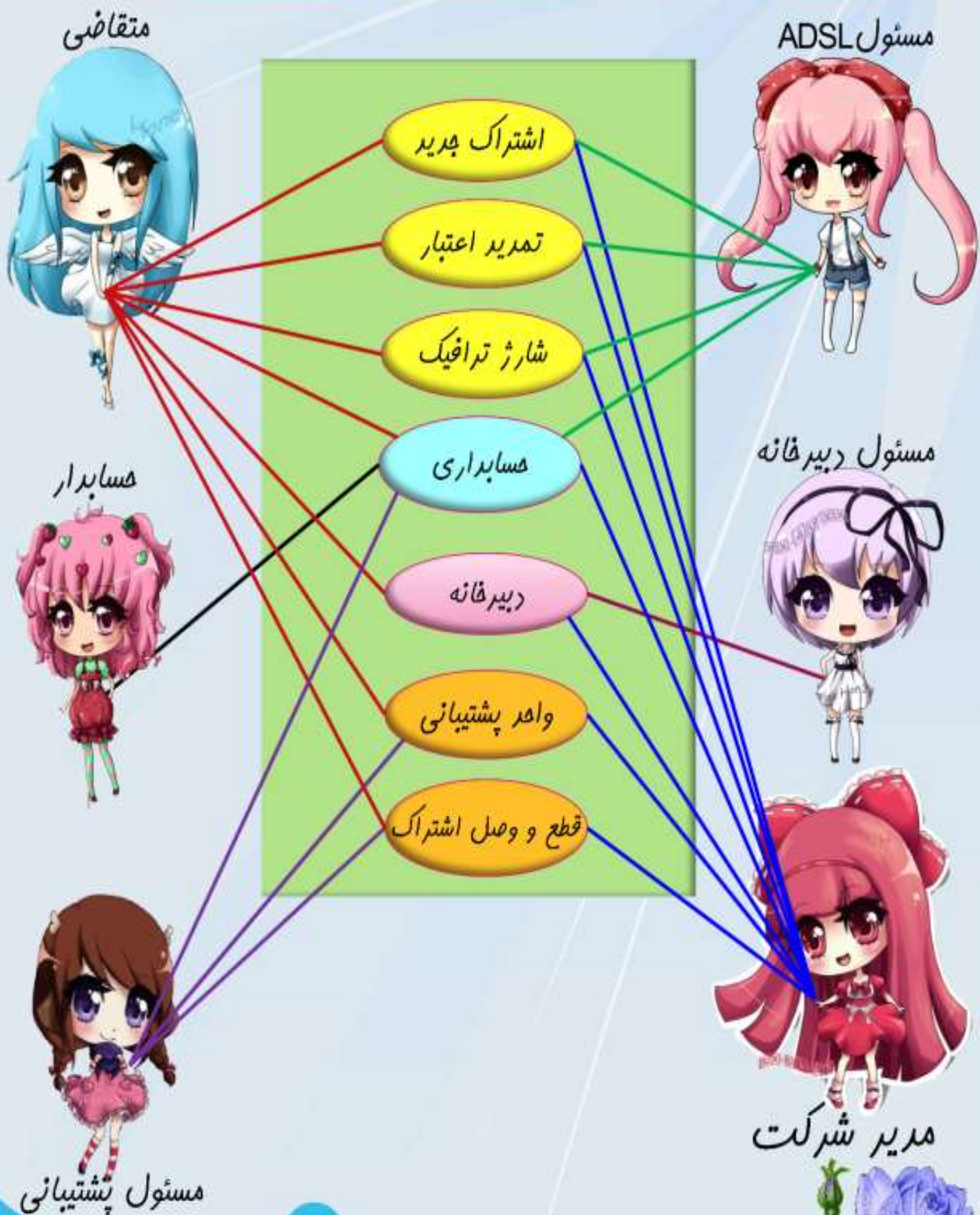


# نمودار ERD





# نمودار وضعیت



## درخت تصمیم گیری

