

"بسم الله الرحمن الرحيم"

جزوه ی درس تجزیه و تحلیل سیستم ها

دانشگاه : شهاب دانش

نام استاد : محمد علی صادقی

نیم سال اول سال ۰۰ - ۹۹

تعریف مهندس: مهندس کسی است که بر اساس دانش متکی بر علم خود به تحلیل مسائل می پردازد، سپس راه حلی قابل انجام و پیاده سازی ارائه می دهد و تولید محصول را مدیریت، درستی و صحت تولید را بررسی و سپس محصول تولید شده را در اختیار کاربران نهایی می گذارد.

تعریف کار مهندسی: کاری است که در آن از اصول علمی استفاده می شود و شامل فرآیندها، ابزارها و روش هاست.

تعریف نرم افزار: مجموعه ای از برنامه های کامپیوتری، روال ها، قوانین، مستندات و داده ها را نرم افزار می گوئیم. (مجموعه ای از آیتم ها و یا اشیاء است که در کنار هم یک پیکربندی را تشکیل می دهند که شامل برنامه های کامپیوتری، مستندات، داده و ... می شود).

تعریف مهندسی نرم افزار: تعریف اول: مهندسی نرم افزار، شاخه ای است از مهندسی، که با بهره گیری از دانش علمی، به ارائه ی راه حل هایی مقرون به صرفه، در قالب دستاوردهای نرم افزاری و به منظور حل مسائل و مشکلات عملی و خدمت به جامعه ی بشری، اقدام می نماید. تعریف دوم: مهندسی نرم افزار یک رشته مهندسی است که تمامی جنبه های تولید نرم افزار را هدف قرار می دهد. مهندسان نرم افزار باید رویکردی سیستمی و سازمان یافته به کار خود داشته باشند و از ابزارها و تکنیک های مناسب، مطابق با مشکلی که باید حل شود، محدودیت های تولید و منابع در دسترس استفاده نمایند. تعریف سوم: مهندسی نرم افزار یعنی استفاده از اصول مهندسی بجا و مناسب برای تولید و ارائه محصول نرم افزاری با کیفیت که قابل اطمینان و با صرفه بوده و بر روی ماشین های واقعی به طور کارآمدی عمل کند. مهندسی نرم افزار یک روش سیستماتیک، منظم و دقیق برای ساخت و ارائه محصول نرم افزاری با کیفیت است. مهندسی نرم افزار اغلب شامل فرایند خطی تحلیل، طراحی، پیاده سازی و پشتیبانی است؛ که با به کارگیری روش های فنی و علمی از علوم مهندسی موجب تولید نرم افزاری با کیفیت مطلوب می شود.

تعریف چهارم: به کلیه روشها و راه حل هایی که با بکارگیری ابزار و تکنیکهای مختلف، باعث تولید یا ارائه نرم افزاری، با قابلیت های زیر می شوند مهندسی نرم افزار گفته می شود: کارایی و کیفیت بالا - منطبق بر نیازهای مشتری - سهولت در کارکردن - کمترین هزینه - سازگاری با انواع سیستم ها - قابلیت نگهداری و توسعه

وظیفه ی علم مهندسی نرم افزار: ایجاد روندی سیستماتیک، منظم و قابل اندازه گیری برای تولید و نگهداری نرم افزار را وظیفه ی علم مهندسی نرم افزار می دانیم.

معیار مهم مهندسی نرم افزار: ۱. زمان ۲. هزینه ۳. کیفیت نرم افزاری که می خواهیم تولید کنیم.

خصوصیات مهندسی نرم افزار: ۱- اقتصاد تمامی کشورهای توسعه یافته وابسته به نرم افزار است ۲- سیستم ها به طور فزاینده ای توسط نرم افزارها کنترل می شوند ۳- مهندسی نرم افزار شامل تئوری ها، متدها و ابزارهایی برای تولید حرفه ای نرم افزار ۴- هزینه مهندسی نرم افزار نشان دهنده یک بخش قابل توجه ای در تولید خالص ملی کلیه کشورهای توسعه یافته می باشد. (**نکته:** مهندسی نرم افزار هم ارز با برنامه نویسی نیست، بلکه برنامه نویسی فقط جزئی از مهندسی نرم افزار است.)

تعریف مهندسی نرم افزار از دیدگاه Pressman : به مجموعه ای از فن آوری ها، روش ها و ابزارهای مبتنی بر اصول مهندسی که در توسعه ی نرم افزار استفاده می شود، مهندسی نرم افزار گفته می شود.

تعریف مهندسی نرم افزار از دیدگاه Bauer : مهندسی نرم افزار عبارت از توسعه و استقرار نرم افزار با استفاده از اصول دقیق مهندسی برای دستیابی به نرم افزاری با صرفه اقتصادی است که دو شرط قابل اعتماد و کارا را دارد.

تعریف مهندسی نرم افزار از دیدگاه IEEE : بکارگیری روشی سیستماتیک ، اصولی و قابل بررسی برای توسعه ، اجراء و پشتیبانی نرم افزار می باشد.

ویژگی های نرم افزار مبتنی بر علم مهندسی : ۱- قابل نگهداری باشد. ۲- قابل اعتماد از دید کاربر باشد. ۳- کارا (استفاده مطلوب و بهینه از منابع سخت، نرم افزار و محیط) باشد. ۴- ارتباط ساده با کاربر (حداقل ورود اطلاعات، حداکثر خروجی) داشته باشد.

تعریف مهندسی نرم افزار از دیدگاه Sommerville : مهندسی نرم افزار در ارتباط با توسعه ی سیستم های نرم افزاری به شکل فعالیت گروهی است و از قواعد مهندسی استفاده می کند و شامل جنبه های فنی و غیرفنی است.

هدف مهندسی نرم افزار : ۱= هدف مهندسی نرم افزار آموزش، پژوهش و اعمال روش ها و فرآیندهای مهندسی در تولید نرم افزار است (PQCT) : ۱- افزایش بهره وری تیم توسعه (Productivity)

۲- افزایش کیفیت محصول (Quality)

۳- کاهش هزینه تولید (Cost)

۴- کاهش زمان تحویل (Time to Market).

۲= تولید سیستم به گونه ای که دوباره کاری و تغییر حداقل شود. ۳= ارائه روشی جامع جهت تولید یا ارائه نرم افزاری با کیفیت و مبتنی بر نیازهای واقعی متقاضیان میباشد.

دلایل استفاده از مهندسی نرم افزار : ۱- همه جا است ۲- تولید نرم افزار مستلزم کار گروهی است ۳- برای کار تیمی نیاز به موارد مختلف هست ۴- پس تولید نرم افزار باید به صورت مهندسی باشد.

ویژگی های نرم افزار : ۱- نرم افزار را مثل یک محصول کارخانه ای نمی بینیم زیرا در ورای نرم افزار یک منطق وجود دارد و محصولی منطقی است و محصول فیزیکی نیست ۲- نرم افزار دور انداختنی نیست . با توجه نگرش جدید، منحنی عمر مفید نرم افزار با سرعت کمی بالا برود . ۳- نرم افزار ها بیشتر بر اساس نیاز مشتریان ساخته می شود .

مشکلات نرم افزار : ۱- پیشرفت سخت افزار آنچنان سریع است که نرم افزار امکان رقابت با آن را ندارد. ۲- عدم تطابق نرم افزار با نیاز واقعی مشتری و بازار ۳- عمومی شدن کاربرد کامپیوتر در جامعه و عدم امکان تعویض نرم افزار به سرعت ۴- فشار عمومی و زیاد برای تولید نرم افزاری قابل اعتماد و دارای کیفیت مناسب ۵- قدرت پشتیبانی و نگهداری نرم افزارهای موجود بسیار محدود است.

مسائل و مشکلات نرم افزار در دنیای کنونی: ۱. قابلیت اطمینان نرم افزار : بدان معنا که نرم افزار به درستی اجرا شود. ۲.

هزینه ی نرم افزار : هدف: کاهش هزینه ی خرید نرم افزار با حفظ کیفیت. ۳. اعمال تغییرات و دوباره کاری

انواع نرم افزار: ۱. چکشخوار (قابل اعمال تغییرات) ۲. غیر چکشخوار (غیرقابل تغییر)

چگونگی در نظر گرفتن تولید نرم افزار به صورت یک روند : تولید نرم افزار از مجموعه ای از فعالیتها ساخته می شود. در

تولید یک نرم افزار دارای محدودیتهایی هستیم : ۱. زمان ۲. هزینه ۳. محدودیتهای تکنیکی در تولید نرم افزار هدف ساخت یک نرم افزار با کیفیت بالا و هزینه کم می باشد. تولید نرم افزار یک روال و یا روندی است که از مجموعه ای از کارها تشکیل شده است.

ویژگی های روال های تولید نرم افزار: ۱. قابل پیش بینی بودن : (۱. کیفیت ۲. هزینه ۳. زمان ۴. پیش بینی ارتباط بین

فعاليتها (اولویت در ترتیب انجام مراحل)). ۲. هر روال یا روند تولید باید قابل تست باشد. ۳. امکان روال های تولید جهت حذف سریع خطاها و جلوگیری از به وجود آمدن خطاها ۴. اصلاح روال تولید. ۵. هر روال از یک سری فازهای متنوع ساخته شده است. ۶. هر فاز با یک خروجی مشخص خاتمه پیدا می کند. (وقتی فاز تمام شد، نتیجه ی آن یک محصول است. مثل : گزارش ، برنامه ، ...) ۷. فازهای تولید نرم افزار در روال های مختلف با ترتیب و توالی مختلف انجام می شود.

ویژگی های یک نرم افزار به صورت یک محصول (Software as a product product): ۱. نرم افزار یک محصول مهندسی

است و با اصول مهندسی باید تولید شود. ۲. نرم افزار یک محصول قابل تغییر یا چکشخوار است. ۳. نرم افزار به دلیل اینکه محصولی فیزیکی نیست ، خراب یا مستهلک نمی شود. اما در عمل به دلیل اعمال تغییرات مداوم شاید دیگر قابل استفاده نبوده و می بایست نرم افزار دیگری جای آن را بگیرد. ۴. نرم افزار برخلاف بسیاری از محصولات مهندسی دیگر ، غالباً به صورت سفارشی ساخته می شود و از اجزای آماده در آن کمتر استفاده می شود که یکی از اهداف مهندسی نرم افزار ، افزایش استفاده از قطعات نرم افزاری آماده است.

دلایل استفاده از مهندسی نرم افزار در پروژه های مهندسی : مهندسی نرم افزار نقش اساسی در بالا بردن کیفیت نرم افزار و کاهش هزینه ها دارد.

نقش مهندسی نرم افزار در پروژه های مهندسی : ۱. کاهش وابستگی به افراد متخصص به صورت خاص ۲. بالا بردن

کیفیت ارتباطات تیمی ۳. تخمین مناسب شامل تخمین زمان و هزینه ۴. مدیریت تغییرات ۵. کنترل زمان انجام پروژه ها ۶. برقراری ارتباط و درک متقابل از نرم افزار بین تولید کنندگان، کاربران و مدیران ۷. انجام و ارائه ی آموزش های مناسب ۸. انجام پیش بینی های لازم جهت مواجهه با افزایش توقع کاربران

اهداف مهندسی نرم افزار : ۱. بالا بردن کیفیت : ۱- تطبیق نرم افزار با نیازمندیها ۲- جوابگویی نیازهای کاربران ۳- فارغ از خطا بودن یا کم خطا بودن و کارایی بالای نرم افزار. ۲. قابل دسترسی باشد. ۳. اهداف متناقض باید بصورت تعادل درآیند. مهندس نرم افزار فردی است که قواعد و اصول علم مهندسی نرم افزار را در روند ایجاد یک نرم افزار یا در حین تولید یک پروژه ی نرم افزار ی استفاده می کند.

* نرم افزار با کیفیت مناسب نرم افزاری است که هم نیازهای صریح و هم نیازهای ذهنی ما را رفع نماید. هر چقدر نرم افزار از منابع کمتری استفاده کند، کارایی بالاتری دارد.*

ویژگی های یک مهندس نرم افزار ایده آل: ۱. یک برنامه نویس خوب باشد. ۲. با روش های مختلف طراحی آشنایی داشته باشد. ۳. امکان ترجمه و تبدیل نیازهای کاربران ۴. قابلیت ارتباط با طیف مختلف کاربران و مدیران ۵. دارا بودن قابلیت بالای مدیریتی

چرا روال تولید به صورت فازبندی یا مرحله بندی شده است؟ ۱. هر فاز یا مرحله نگرشی متفاوت به نرم افزار ارائه می دهد. ۲. شکستن یک مسئله ی بزرگ به مسائل کوچکتر باعث آسان تر شدن حل مسئله می شود. ۳. ارتقاء کیفیت نرم افزار با فازبندی به دلیل کنترل کیفیت در حین تولید آن انجام می شود. ۴. فازبندی شدن تولید نرم افزار باعث کاهش هزینه ی تولید می شود. با بالا رفتن کیفیت؛ هزینه ی نگهداری کاهش - قابل بازبینی شدن اشکالات هر مرحله یا هر فاز - کار کردن افراد متخصص به آن فاز در هر فاز و با کیفیت تر انجام شدن کار.

فازها یا مراحل تولید نرم افزار: ۱. تعیین یا مشخص کردن نیازمندی ها و ارائه ی آن در ی فاز قابل فهم. ۲. تعیین اینکه کار چطور باید انجام شود تا کیفیت بالا رود. برای اینکه کدام راه، راه مناسب تری برای انجام نیازمندی ها ست. ۳. ارائه ی راهکاری برای پیاده سازی برنامه ای که قبلاً نیازهای آن را شناخته ایم. اینکه نیازمند چه اجزایی هستیم و هر جزء باید در کجا قرار بگیرد و چگونه به اجزای دیگر متصل شود. (به عنوان مثال در یک طراحی به گونه ی ساخت یافته) مثل زمانی که با زبان C برنامه ای را می نویسیم. (باید مشخص شود چه ماژول ها یا فانکشن هایی دارد و فانکشن ها چگونه در کنار هم قرار می گیرند). ۴. Implementation. (پیاده سازی یا همان کد نویسی)

۵. Testing. تست کد نوشته شده: ۱: unit - یا واحد: تست کوچکترین عناصر برنامه، تست ماژول ها یا function ها

۲: system - قسمت های مختلف سیستم را به هم متصل کرده و تست می کنیم. به این نوع تست، تست آلفا گفته می شود. تستی که در گروه تولید کننده ی نرم افزار به مجتمع برای کل برنامه انجام میشود.

۳: acceptance - تست تجاری یا تست بتا. تست قابل قبول بودن پس از تست شرکت یا system به برخی از مشتریان حرفه ای برای تست داده می شود.

۶. conversion یا انتقال: تحویل به مشتری و بردن محصول به محیط کاری واقعی و راه اندازی آن در محیط واقعی. ۷. مرحله ی نگهداری یا پشتیبانی: maintenance اگر نرم افزار مشکلاتی داشته باشد، مجدداً به سازنده مراجعه شده و بررسی می شود.

شرح روش های conversion یا انتقال مرحله ی تولید نرم افزار: ۱ - استراتژی موازی: parallel strategy کار کردن همزمان نرم افزار با نسخه های قبلی برای سیستم هایی که اهمیت بالایی دارند و در صورت ایجاد مشکل، مشکل بزرگ و غیرقابل جبران باشد. ۲ - قطع ناگهانی: Direct Cutover اینکه تصمیم گرفته شود نرم افزار قبلی از امروز نار گذاشته شود و نرم افزار جدید مورد استفاده

قرار گیرد. (پس از اطمینان صحت عملکرد). ۳ - راه اندازی نمونه ای : Pilot Study نرم افزار را مثلاً برای دو کاربر خاص نوشته و برای تست مورد استفاده قرار می دهیم. ۴ - راه اندازی فاز بندی شده : Phased محصول را مرحله به مرحله راه اندازی می کنیم.

مهندسی نرم افزار شامل چه کارهایی ست ؟ ۱. بررسی مسئله از دیدگاه های متفاوت. ۲. تجزیه و تحلیل مسئله با اصول مهندسی نرم افزار. ۳. انتخاب بهترین راه حل برای انجام پروژه های نرم افزاری. ۴. پیاده سازی نرم افزاری. ۵. مدیریت پروژه و نظارت کامل بر مراحل اجرای کار پروژه (بایستی بالا بردن کیفیت - به صرفه بودن پروژه و تحویل در وقت مقرر در نظر گرفته شود). ۶. تست پروژه برای اطمینان از نحوه صحیح عملکرد پروژه. ۷. پشتیبانی پروژه

فعالیت های فرایند مهندسی نرم افزار : ۱. تعریف (بر چستی تاکید دارد)

- مشخص کردن هدف ، نیازهای موجود و رفتارهای مورد انتظار سیستم، جمع آوری اطلاعات، برنامه ریزی، تجزیه و تحلیل مسئله، شرح نیازمندی ها ، تهیه مستندات

۲. توسعه (بر چگونگی تاکید دارد)

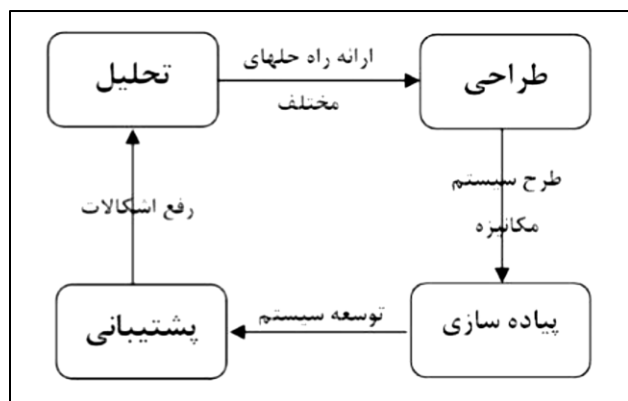
- تولید دیاگرام های DFD و ERD ، سازماندهی داده ها ، تکمیل ارتباطات، چگونگی انجام عمل پردازش ها ، طراحی نرم افزار، کدنویسی، عملیات تست

۳. پشتیبانی (بر تغییرات تاکید دارد)

- رفع اشکالات موجود در نرم افزار، اصلاح خطاهای موجود توسعه و در صورت نیاز تولید بخش های جدید.

تعریف چرخه حیات تولید نرم افزار : مراحل تحلیل، طراحی، پیاده سازی و پشتیبانی نرم افزار را چرخه حیات تولید نرم افزار می نامند.

تعریف مراحل چرخه حیات تولید نرم افزار : ۱. تحلیل: در این مرحله، نیاز کاربران مشخص شده ، راه حل های مختلف ارائه و بهترین راه حل جهت رسیدن به سیستم جدید انتخاب می گردد. ۲. طراحی: طرح سیستم مکانیزه با تعریف نیازمندی های مورد نظر جهت خروجی / ورودی، ذخیره سازی و پردازش و کنترل های لازم ارائه می گردد. ۳. پیاده سازی: کدنویسی، تست و اجرا در این مرحله صورت می گیرد. ۴. پشتیبانی : بررسی مجدد، رفع اشکالات و ترمیم و توسعه سیستم از عملیات این مرحله می باشند.



شکل نشان دهنده مراحل چرخه حیات تولید نرم افزار :

روال های اصلی تولید نرم افزار : ۱. مدیریت پروژه : جزء کارهای اصلی نیست ولی پروژه حتما باید مدیریت شود. ۲. روال مدیریت تغییرات : مدیریت پیکربندی هدف: ایجاد یک روال که تغییرات به صورت منظم و مدون باشد. ۳. روال مدیریت خود روند تولید: خود روندهای تولید نیز ممکن است در سیر زمان نیازمند تغییر باشند و باید هر جای آن را که ایراد داشت عوض یا اصلاح کنیم.

*** هر نرم افزار یک خط تولید اصلی و ۳ خط تولید فرعی دارد.***

روال های جانبی تولید نرم افزار : مدل تجاری Business Models : هر نرم افزار دارای یک سری قواعد و اصول است. به قواعد و قوانینی که سیستم در حالت فعلی دارد، مدل تجاری می گویند.

Environment: نیازمندیهای محیطی جهت اجرا یا پیاده سازی نرم افزار تولید شده ، مثل نیازمندی های سخت افزاری و تعیین کاربرانی که قرار است با آنها کار کند.

روال های تولید نرم افزار Process Models :

۱. روال های تولید خطی : liner process model وقتی تغییری انجام دادیم و طراحی کردیم دیگر قادر به بازگشت به مرحله ی قبل و تغییر در آن نیستیم.

۱.۲ prototyping

۱.۱ روال آبشاری waterfall model

۲. روال های تولید آزمایشی یا تکراری

۲.۱ روال افزایشی

۲.۲ روال مارپیچی

۲.۳ روال معرفی گرا

روال های تولید یک نرم افزار سبک : روال هایی که در حین تولید مستندات زیادی تولید نمی کنند مثل XP

روال های تولید یک نرم افزار سنگین : روال هایی که در حین تولیدشان حجم زیادی از مستندات تولید می شود .

در روال های تولید نرم افزار به صورت خطی دو روال نقش عمده دارند:

۱. روال خطی (Sequential) : ۱.۱ روال آبشاری waterfall

۱.۲ مدل اجرایی prototype

۲. آزمایشی یا چرخشی Iterative : فرایند تولید یک سویه یا خطی نیست بلکه یک فرایند چرخشی است.

مواردی که در الگوی انتخاب روال تولید می تواند مؤثر باشد : ۱. سبک و پیچیدگی پروژه ۲. تجربه ی تیم پروژه از لحاظ قابلیت های فنی و آشنایی با حیطه ی مورد نظر ۳. زمان و هزینه ی در اختیار

اولین روال تولید (روال آبشاری) : ۱. شناخت وضع موجود و نیازمندیها ۲. طراحی ۳. مرحله ی برنامه سازی

اشکالات روال آبشاری: ۱. در صورت وجود ایراد، قادر به بازگشت و اصلاح نیستیم. ۲. اگر نیازمندیها در محیط تغییر بکنند، قادر به تغییر و اصلاح تغییرات نیستیم. تحلیل شده، به فاز طراحی رفته، ولی نیازمندیها عوض شده یا بیشتر شده اند. ۳. کاربر تا انتهای کار، نسخه ای اجرایی نمی بیند. کاربر برای دیدن یک کد باید تا انتهای کار منتظر بماند چون تا برنامه تست نشود، تحویل کاربر نمی شود. Prototype، اصلاح شده ی waterfall است. در prototype وسط نسخه ای اجرایی برای تست تحویل کاربر می شود و طبق نیازهای کاربر، بقیه ی کار انجام می شود.

روال چرخشی یا iterative: ۱. افزایشی incremental: روش خوبی است. مناسب پروژه های سبک است. ۲. مارپیچی spiral: روال سنگین ۳. مؤلفه گرا component base ۴. RUP روال سنگین ۵. XP روال سبک

مزایای روش های تولید چرخشی: ۱. امکان اعمال تغییرات در نیازمندیها ۲. اتصال قطعات مختلف نرم افزار به صورت ۳. کاهش ریسک ۴. امکان یادگیری و آموزش نرم افزار به صورت مرحله ای ۵. بالا بردن کیفیت محصول

تعریف مدل مارپیچی (Spiral): در مدل مارپیچی فرایند تولید نرم افزار به تعدادی ناحیه تقسیم می شود که این تعداد و شرح وظایفی که در هر ناحیه از آنها انجام می شود، به عهده ی مدیر پروژه می باشد. اما معمولاً نواحی بین ۳ تا ۶ ناحیه تعریف می شوند. در چرخه های درونی، معمولاً تمرکز بیشتر بر روی شناخت و تحلیل نیازمندیهاست و در چرخه های بیرونی تمرکز بیشتر بر روی برنامه سازی و تست است.

تعریف مدل مؤلفه گرا یا Component base: مدل مؤلفه گرا یک تکه نرم افزار تولید شده ی قابل اجراست. در این تفکر تأیید بر استفاده از component های از قبل آماده است. اگر برای قسمتی از نرم افزار، component آماده هستین، چه نوشته شده توسط خودمان و چه در بازار نرم افزار پیدا نشد، اقدام به تولید component جدید می کنیم. اما با این تفکر که آن component به صورت جامع و کاملی تولید شود که برای پروژه های بعدی نیز قابل استفاده باشد.

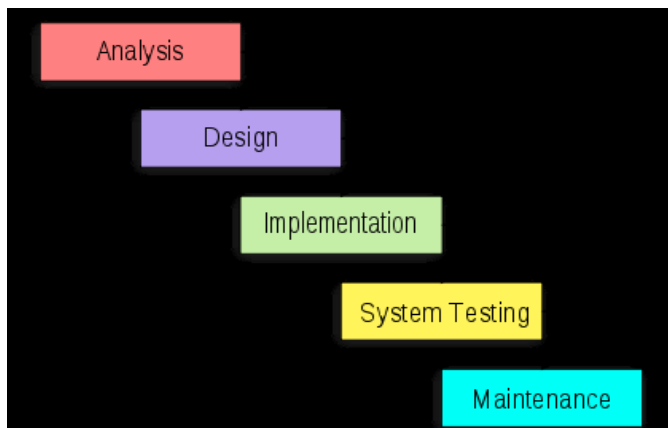
منظور از سفارشی کردن روند تولید: تعیین کلیدی نکات، شامل ساختار مستندات، نحوه ی تحلیل، نحوه ی طراحی و .. می باشد.

نکات سفارشی کردن روند process tailoring: - روال های تولید قالب کلی روند انجام کار را برای ما مشخص می کند. اما انتخاب جزئیاتی از قبیل ساختار مستندات، نحوه ی طراحی، نحوه ی برنامه سازی و ... به صورت کامل در آنها تعریف نمی شود. - این سفارشی کردن در دو سطح اتفاق می افتد: سطح macro و micro. (سطح macro مربوط به یک گروه یا شرکت تولید کننده ی نرم افزار بوده و در آن کلیه ی استانداردهای مورد نظر آن گروه یا شرکت تعریف می شود و سطح micro معمولاً برای هر پروژه و با توجه به ماهیت آن پروژه تعیین می شود.)

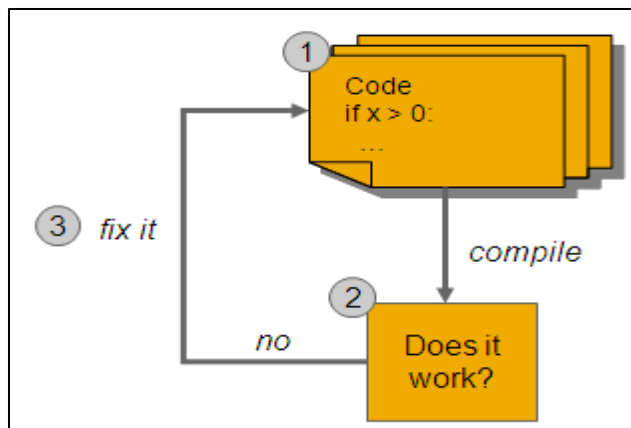
متدولوژی چیست؟ متدولوژی، خط مشی های گام به گامی است که برای تکمیل یک یا چند مرحله از مراحل چرخه تکامل بکار گرفته می شود.

انواع متدولوژی و فرایندها: Scrum -- Iterative - RAD - Spiral - XP - Agile - Incremental - prototype - waterfall - Code & fix و

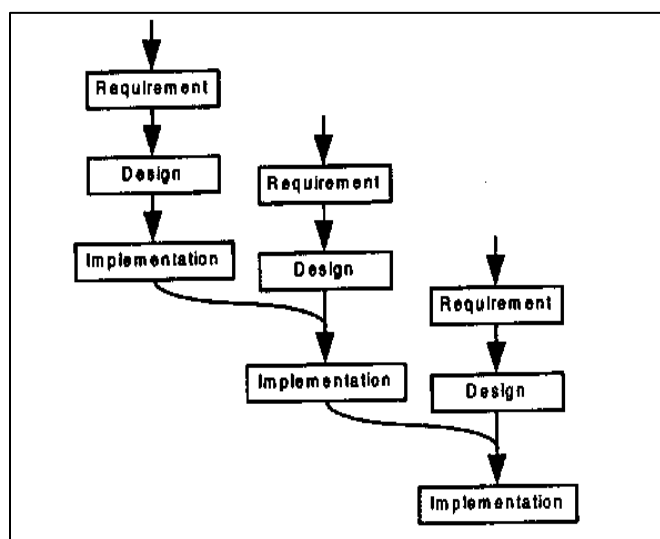
...



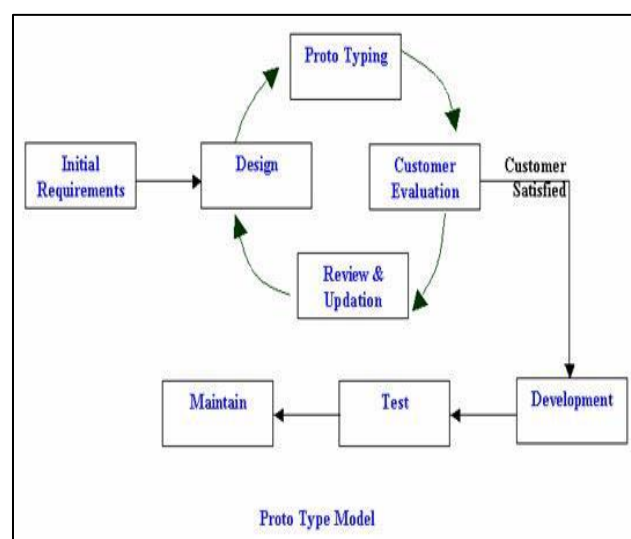
WATERFALL : بعد از اتمام هر مرحله به طور کامل ، امکان رفتن به مرحله ی بعد وجود دارد



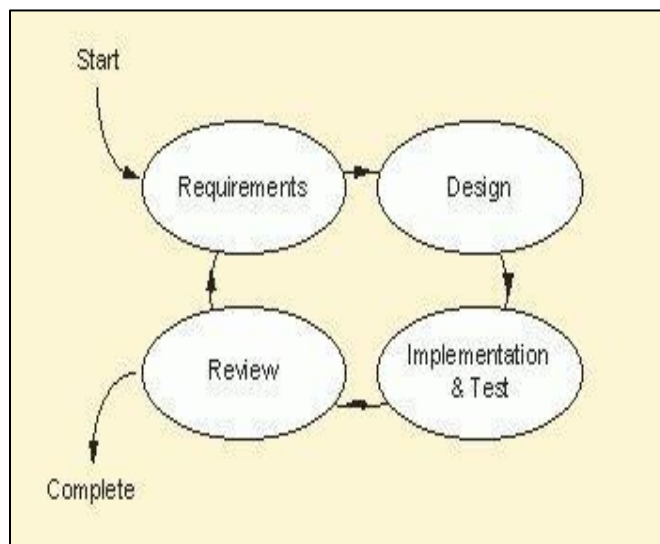
CODE & FIX



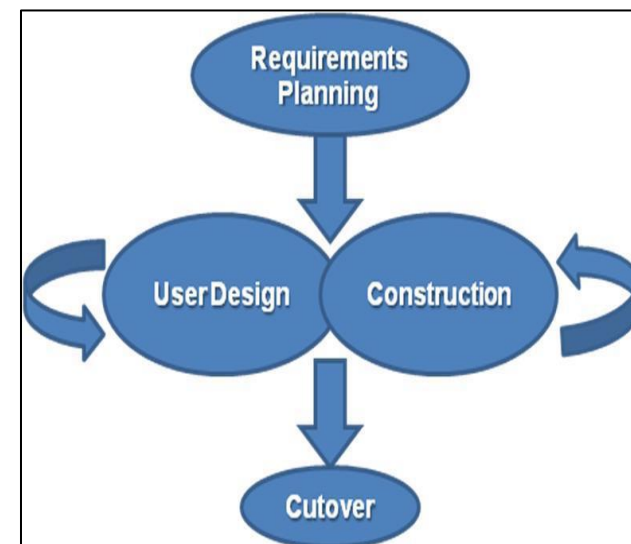
INCREMENTAL : هم زمان با پیشرفت کمی کار، کیفیت نیز ارتقا می یابد در واقع توسعه در طی روند کار صورت میگیرد.



PROTOTYPE : قبل از پیاده سازی سیستم ، یک تصویر کلی از سیستم ارائه می شود

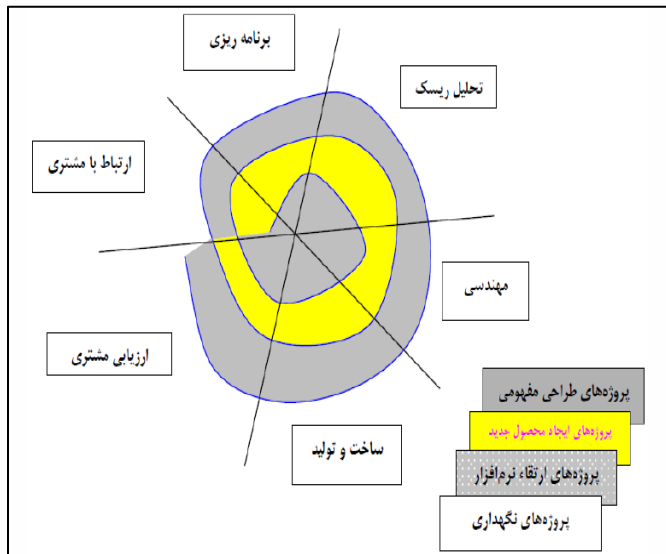


ITERATIVE : به عنوان مدل آشناری هم توصیف شده است که یک سری و مجموعه از آشنار های کوچک هستند

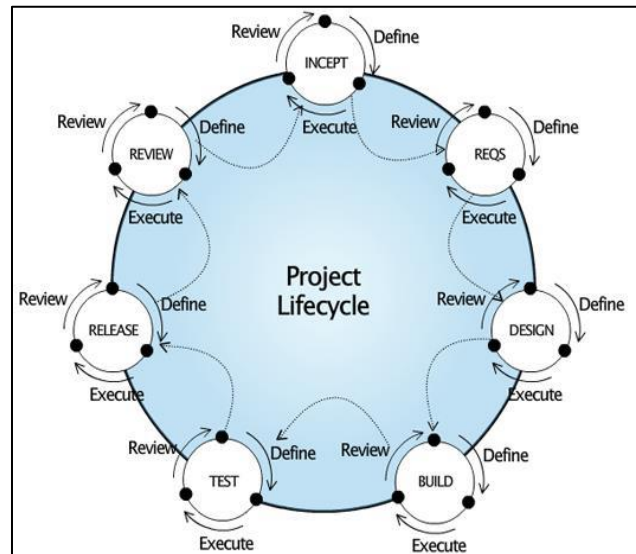


RAD : روش ساخت یک سیستم بسیار سریع و کوتاه می باشد.

موانع در مقابل متدولوژی RAD : ۱. هرچه پروژه بزرگتر شود، تشخیص تعداد تیم های RAD مشکلتر میشود. ۲. محدودیت زمانی (شصت تا نود روز) ۳. برخی از سیستم ها نمیتوانند پیمانه بندی شود.



SPIRAL: خصوصیت کلیدی مدل حلزونی مدیریت ریسک در تمام مراحل چرخه تولید نرم افزار است.



SCRUM: اسکرام یک روش چابک، تکراری و افزایشی برای مدیریت پروژه است.

*** تمامی این روش ها دارای مزایا و معایبی هستند که بسته به نوع مشخصات سیستم مورد نظر و نیازهای کاربر انتخاب میشوند و هیچکدام را به طور قطع، نمی توان برتر در نظر گرفت. ***

شرح کامل چرخه حیات تولید نرم افزار SDLC: روندی است که تحلیلگران فن آوری اطلاعات با استفاده از آن اقدام به تولید و طراحی سیستم های نرم افزاری و انطباق آن ها با نیازهای مشتریان نموده و در تحلیل خود نیازمندی های دنیای واقعی را مدنظر قرار می دهند. SDLC حروف اختصاری Software Development Life Cycle است که به معنی چرخه حیات تولید نرم افزار یا سیستم است و برخی از تحلیلگران آن را روند تولید نرم افزار می نامند. چرخه حیات نرم افزار چارچوبی است که کلیه عملیاتی را که در هر مرحله از تولید نرم افزار انجام می گیرد تعریف می نماید. چرخه حیات تولید نرم افزار (SDLC) روندی است که در تولید و یا توسعه و یا تجدید طراحی يك نرم افزار در يك سازمان دنبال می گردد. این روند کلیه عملیات از جمله نحوه تولید، نگهداری، جایگزینی و یا تغییر و یا بهبود کیفی نرم افزار را در بر می گیرد.

مراحل مختلف چرخه حیات تولید نرم افزار SDLC: برنامه ریزی - تعریف - طراحی - تولید نرم افزار - آزمایش - پیاده سازی

توضیح مرحله ۱: برنامه ریزی و تجزیه و تحلیل نیازمندی ها: تجزیه و تحلیل نیازمندی ها، مهمترین و اساسی ترین مرحله در روند چرخه حیات تولید نرم افزار محسوب می گردد. این امر توسط کارشناسان ارشد و اعضای متبحر تیم با استناد به داده های ورودی مشتریان، بخش فروش و بررسی های بازار و متخصصین این حوزه از صنعت انجام می گیرد. پس از این مرحله، اطلاعات گردآوری شده، جهت طراحی اساس و پایه پروژه و مطالعه امکان سنجی آن در حوزه های اقتصادی، عملیاتی و فنی مورد استفاده قرار می گیرد. همچنین

شناسائی ریسک ها و خطرات بالقوه پروژه نیز در مرحله برنامه ریزی انجام می گیرد. نتیجه مطالعات و امکان سنجی ها موجب بدست آوردن دیدگاه های مختلف فنی خواهد بود که می تواند باعث پیاده سازی موفقیت آمیز پروژه با حداقل ریسک گردد.

توضیح مرحله ۲: تعریف نیازمندی ها: زمانی که تجزیه و تحلیل ها به پایان میرسد مرحله بعد یعنی تعریف دقیق نیازمندی ها و مستند نمودن آن آغاز می شود. این مستندات شامل نیازهای نرم افزار مورد نظر بوده که مورد تأیید مشتری و همچنین تحلیلگران بازار قرار گرفته است. سند اصلی در این مرحله سند مشخصات نیازمندی های نرم افزار است که شامل کلیه نیازهای محصول بوده که در روند چرخه حیات پروژه می باید طراحی و فراهم گردد.

توضیح مرحله ۳: طراحی معماری نرم افزار: سند مشخصات نیازمندی های نرم افزار (SRS) Specification Requirement Software يك سند مرجع برای معماری نرم افزار محسوب می گردد که در آن انواعی از بهترین معماری ها جهت تولید نرم افزار مورد نظر تشریح گردیده است. کارشناسان معماری سیستم ها معمولاً براساس نیازمندی های مندرج در مستند مذکور، يك و یا دو طرح پیشنهادی را جهت تعیین نهائی معماری مورد نیاز انتخاب و در مستند دیگری بنام سند مشخصات طرح (DDS) Specification Document Design ارائه و پیشنهاد می نمایند. سند مذکور توسط کلیه ذینفعان و براساس معیارهای گوناگون نظیر ارزیابی ریسک، پیمانگی طرح (modularity)، قابلیت استفاده در شرایط مختلف، محدودیت زمانی، بودجه و ... مورد بررسی و ارزیابی قرار گرفته و سرانجام بهترین طراحی جهت تولید نرم افزار انتخاب می گردد. در طراحی نرم افزار کلیه و تمامی جنبه های معماری و همچنین روش های ارتباطی و جریان ارسال و دریافت داده ها از سامانه های دیگر (در صورت وجود آنها) مد نظر قرار می گیرد. در معماری پیشنهادی، باید طراحی داخلی کلیه مدل ها و پیمانه ها بنحوی کاملاً واضح و آشکار و با ذکر تمامی جزئیات در سند مشخصات طرح (DDS) درج گردد.

توضیح مرحله ۴: تولید نرم افزار: در این مرحله از چرخه حیات تولید نرم افزار، روند عملیاتی آغاز و نرم افزار تولید می گردد. در این مرحله کد برنامه به همان صورتی که در سند مشخصات طرح تأکید گردیده بود تهیه می گردد. چنانچه طرح بر مبنای جزئیات مندرج در سند مذکور و بصورت سازمان داده شده پیاده شود تولید نیز بدون هیچ مشکلی انجام خواهد شد. تولید کنندگان نرم افزار می بایست از استانداردهای کدنویسی که بوسیله سازمان خود و ابزار برنامه نویسی تعریف شده است تبعیت نمایند. زبان های مختلف برنامه نویسی سطح بالا نظیر C# و ++C، جاوا، PHP، پایتون و برای برنامه نویسی عموماً مورد استفاده قرار می گیرند. انتخاب زبان برنامه نویسی معمولاً با توجه به نوع نرم افزاری که باید تولید شود انتخاب می گردد.

تولید مرحله ۵: آزمایش محصول: این مرحله معمولاً زیر مجموعه ای از کلیه مراحل دیگر در مدل های نوین چرخه حیات تولید نرم افزار محسوب می گردد. عملیات مربوط به آزمایش محصول اغلب در کلیه مراحل تولید محصول انجام می گیرد. ولی آزمایش در این مرحله زمانی انجام می گیرد که نقص یا کاستی هایی مشاهده و یا گزارش شده باشد. بدین ترتیب کارشناسان پس از پیگیری نقص و یافتن علل، نقص یا کاستی را بر طرف و مجدداً تا زمانی که محصول در سطح استاندارد که در سند مشخصات نیازمندی های نرم افزار مشخص گردیده است قرارگیرد نرم افزار را آزمایش می نمایند.

مرحله ۶: پیاده سازی در بازار و نگهداری: زمانی که نرم افزار مراحل نهایی آزمایش را طی و آماده پیاده سازی شد، عرصه رسمی آن در يك بازار ویژه آغاز می گردد. گاهی اوقات پیاده سازی نرم افزار در يك سازمان بصورت مرحله ای انجام می گیرد و این امر بستگی به استراتژی کسب و کار سازمان مذکور دارد. امکان دارد که يك نرم افزار در ابتدای امر در بخش محدودی از يك محیط تجاری واقعی پیاده سازی شود که این امر "آزمایش به شرط قبول کاربر" نامیده می شود. در این شرایط و با توجه به بازخوردها امکان دارد که پیشنهادات دیگری برای کارکرد بهتر محصول ارائه و تغییراتی متناسب با نیازهای بازار انجام گیرد. پس از پیاده سازی نرم افزار مرحله نگهداری و پشتیبانی از مشتری موجود آغاز می گردد.

انواع مدل های چرخه حیات تولید نرم افزار: مهمترین و مشهور ترین مدل های چرخه حیات تولید نرم افزار: ۱. مدل آبشاری (waterfall model) ۲. مدل تکراری (iterative model) ۳. مدل حلزونی (spiral) ۴. مدل V (V model) ۵. مدل بیگ بنگ (Big Bang model) ۶. مدل سریع Agile model ۷. مدل تولید سریع برنامه کاربردی (RAD) Development Application Rapid

تعریف مدل آبشاری: مدل آبشاری اولین روش در زمینه چرخه حیات تولید نرم افزار محسوب شده و نرم افزارهای اولیه با این روش تهیه می شدند. این مدل به نام چرخه خطی و پی در پی نیز شهرت دارد. فراگیری و استفاده از این مدل بسیار آسان و ساده می باشد. در مدل آبشاری، هر مرحله می باید قبل از اینکه مرحله بعدی شروع شود پایان یابد و بین هیچ يك از مراحل تداخل یا هم پوشانی وجود ندارد و تمامی مراحل مستقل از یکدیگر می باشند.

مراحل پی در پی در مدل آبشاری عبارتند از: ۱. جمع آوری و تجزیه و تحلیل نیازمندی ها: کلیه نیازمندی های لازم جهت تولید نرم افزار در این مرحله جمع آوری و در سند مشخصات نیازمندی ها ثبت می گردد.

۲. **طراحی سیستم:** مشخصات کامل نیازمندی ها که در مرحله اول جمع آوری شده است در این مرحله مورد مطالعه دقیق قرار گرفته و طراحی سیستم انجام و طرح لازم بدست می آید. در طراحی سیستم نیازهای سخت افزاری و نرم افزاری مشخص شده و باعث می گردد که معماری کلی سیستم تعریف گردد.

۳. **اجرا:** در این مرحله با استفاده از طرح سیستم، ابتدا برنامه هایی به مقیاس كوچك بنام واحدها نوشته می شود که در مرحله بعدی با یکدیگر ادغام می گردند.

۴. **ادغام و آزمایش:** کلیه واحدهایی که در مرحله تولید و اجرا با یکدیگر ادغام شده و به شکل يك سیستم در آمده اند مورد آزمایش قرار می گیرند. پس از ادغام، کل سیستم جهت رفع نقایص و کمبود ها مجدداً آزمایش می گردد.

۵. **پیاده سازی:** زمانی که کلیه آزمایشات با موفقیت به انجام رسید، محصول در محیطی که مشتری پیش بینی نموده است پیاده سازی می گردد و یا اینکه مستقیماً به بازار فروش ارسال می گردد.

۶. **نگهداری:** برخی از مشکلات وجود دارند که صرفاً در محیط مشتری بروز می نمایند. برای رفع این مشکلات اغلب تغییرات و اصلاحاتی (patches) انجام می گیرد و گاهی برای ارتقای سیستم نسخه جدیدی تهیه می گردد. تغییرات و یا نسخه جدید مجدداً در محیط کار مشتری نصب و راه اندازی می گردد.

*** هر نرم افزاری دارای چرخه حیات مطلوب و متفاوت خود می باشد و این امر مبتنی بر يك سری عوامل داخلی و خارجی می باشد. ***

کاربرد مدل آبشاري : ۱. نیازها کاملاً روشن و ثابت و مستند به عوامل شناخته شده می باشند. ۲. تعریف محصول ثابت و پایدار است. ۳. فن آوری کاملاً درك شده و تغییر پذیر نمی باشد. ۴. نیازها بصورت مبهم و غیر واضح نمی باشند. ۵. منابع و کارشناسان مجرب فراوان جهت پشتیبانی از محصول وجود دارند. ۶. طول عمر پروژه کوتاه است.

مزایا مدل آبشاري : مزیت تولید نرم افزار بر طبق مدل آبشاري این است که در این مدل عملیات هر يك جداگانه انجام و کنترل می گردند. در این مدل میتوان يك برنامه کاری تنظیم و برای هر مرحله نقطه آغاز و پایانی در نظر گرفت و محصول می تواند براساس برنامه مذکور مرحله به مرحله پیشرفت و هر يك از مراحل بعد از کنترل مرحله بعد را بدنبال داشته باشد.

معایب مدل آبشاري : یکی از معایب مدل آبشاري عدم امکان بازگشت و تجدید نظر در تولید نرم افزار یا محصول است. هنگامی که یکی از مراحل در جریان تست و آزمایش قرار دارد، بازگشت به عقب و اعمال تغییر در فعالیت هایی که بصورت مستند در نیامده است بسیار مشکل می باشد.

نظرات مخالفین مدل آبشاري: ۱. هیچ نرم افزاری در این مدل تا پایان چرخه حیات تولید نمی گردد. ۲. ریسك و عدم اطمینان در این مدل بسیار بسیار بالا است. ۳. مدل خوبی برای پروژه های پیچیده و یا شی گرا نمی باشد. ۴. برای پروژه های طولانی مدت و مداوم مدل مناسبی نیست. ۵. برای پروژه هایی که نیازمندی ها و تامین آنها در معرض ریسك بالا و یا متوسط هستند و نیازمندی ها باید تغییر یابند مدل مناسبی نمی باشد. ۶. اندازه گیری پیشرفت کار در هر مرحله مشکل است. ۷. این مدل نمی تواند سازگار با تغییر نیازها باشد. ۸. تعدیل هدف در طول چرخه حیات می تواند به انجام پروژه پایان دهد.

نظرات موافقین مدل آبشاري : ۱. این مدل بسیار ساده و درك و استفاده از آن نیز آسان است. مدیریت مدل به لحاظ استحکام مدل آسان است چون هر يك از مراحل دارای اقلام خاص قابل تحویل و روند بررسی خاص خود می باشد. ۲. مراحل و روندها در زمان خود پردازش و تکمیل می گردند. ۳. اجرای پروژه های کوچکتر در صورتیکه نیازها بخوبی فهمیده شوند بسیار آسان است. ۴. مراحل را می توان بخوبی و روشنی از یکدیگر تفکیك نمود. مراحل اساسی بخوبی قابل درك میباشند. ۵. ترکیب و تعریف وظایف بسیار آسان است. ۶. روندها و نتایج را میتوان بخوبی مستند نمود.

تعریف مدل تكراري (Iterative model) : در مدل تكراری، عملیات با اجرای ساده يك مجموعه از نیازهای نرم افزاری شروع و رفته رفته نسخه های کامل تری تهیه و اضافه می شود تا نهایتاً سیستم بطور کامل پیاده سازی شود. در مدل چرخه حیات تكراری، تلاش برای شروع با تمامی نیازمندی ها انجام نمی گردد، بلکه تولید با شناسائی و اجرای صرفاً بخشی از نرم افزار شروع و سپس بررسی برای

شناخت نیازهای بیشتر آغاز می گردد. این روند سپس تکرار شده و در پایان هر مرحله تکرار مدل، نسخه جدیدی از نرم افزار تهیه می شود. ایده اصلی در این روش، تولید سیستم از طریق چرخه های تکرار در مدت زمانی کمتر می باشد. در مدل تکراری، کلیه نیازمندی ها را در بخش های مختلف تقسیم می کنند. طی هر دوره تکرار، از مرحله نیازمندی ها به بخش های طراحی، تولید و تست عبور می نماید و نسخه جدید نرم افزار وظایف و عملیات جدیدی را نسبت به نسخه قبل اضافه می نماید. این روند تا تکمیل شدن کامل و انطباق با کل نیازمندی ها ادامه می یابد. کلید کاربرد موفقیت آمیز این مدل، شناخت کامل و صحیح نیازمندی ها تا تأیید و تصدیق آنها و تست و آزمون هر نسخه از نرم افزار با نیازمندی ها در هر چرخه از مدل می باشد. در هر مرحله که نرم افزار در چرخه های پی در پی پیش می رود آزمایش ها نیز جهت تأیید هر نسخه از نرم افزار نیز می بایست تکرار شوند. این مدل را می توان بعنوان مدل تکاملی و یا روش صعودی نیز نام گذاری کرد.

کاربرد مدل تکراری : ۱. نیازمندی های کل سیستم تعریف و شناسائی شده باشند. ۲. نیازمندی های عمده می باید تعریف شده و عملکردهای جدید و یا ارتقا جدید در نرم افزار در طی زمان اضافه شوند. ۳. محدودیت های زمانی در بازار باید مد نظر قرار گیرد. ۴. فناوری های جدید می باید توسط تیم تولید در زمان کار بر روی پروژه مورد استفاده قرار گیرد. ۵. مجموعه ای از منابع انسانی و مهارت های مورد نیاز که در وضعیت موجود وجود ندارند در هر مرحله از حیات از طریق انعقاد قرارداد در انجام پروژه به کار گرفته شوند. ۶. برخی ویژگی ها و یا اهداف پر ریسک وجود دارند که در آینده می باید نسبت به تغییر آنها اقدام نمود.

مزایا مدل تکراری : امتیاز این مدل در این است که در اولین مرحله تولید، مدلی از کارکرد سیستم وجود دارد که می توان از روی آن کاستی های عملیاتی و یا طراحی را یافت. یافتن این موارد در مراحل اولیه تولید، کارشناسان را قادر می سازد با یک بودجه محدود اقدامات اصلاحی انجام دهند.

معایب مدل تکراری : عیب این مدل این است که این مدل صرفاً برای پروژه های عظیم و حجیم کاربرد دارد. علت این مسئله اینست که شکستن یک سیستم نرم افزاری کوچک و تقسیم آن به مدل های سرویس دهنده کوچکتر بسیار مشکل می باشد.

نظرات مخالف مدل تکراری : ۱. در این مدل منابع بیشتری مورد نیاز میباشد. ۲. گرچه هزینه اعمال تغییر در این مدل کمتر است ولی جهت تغییر نیازمندی ها چندان مطلوب نمی باشد. ۳. این مدل نیاز به توجه بیشتری از سوی مدیریت دارد. ۴. در معماری و یا طراحی سیستم ممکن است مسائل بیشتری بوجود آید زیرا در شروع تمامی چرخه های حیات نمی توان تمامی نیازمندی ها را گردآوری نمود. ۵. تعریف هر یک از مراحل تکراری نیاز به تعریف سیستم بطور کلی دارد. ۶. این مدل برای پروژه های کوچک مناسب نیست. ۷. پیچیدگی عملیات مدیریتی بیش از سایر مدل ها است. ۸. پایان پروژه قابل پیش بینی نیست و این امر دارای ریسک میباشد. منابع انسانی کاملاً متبخر جهت تجزیه و تحلیل ریسک مورد نیاز می باشد. ۹. پیشرفت پروژه کاملاً وابسته به مرحله تجزیه و تحلیل ریسک می باشد.

نظرات موافق مدل تکراری : ۱. برخی عملیات را می توان سریعاً و در مراحل اولیه چرخه حیات تولید داد. ۲. نتایج عملیات بسیار سریع و بصورت دوره ای بدست می آید. ۳. در این مدل میتوان تولید را بصورت موازی برنامه ریزی نمود. ۴. پیشرفت تولید را میتوان اندازه گیری کرد. ۵. تغییر طرح نهائی و نیازها با هزینه کمتری انجام میگردد. ۶. آزمایش و اشکال زدائی در یک زمان کوتاه تر انجام میگردد. ۷. ریسک ها در یک دوره کوتاه تر شناسائی و حل میگردند و مرحله تکراری به سادگی مدیریت میشود. ۸. مدیریت ریسک در این مدل آسان

تر است و مدیریت ریسک های بزرگتر قبل از دیگر ریسک ها انجام میگیرد. ۹. در این مدل در هر مرحله بخشی از محصول تهیه میگردد. ۱۰. حل مسائل، چالش ها و ریسک ها که در هر مرحله شناسائی میگردند میتواند در مرحله بعدی مورد استفاده قرار گیرد. ۱۱. تجزیه و تحلیل ریسک ها بنحو مطلوب تر انجام میگیرد. ۱۲. این مدل از تغییر نیازمندی ها پشتیبانی مینماید. ۱۳. زمان انجام عملیات ابتدائی کوتاه تر است. ۱۴. کاربرد این مدل برای پروژه های بلند مدت و یا حیاتی مناسب تر است. ۱۵. در خلال چرخه حیات، نرم افزار سریع تر تولید شده و ارزیابی مشتری را آسان تر می سازد.

مدل حلزونی (spiral model): مدل حلزونی را می توان ترکیبی از مدل چرخه حیات تکراری و جنبه های کنترل شده و اصولی مدل آبشاری دانست. مدل حلزونی ترکیبی از مدل روند تولید تکراری و مدل تولید خطی و متوالی یعنی مدل آبشاری با تاکید بسیار بر روی تجزیه و تحلیل ریسک می باشد. در این مدل هر بار تکرار عملیات در حلقه حلزونی می توان محصول یا نرم افزار را ارتقا داده و در هر بار آنرا پالایش داد.

مراحل مدل حلزونی: هر پروژه تولید نرم افزار بصورت مکرر از این مراحل عبور کرده و مسیر حلزونی را طی می نماید: ۱- **ارتباط با مشتری و شناسائی:** این مرحله با گردآوری نیازمندی های کسب و کار شروع می شود. در مسیر حلزونی بعدی زمانی که محصول کاملتر گردید، شناسائی نیازمندی های سیستم، سیستم های فرعی و همچنین نیازمندی واحدها در این مرحله انجام می گیرد. درك نیازمندی های سیستم از طریق ارتباط دائم مشتری و تحلیلگر سیستم امکان پذیر می باشد. در پایان این مسیر محصول یا نرم افزار در بازار مورد نظر پیاده سازی می شود. ۲- **طراحی و ارزیابی ریسک:** این مرحله با طراحی ذهنی و در مرحله اولیه انجام می گیرد که شامل طراحی معماری، طراحی منطقی واحدها و بخش های مختلف محصول (پیمانه ها) طراحی فیزیکی و تهیه طرح نهائی است که در مراحل بعدی انجام می گردد. تجزیه و تحلیل ریسک عبارتست از، شناسائی، پایش امکانات فنی و مدیریت ریسک از جمله انطباق محصول با طرح نهائی و محاسبه هزینه با برآورد اولیه. ۳- **مهندسی و ساخت یا تولید:** در این مرحله، نرم افزار یا محصول واقعی با توجه به طرح نهائی و با در نظر گرفتن نیازهای مشتری تولید می گردد. پس از تولید نرم افزار یا محصول جهت آزمایش با درج شماره نسخه جهت دریافت نظر مشتری و اعمال تغییرات نهائی ارسال می گردد. ۴- **بررسی و ارزیابی:** در این مرحله مشتری نرم افزار را ارزیابی و نظرات نهائی خود را اعلام می نماید.

کاربرد مدل حلزونی: مدل حلزونی بنحو بسیار گسترده ای در صنعت نرم افزار مورد استفاده می باشد زیرا این مدل همگام با روند تولید طبیعی هر محصول بوده و در حالی که کمترین ریسک را برای مشتری می تواند به دنبال داشته باشد، منافع شرکت تولید کننده را نیز به همراه دارد. ۱. در مواردی که بودجه دارای محدودیت است و هزینه ها جزو مسائل حیاتی بشماره می رود و ارزیابی نیز با اهمیت تلقی می گردد این مدل می تواند مطلوب باشد. ۲. این مدل برای پروژه پر ریسک و نیز با ریسک متوسط، مدلی موثر محسوب میگردد. ۳. این مدل برای پروژه های بلند مدت نیز مدلی مطلوب بشمار می رود. زیرا تعهدات انجام این پروژه بلحاظ ایجاد تغییرات بالقوه در الویت های اقتصادی بسیار زیاد بوده و نیازمندی ها نیز در طول زمان دچار تغییر می گردد. ۴. در مواردیکه مشتری به نیازمندی های خود کاملاً اطمینان ندارد نیز این مدل می تواند کاربرد داشته باشد. ۵. در مواردیکه نیازمندیهای مشتری پیچیده ست و روشن ساختن آنها نیاز به ارزیابی بیشتر دارد نیز این مدل میتواند کاربرد داشته باشد. ۶. در مورد خطوط تولید محصول جدید که می باید در مراحل مختلف انجام گرفته و در هر

مرحله نظرمشتری کسب شود نیز این مدل میتواند مورد استفاده قرار گیرد. ۷. در مواردیکه در چرخه تولید محصول انتظار ایجاد تغییرات اساسی می رود نیز این مدل موثر باشد.

مزایا مدل حلزونی : یکی از مهمترین فواید مدل چرخه حیات حلزونی اینست که در این مدل میتوان عناصر تولید را در هر زمان که مورد شناسایی و یا در دسترس قرار گیرد به محصول اضافه نمود و این امر تضمین می نماید که بین نیازمندی های قبلی و طرح تدوین شده مغایرتی به وجود نخواهد آمد. یکی دیگر از جنبه های مثبت این مدل اینست که کاربران را در همان اوایل کار درگیر کار بر روی تولید سیستم می نماید.

معایب مدل حلزونی : از سوی دیگر، این مدل، مدیریت را در تکمیل پروژه بسیار سخت مشغول نموده و این خطر وجود دارد که مدل حلزونی مدیریت را در یک جنبه نامعین گرفتار نماید، بدین خاطر این مدل نیاز به نظمی سخت در ایجاد تغییرات و دامنه درخواست تغییرات دارد و این نظم نقش بسیار مهمی در تولید و پیاده سازی موفقیت آمیز پروژه خواهد داشت.

نظرات مخالف مدل حلزونی : ۱. مدیریت در این مدل بسیار پیچیده می باشد. ۲. پایان پروژه را نمی توان بزودی تخمین زد. ۳. با پروژه های کوچک و یا طرحهای کم ریسک سازگار نبوده و هزینه آن برای پروژه های کوچک بسیار بالا میباشد. ۳. روند عملیات پیچیده میباشد. ۴. تعداد بسیاری از مراحل میانی نیازمند مستند سازی بیش از اندازه میباشد.

نظرات موافق مدل حلزونی : ۱. تغییر نیازمندیها را می توان با طرح وفق داد. ۲. استفاده گسترده از پیش الگو ها امکان پذیر است. ۳. مقررات را می توان بطور صحیح اعمال نمود. ۴. اربابان بسیار زود سیستم را مشاهده خواهند نمود. ۵. تولید را می توان به بخشهای کوچکتر تقسیم و یا بخشهای پر ریسک را زودتر انجام داد که این امر مدیریت ریسک را آسانتر می سازد.

تعریف مدل چابک (Agile Model) : مدل چابک در چرخه حیات تولید نرم افزار ترکیبی از مدل تکراری و مدل آبشاری است که بر توافق و رضایت مشتری توجه داشته و بر تحویل سریع محصول نرم افزاری به مشتری تاکید دارد. در مدل چابک روند تولید محصول به مراحل کوچک تقسیم می گردد و مراحل بصورت مکرر ادامه می یابد. مراحل تکراری هر یک در هر دوره یک الی سه هفته به درازا می کشد و در مرحله تکراری تیم های کاری بصورت همزمان بر در حوزه های مختلف نظیر برنامه ریزی، تجزیه و تحلیل نیازمندی ها، طراحی، کد نویسی، آزمون دستگاه و آزمون قبولی کار می کنند. در پایان مراحل تکراری، محصول به مشتریان و ذینفع های عمده نمایش و ارائه می گردد. گروه هایی که پروژه تولید نرم افزار را از طریق مدل چابک انجام می دهند، اعتقاد دارند که هر پروژه ای را باید بصورت مختلف اجرا نمود و روش های موجود را باید با مناسب ترین شکل با نیازمندی های پروژه انطباق داد. در مدل چابک، وظایف به ظرفیت های زمانی (چهارچوب های زمانی کوتاه) تقسیم می گردد تا در این زمان ها بتوان ویژگی ها و خصیصه هائی را جهت انتشار و تهیه نسخه ای از محصول بدست آورد. اتخاذ شیوه تکراری در مدل چابک باعث می گردد که پس از هر مرحله تکرار، بخشی از نرم افزار تحویل گردد. هر یک از بخش ها از نظر ویژگی ها حالت تصاعدی و افزایشی داشته و بخش نهایی دارای کلیه ویژگی های مورد نیاز مشتری میباشد.

روش های مختلف مدل چابک : روش مدلسازی چابک در تولید محصولات نرم افزاری بسیار زود آغاز گردید و با گذشت زمان بلحاظ

داشتن قابلیت انعطاف و سازگاری، شهرت بسیار پیدا کرد و در سالهای اخیر بصورت عمده در جهان نرم افزار پذیرفته شده است.

Rational unified process (۱۹۹۴) – Scrum (۱۹۹۵) – Crystal clear Extreme programming (۱۹۹۶) – Adaptive software Development (ASD) – Feature Driven Development (FDD) – Dynamic system Development Method (DSDM)(۱۹۹۵)

اصول مشترک روش های مختلف مدل چابک: ۱- افراد و فعل و انفعالات: در روش چابک، خود سازمان دهی و انگیزه بهمان میزان تعامل و همکاری درون گروهی و موقعیت و همچنین برنامه نویسی گروهی دارای اهمیت است. ۲- نرم افزار کارآمد: نرم افزار نمایشی کارآمد بعنوان بهترین وسیله ارتباطی با مشتری در جهت شناخت هر چه بیشتر نیازهای وی محسوب می گردد و این شیوه به ارائه مستندات ارجحیت دارد. ۳- همکاری مشتری: از آنجایی که شناخت نیازمندی ها بنابر عوامل متعدد در ابتدای شروع پروژه بصورت کامل قابل جمع آوری نمی باشد، لذا همکاری متقابل و مداوم مشتری در جهت بدست آوردن نیازمندی های محصول از اهمیت خاصی برخوردار می باشد. ۴- واکنش در قبال تغییرات: مدل چابک دارای قابلیت پاسخگوئی سریع به تغییرات و تولید مداوم دارد.

مقایسه مدل چابک با مدل های سنتی: ۱. مدل چابک مدلی مبتنی بر روش تولید تطبیقی نرم افزار است در حالی که مدل های سنتی نظیر مدل آبشاری مبتنی بر دیدگاه فرضیات از پیش تعیین شده می باشد. ۲. گروه یا تیمی که با روشهای سنتی سروکار دارند معمولاً از روی برنامه ای که جزئیات آن بدقت تعیین گردیده کارکرده و کلیه وظایف به دقت کامل پیش بینی گردیده و زمان تحویل محصول نیز در چند ماه بعد و یا در طول چرخه حیات مشخص و معین گردیده است. این مدل ها کاملاً بستگی به تجزیه و تحلیل نیازمندی ها و برنامه ریزی هائی دارد که در آغاز چرخه حیات انجام گرفته است. ۳. چنانچه در پروژه ای که با مدل های سنتی انجام می گیرد تغییری گنجانده شود مدیریت کنترل تغییرات، اولویت اعمال آنها پس از درخواست تغییر مورد بررسی قرار می دهد. ۴. در مقابل در مدل چابک که از دیدگاه انطباقی تبعیت می نماید هیچ گونه برنامه ریزی مشروح و دقیق وجود نداشته و وظایف آتی زمانی مشخص و روشن می گردند که ویژگی های پروژه نیاز به تغییر و یا تولید داشته باشند. در این مدل، تولید براساس ویژگی ها انجام و تیم یا گروه کاری خود را بصورت پویا با نیازمندی مربوط به ایجاد تغییر در محصول وفق می دهند. محصول، مکرراً در هر مرحله آزمایش شده و این مسئله ریسک های ناشی از عدم موفقیت های آتی را به حداقل می رساند. ۵. همکاری متقابل مشتری با گروه کاری، ستون فقرات متدولوژی چابک محسوب می گردد و ارتباط نزدیک همراه با تهیه حداقل مستندات، ویژگی بارز محیط تولید چابک را تشکیل می دهد. ۶. افراد گروه کاری در مدل چابک تشریک مساعی کامل با یکدیگر داشته و در اغلب اوقات در يك موقعیت جغرافیائی با یکدیگر همکاری می کنند.

نظرات مخالف مدل چابک: ۱. جهت اجرای پروژه های پیچیده مطلوب و مناسب نیست. ۲. تداوم، حفظ مراحل و تولید پذیری آن با ریسک توأم می باشد. ۳. داشتن برنامه کلی و مدیری چابک جهت انجام عملیات مدیریت پروژه ضروری می باشند که بدون آن این مدل بی فایده خواهد بود. ۴. مدیر پروژه، حوزه عملیات، عملکرد و تعدیل و تغییر ضرب العجل ها را تعیین می نماید. ۵. بستگی کامل به فعل انفعال مشتری دارد و چنانچه مشتری فاقد آگاهی باشد تیم پروژه می تواند مسیری نادرست را طی کند. ۶. این مدل بستگی بسیار به فرد داشته و بنابراین حداقل مستندات تهیه میگردد. ۷. انتقال فن آوری به اعضای جدید تیم احتمالاً بنا به دلیل کمبود مستندات می تواند بسیار چالش برانگیز باشد.

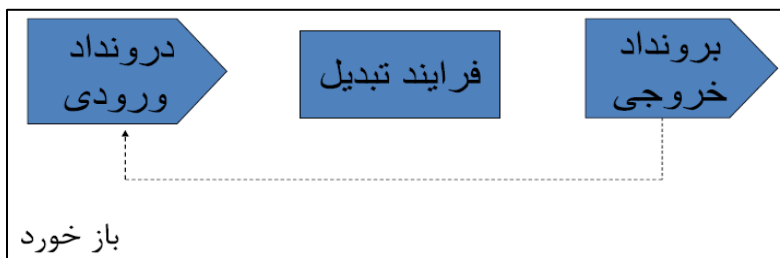
نظرات موافق مدل چابک : ۱. دیدگاهی بسیار واقع گرایانه در تولید نرم افزار است. ۲. کارگروهي را ترویج داده و آموزش متقابل را ارتقا می بخشد. ۳. کارکردها به سرعت تکامل یافته و اثبات می گردد. ۴. نیاز به منابع بسیار تقلیل می یابد. ۵. برای نیازمندی های ثابت و یا متغیر روشی مطلوب میباشد. ۶. راه حلهاي عملی مقدماتی سریعاً تحویل میشوند. ۷. برای محیط هائی که دائماً در حال تغییر میباشند مدلی مناسب بشمار میرود. ۸. با پیروی از کمترین قوانین و مقررات، مستندات به سادگی تهیه میگرددند. ۹. با تولید یا تولید همزمان، تحویل محصول در زمان برنامه ریزی شده میسر میگردد. ۱۰. نیاز به برنامه ریزی نداشته و یا به حداقل آن نیاز دارد. ۱۱. مدیریت آن آسان است. ۱۲. برای برنامه نویسان قابلیت انعطاف ایجاد می نماید.

تعریف سیستم (بامثال) : تعریف اول : مجموعه ای از اجزاء مرتبط با هم که برای تامین هدف یا اهداف مشترک با یکدیگر تعامل دارند را سیستم (سامانه) گویند. بدین ترتیب، اصطلاح سیستم مفهوم وسیع و گسترده ای دارد که شامل مصادیق مختلفی در زیست شناسی و امور فیزیکی و اجتماعی و نظائر آنها می شود. به عنوان مثال، در زیست شناسی بدن موجود زنده یک سیستم تلقی می شود که از اجزاء یا سیستم های فرعی متعددی (مانند دستگاه اعصاب، دستگاه گردش خون، دستگاه گوارش و غیره) تشکیل یافته است. در نتیجه فعل و انفعال و تأثیر متقابل اجزاء وابسته، پدیده کل (سیستم) ظاهر می شود. یا بعنوان مثال سازمان می تواند یک سیستم باشد. تعریف دوم : سیستم مجموعه ای است که از اجزاء به هم وابسته که وابستگی حاکم بر اجزای خود کلیت جدید را احراز کرده و از نظم و سازمان خاصی پیروی می نماید و در جهت تحقق هدف معینی که دلیل وجودی آن است فعالیت می کند.

سه خصلت اساسی سیستم ها : مفهوم کلیت - روابط متقابل بین اجزاء - وجود هدف

اجزاء چهار گانه سیستم: ۱- در ونداد (input) : آنچه بنحوی وارد سیستم می شود و سبب تحرک سیستم می شود ۲- فرایند تبدیل (process) : جریان تغییر و تبدیل آنچه وارد سیستم می شود ۳- برونداد (output) : آنچه از تغییر و تبدیل از سیستم (به شکل کالایا خدمات) خارج می شود ۴- بازخورد (feed back) : فرایندی دورانی که قسمتی از ستاده به عنوان اطلاعات به درونداد پس خورانده می شود.

ارتباط بین اجزا سیستم :



تعریف محیط سیستم : هر سیستم در محیطی قرار دارد . محیط سیستم شامل کلیه متغیرهایی است که می تواند در وضع سیستم مؤثر باشند و یا از سیستم تأثیر پذیرند عوامل محیطی در بر گیرنده عواملی همچون عوامل طبیعی ، فرهنگی ایدئولوژی ، اجتماعی ، سیاسی ، اقتصادی و غیره هستند.

تعریف آنتروپی : در هر سیستم عواملی وجود دارند که بر خلاف جهت نظم سیستم عمل می کنند و مختل کننده انتظام سیستم هستند این عوامل را آنتروپی می نامند.

انواع آنتروپی : ۱- **آنتروپی مثبت :** عملکردش در خلاف جهت نظم سیستم است ۲- **آنتروپی منفی :** عملکردش خلاف جهت آنتروپی مثبت است و برای ایجاد تغییرات تعدیلاتی در جهت اصلاح انحرافات به منظور بقاء سیستم در محیط عمل می کند.

طبقه بندی سیستم ها : سیستم اصلی و فرعی

تعریف سیستم اصلی و فرعی (بامثال) : حدود سیستم اصلی و سیستم های فرعی (Sub Systems) همیشه تابع قواعد و ضوابط معینی نیست. به عنوان مثال، اگر بدن انسان را یک سیستم اصلی فرض کنیم، هر یک از اجزاء آن (مانند دستگاه گوارش) سیستم فرعی بشمار می رود، ولی چنانچه خود دستگاه گوارش را سیستم اصلی تلقی نمائیم، اجزاء وابسته (مانند معده ، روده ، کبد و غیره) سیستم های فرعی سیستم اصلی (دستگاه گوارش) را تشکیل می دهند. به همین ترتیب، در داخل یک سازمان (به عنوان سیستم اصلی)، سازمان وابسته یک سیستم فرعی محسوب می شود، ولی اگر سازمان وابسته خود سیستم اصلی تلقی شود، در اینصورت سیستم های فرعی رفتار عبارتند از شبکه روابط اجتماعی ، شبکه روابط تخصصی، شبکه قدرت و غیره ...

انواع سیستم ها در ارتباط با محیط جانی : ۱- سیستم بسته (closed system) ۲- سیستم باز (open system)

تعریف سیستم بسته (closed system) : سیستمی ساده است که با محیط خود ارتباط برقرار نمی کند و بیشتر شبیه سیستم های مکانیکی در فیزیک و علوم تجربی است. در نظریه های کلاسیک سازمان را به عنوان سیستمی بسته تلقی می کردند که فقط متکی به عوامل و نیروهای داخلی است و بدون ارتباط با محیط خارج می تواند مسایل خود را از طریق سلسله مراتب و تقسیم کار و روابط رسمی در داخل دستگاه حل و فصل کند.

تعریف سیستم باز (open system) : سیستمی را باز گویند که با محیط اطراف خود کنش متقابل داشته باشد. در نظریه های جدید مدیریت، سازمان ، سیستم بازی است که به طور دائم با محیط اطراف خود در تعامل است. سیستم های مدیریت اطلاعات نمی توانند بسته باشند.

خواص سیستم های باز : ۱- کلیت و جامعیت وجودی ۲- سلسله مراتب ۳- همبستگی بین اجزاء ۴- تناسب بین اجزاء نگهداری ۵- گردش دایره وار ۶- خاصیت تولید مثل ۷- همپایانی ۸- گرایش به فنا ۹- گرایش به تکامل ۱۰- گرایش به تعادل یا خود

تعریف کلیت و جامعیت وجودی : سیستم در کلیت وجودی خود خواصی را ظاهر می سازد که در اجزاء تشکیل دهنده آن به تنهایی وجود ندارد . این کلیت نتیجه ارتباط اجزاء با یکدیگر و نحوه ترکیب اجزاء و سازمان یافتن آنها نیز کلیت سیستم را بوجود می آورد.

تعریف گرایش به فنا : در سیستم ها نوعی سلسله مراتب از نظر ساختاری ، عملکرد و رفتاری وجود دارد . در هر سیستم عناصری وجود دارد که به نوبه خود عناصر کوچکتری هستند که ساخت و عملکرد ساده تری دارند.

تعریف سلسله مراتب : در سیستم ها نوعی سلسله مراتب از نظر ساختاری ، عملکرد و رفتاری وجود دارد . در هر سیستم عناصری وجود دارد که به نوبه خود عناصر کوچکتری هستند که ساخت و عملکرد ساده تری دارند.

تعریف سلسله مراتب سیستم ها از دیدگاه بولدینگ : از دیدگاه بولدینگ سیستم ها از ساده به پیچیده به نه مرتبه تقسیم بندی می شوند: سطح اول : ایستا یا بافتها . سطح دوم : دینامیک ساده . سطح سوم : سایبرنتیک . سطح چهارم : سطح یاخته یا سیستم های باز . سطح پنجم : نباتات یا ارگانیسم ها . سطح ششم : سطح حیوان . سطح هفتم : سطح انسان . سطح هشتم : سیستم های اجتماعی و چارچوب وجودی . سطح نهم : سیستم های نمادین یا استعلایی

تعریف همبستگی بین اجزاء : هر جزء در سیستم به نحوی با سایر اجزاء مرتبط است و به علت وجود این همبستگی چنانچه در جزئی خللی وارد شود ، سایر اجزاء نیز از آن خلل متأثر می شوند .

تعریف تناسب بین اجزاء نگهداری : بین اجزاء سیستم تناسب ، سنخیت و کمال متقابل موجود است و وجود تناسب سبب حفظ هویت و کلیت سیستم می شود .

تعریف گردش دایره وار : فرایند در ونداد ، تبدیل و برون داد جریانی مستمر و مداوم است . به این معنی که با صدور برون داد ، سیستم بار دیگر آماده کسب نیرو و تجدید فعالیت گردیده و این جریان به شکل گردشی دایره وار ادامه می یابد .

تعریف خاصیت تولید مثل : سیستم ها گرایش به جاودانه سازی خود دارند و تا جایی که امکان داشته باشد به حیات خویش ادامه می دهند و چنانچه در کار سیستم نقصی پدید آید در رفع آن می کوشند و برای ادامه حیات تلاش می کنند ، در غیر این صورت از طریق تولید مثل وجود خود را در دیگری ادامه می دهند .

تعریف همپایانی : همپایانی بدین معنی است که سیستم می تواند از راهها و مسیرهای متفاوتی به هدف واحدی برسد .

تعریف گرایش به فنا : در درون سیستمها عواملی بوجود می آیند که سیستمها از جهت اصلی آن منحرف می سازند و به سمت عدم تعادل سوق می دهند . این عوامل را آنتروپی می خوانند .

تعریف گرایش به تکامل : منظور از تکامل از پیچیدگی ساخت و تنوع خواص است و چنانچه ساختار سیستم پیچیده تر شود و در اثر آن پیچیدگی ، عملکردهای متنوعتری از سیستم به ظهور رسد و خواص بیشتری ارائه شود سیستم متکامل تر شده است . *تکامل سیستم ها به دو شکل تدریجی و یا جهشی و ناگهانی انجام می گیرد . *

تعریف گرایش به تعادل یا خود نگهداری پویا : این خصیصه که به هوموستاسیس معروف است بیانگر تلاش سیستم در حفظ متغیرهای ضروری خود در محدوده های معین به منظور ادامه حیات سیستم می باشد .

نظریه عمومی سیستم ها : این نظریه توسط برتalanفی ارائه گردید و بر اساس این نظریه یک ارگانیسم ، صرفاًمجموع عناصر جداگانه ای نبوده بلکه سیستمی است که دارای نظام و کلیت می باشد که مرتباًدر حال تغییر و تبدیل است به اعتقاد وی ارگانیسم را نمی توان با شیوه

تفکر و روشهای معمول در مکتب مکانیسمی شناخت و باید طرز تفکر نوینی را برای شناخت موجودات ارگانیسم ابداع کرد این نظر برتالانی به نظریه عمومی سیستم ها شهرت یافت.

نگرش سیستمی: این نگرش چارچوبی منطقی و علمی ارائه می دهد که چند بعدی بوده و چارچوبی برای در نظر گرفتن عوامل محیطی، داخلی و خارجی سیستم به عنوان یک کل متشکل ارائه می دهد و به پدیده های اطراف بصورت یک کل به هم پیوسته می نگرد

ویژگی های سیستم: نظم و سازماندهی - رابطه متقابل اجزا با یکدیگر - وابستگی اجزا به یکدیگر - یکپارچگی و اتحاد اجزا - هدف مرکزی

عناصر یک سیستم: ۱. ورودی (Input) ۲. خروجی (Output) ۳. پردازش (process) ۴. کنترل (Control) ۵. محیط سیستم (Environment) ۶. بازخورد (Feedback)

تعریف ورودی (Input): منظور داده های ورود سیستم هستند.

تعریف خروجی (Output): نتیجه کار و محصول تولید شده توسط سیستم که درواقع هدف سیستم را می سازد.

تعریف پردازش (process): فرآیندی است که بر داده ورودی عمل می کند و خروجی مناسب تولید می کند.

تعریف کنترل (Control): امور نظارت بر سیستم را بر عهده دارد ونحوه کار سیستم را ارزیابی میکند

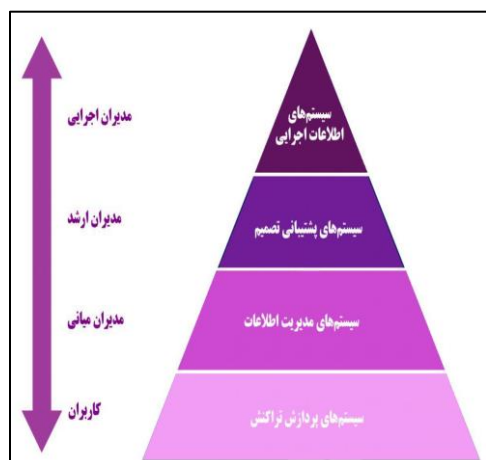
تعریف محیط سیستم (Environment): هر سیستم در محیطی قرار دارد و از عوامل محیط تاثیر می پذیرد و بر آنها اثر می گذارد . به این ترتیب، محیط هر سیستم را عواملی تشکیل می دهند که اگر چه جزء سیستم نیستند، اما تغییر در هر یک از آنها می تواند موجب تغییراتی در سیستم شود.

تعریف بازخورد (Feedback): بازخورد فرایندی دورانی است که در آن قسمتی از خروجی به عنوان اطلاعات به ورودی پس داده می شوند و به این ترتیب، سیستم را خود کنترل می سازد.

تعریف سیستم های اطلاعاتی (Information Systems): حجم زیادی از کار روزانه مدیران سازمان ها صرف اطلاعات می شود. از جمله دریافت اطلاعات، برقراری ارتباط و استفاده از اطلاعات در طیف وسیعی از امور مختلف. از آنجایی که اطلاعات مبنای تمام فعالیت های یک سازمان است، سیستم هایی باید وجود داشته باشند که اطلاعات را تولید و مدیریت کنند. هدف چنین سیستم هایی ایجاد تضمین در ارائه اطلاعات صحیح و قابل اطمینان در مواقع مورد نیاز و در شکل قابل استفاده است. چنین سیستم هایی، سیستم های اطلاعات نامگذاری شده اند. یک سامانه/سیستم اطلاعاتی مجموعه ای از مؤلفه های وابسته به هم است که با گردآوری، پردازش، ذخیره و توزیع داده های اطلاعاتی، از تصمیم گیری و کنترل در سازمان ها پشتیبانی می کند. این سیستم علاوه بر کمک کردن به ایجاد هماهنگی در انجام عملیات سازمانی، به مدیران و کارکنان سازمان ها کمک می کند تا مسائل سازمان را تحلیل یا شبیه سازی نمایند.

سه فعالیت عمده سیستم های اطلاعاتی: ۱. ورود داده ها ۲. پردازش ۳. خروج داده ها

* به عبارتی داده ها وارد سیستم اطلاعاتی می شوند، بر روی آن عملیات پردازش صورت می گیرد تا به اطلاعات در قالب های خاص تبدیل شوند و این اطلاعات در اختیار افراد مصرف کننده آن قرار می گیرد تا از آن استفاده نمایند. *



انواع سیستم های اطلاعاتی در یک سازمان :

۱. سیستم های پردازش تراکنش (Transaction Process Systems TPSs)

۲. سیستم های مدیریت اطلاعات (Management Information Systems MISs)

۳. سیستم های پشتیبانی تصمیم (Decision Support Systems DSSs)

۴. سیستم های اطلاعات اجرایی (Executive Information Systems EISs)

تعریف و مثال سیستم های پردازش تراکنش (Transaction Process Systems) TPS : سیستم هایی هستند که عملیات مهم روزانه سازمان را اجرا و ثبت می کنند. سیستم پایه ای است که سطح عملیاتی سازمان را پشتیبانی می کند. عملیاتی که قبلاً به صورت دستی انجام می شده اند با اهداف زیر مکانیزه می شوند: افزایش کارایی - کاهش زمان و افزایش سرعت - کاهش خطای انجام عملیات - کاهش هزینه های عملیاتی - ترکیب با سایر سیستم های اطلاعاتی و ...

اطلاعات ورودی: عملیات، رویدادها - پردازش: مرتب سازی، لیست کردن، ادغام، به روزرسانی - اطلاعات خروجی: گزارشات تفصیلی، لیست ها، خلاصه ها **کاربران:** کارکنان علمیاتی، سرپرستان

مثال : سیستم های حقوق و دستمزد - سیستم های فروش - سیستم های حضور و غیاب و ...

تعریف، کاربرد، ویژگی و هدف سیستم های سیستم های مدیریت اطلاعات (MIS) : سیستم های مدیریت اطلاعات (یا سیستم های اطلاعات مدیریت) وظیفه برنامه ریزی، کنترل و تصمیم گیری در سطح مدیران میانی را از طریق تامین گزارشات از عملکرد جاری و گذشته سازمان را به عهده دارند.

اطلاعات ورودی: داده های خلاصه عملیات، داده حجیم، مدل های ساده **پردازش:** گزارشات ثابت، مدل های ساده، تحلیل های ساده **اطلاعات خروجی:** گزارشات خلاصه و موارد خاص **کاربران:** مدیران رده میانی

کاربرد : در سطح مدیریت است. **هدف :** تهیه گزارش های خلاصه برای مدیریت است.

ویژگی : ۱. مبتنی بر اطلاعات داخل سازمان است. ۲. معمولاً داده های خود را از سیستم های TPS دریافت می کند. ۳. معمولاً گزارش ها بصورت هفتگی و ماهانه هستند. ۴. مبتنی بر داده های گذشته و حال حاضر است. ۵. مرتبط بر تصمیمات ساخت یافته و نیمه ساخت یافته است. ۶. جنس گزارش ها از نوع کنترلی است.

تعریف و ویژگی سیستم های پشتیبانی تصمیم (DSS Decision Support Systems) (بامثال) : این سیستم ها با استفاده از داده ها و ابزارهای تحلیلی به مدیران برای تصمیم گیری های غیرساخت یافته کمک می کنند.

اطلاعات ورودی: داده های خلاصه شده و با تفصیلی و ابزارهای تحلیلی **پردازش:** تعاملی، شبیه سازی، تجزیه و تحلیل های آماری
اطلاعات خروجی: گزارشات خاص، تحلیل تصمیم ها، پاسخ به پرس و جو **کاربران:** متخصصین، مدیران سطوح میانی

علاوه بر داده ها و اطلاعات داخل (MIS و TPS) ممکن است از خارج سازمان نیز اطلاعاتی را مورد استفاده قرار دهد - در این سیستم ها از مدل های کمی و کیفی تصمیم گیری و تحلیل استفاده می شود- در این سیستم ها ارتباط متقابلی با کاربر برقرار است به نحوی که کاربر می تواند اطلاعات و فرضیات تصمیم گیری را تغییر دهد، سوالات جدید پرسد و یا داده های جدیدی وارد نماید.

مثال : سیستم پشتیبان تصمیم سفرهای دریایی ؛ یک شرکت بزرگ آمریکایی تعدادی کشتی برای حمل کالا دارد. یک سیستم DSS برای این شرکت می تواند به سوالات زیر پاسخ دهد:

- با وارد کردن برنامه زمانبندی ارسال محمولات یک کشتی و نرخ حمل آن، سیستم مشخص می کند تخصیص این محموله ها به کدامیک از کشتی ها بهینه است. - چه سرعتی از کشتی ها بهینه است در صورتیکه بخواهیم هزینه سوخت کمینه شده و زمانبندی حمل محمولات نیز محقق شود. - مسیر حرکتی توقفات کشتی ها بصورت بهینه کدام است.

تعریف، ویژگی سیستم های اطلاعات اجرایی (Executive Information Systems) یا Executive Support Systems (Systems): به آن ها سیستم های پشتیبان مدیران ارشد هم گفته می شود. برای کمک به تصمیم گیری های غیر ساخت یافته و استراتژیک مدیران ارشد بکار می رود. از قابلیت های گرافیکی و ارتباطی استفاده می کند.

اطلاعات ورودی: انبوهی از داده های داخلی و خارجی **پردازش:** ترسیم، شبیه سازی، تعامل اطلاعات **خروجی:** طرح ها، پاسخ به پرس و جو **کاربران:** مدیران ارشد

ویژگی : EIS سطوح استراتژیک سازمان را پشتیبانی می کند - در EIS اطلاعات داخل و خارج سازمان یکپارچه شده و مورد استفاده قرار می گیرد- در آن ها اطلاعات کلی سازمان و محیط خارجی، خلاصه شده و در موارد بحرانی پیگیری می شوند- در این سیستم ها معمولاً نرم افزارهای گرافیکی و تحلیلی استفاده می شوند و اطلاعات پیش بینی برای چند سال آینده جهت تصمیمات استراتژیک سازمان تهیه می شود. - جهت اخذ تصمیمات ساختاریافته بکار می رود- طراحی آن برای هر سازمان منحصر به فرد است- مدیریت ارشد را به سطوح سازمان مرتبط و مسلط می سازد- طراحی، پیاده سازی و نگهداری آن بسیار گران است - نیاز به پشتیبانی گسترده تخصصی و کارشناسی دارد.

* EIS معمولاً می تواند به سوالات زیر پاسخ دهد: -رقبا به چه سمتی می روند؟-نیازهای آینده بازار چیست؟ - تکنولوژی پیشرفته صنعت چیست؟ *

تجزیه و تحلیل نیازمندی ها، مهمترین و اساسی ترین مرحله در روند چرخه حیات تولید نرم افزار محسوب می گردد. (صرف نظر از مدل چرخه)

ویژگی تجزیه و تحلیل : این امر توسط کارشناسان ارشد و اعضای متبحر تیم با استناد به داده های ورودی مشتریان، بخش فروش و بررسی های بازار و متخصصین این حوزه از صنعت انجام می گیرد. و پس از این مرحله، اطلاعات گردآوری شده، جهت طراحی اساس و پایه پروژه و مطالعه امکان سنجی آن در حوزه های اقتصادی، عملیاتی و فنی مورد استفاده قرار می گیرد و همچنین شناسائی ریسک ها و خطرات بالقوه پروژه نیز در مرحله برنامه ریزی و تحلیل انجام می گیرد. نتیجه مطالعات و امکان سنجی ها موجب بدست آوردن دیدگاه های مختلف فنی خواهد بود که می تواند باعث پیاده سازی موفقیت آمیز پروژه با حداقل ریسک گردد.

فعالیت های تجزیه و تحلیل: ۱. مطالعه عملیات فعلی سازمان و سیستم های اطلاعاتی مورد استفاده ۲. تعیین ملزومات و تشخیص خواسته کاربران از سیستم ۳. مطالعه ملزومات و سازماندهی آنها (مطالعه ساختار ملزومات مطابق با ارتباطات داخلی و حذف افزونگی ها و جمع آوری داده ها و تجزیه و تحلیل آن ها برای اینکه بفهمیم: ۱. سیستم در چه وضعیتی است؟ ۲. سیستم چه کارهایی باید انجام دهد؟ ۳. چه چیزهایی باید به سیستم اضافه یا کم شود؟ ۴. نیازهای استفاده کنندگان چیست؟

وظایف تحلیل گر: ۱. نیاز سنجی و امکان سنجی پروژه ۲. تحلیل و مستندسازی عملیات سیستم موجود ۳. تحلیل و شناخت مسائل و راه حل های آنها برای سیستم موجود ۴. تعیین نیازمندی ها جهت توسعه، بهبود و یا جایگزینی سیستم موجود ۵. ارزیابی راه حل ها و پیشنهادات متفاوت برای انجام پروژه ۶. ارزیابی امکانات سخت افزاری و نرم افزاری موجود برای پیاده سازی سیستم ۷. تعیین بودجه، افراد و زمان بندی پروژه ۸. طراحی صفحات رابط کاربری و روال های برنامه نویسی ۹. طراحی ساختار فایل ها و بانک های اطلاعاتی ۱۰. تهیه تست برای آزمون برنامه ها ۱۱. نظارت بر پیاده سازی و نصب سیستم

مهارت های لازم برای یک تحلیلگر موفق: ۱. مهارت های تحلیلی ۲. مهارت های فنی ۳. مهارت های مدیریتی

مهارت های تحلیلی: - فکر سیستمی (مثال آفت کشاورزی) - دانش سازمانی (آشنایی با نحوه کار بخش های مختلف یک سازمان، عملکرد و هدف هر یک، رابطه بخش ها با یکدیگر، ارتباط با مشتری و تولید کنندگان) - تجزیه و تحلیل و حل مسئله و ...

مهارت های فنی: آشنایی با تکنولوژی های متفاوت در مورد: ۱. نحوه کار میکرو کامپیوترها، ایستگاههای کاری، مینی کامپیوترها، کامپیوترهای بزرگ و سوپر کامپیوترها ۲. انواع زبانهای برنامه نویسی ۳. سیستم های مدیریت فایل و پایگاه های داده ۴. ارتباط شبکه های محلی و گسترده ۵. محیط ها و ابزار توسعه سیستم ها (نظیر تولید کننده های گزارش ها و فرم ها، رابط گرافیکی و ابزار طراحی و ...) ۶. ابزار تحلیل داده

مهارت های مدیریتی: مدیریت منابع (Resource management) (تکنولوژی، پول، افراد، اسناد) شامل این توانایی ها است: ۱. پیش بینی برنامه ریزی استفاده از منابع (بودجه بندی) ۲. پیگیری و حسابرسی منابع ۳. یادگیری چگونگی استفاده موثر از منابع ۴. ارزیابی کیفیت منابع استفاده شده و حفاظت منابع از استفاده نابجا (نا مناسب) ۵. متوقف کردن استفاده از منابع وقتی که نیازی به آنها نیست.

روش های جمع آوری اطلاعات : ۱. از طریق پرسیدن، مانند مصاحبه کردن، پرسشنامه و ابزارهای الکترونیکی مانند ایمیل و شبکه های اجتماعی، استفاده از بانک اطلاعاتی و... ۲. مطالعات دیداری: مانند شرکت کردن در محیط کار کارکنان ۳. کارگاه های ساخت یافته و بحث گروهی: مانند کنفرانس ها (Meeting).

صرف نظر از روش های جمع آوری اطلاعات جهت رفع نیازهای کاربران منابع اطلاعاتی زیادی در مورد سیستم وجود دارند. معمولاً هر منبع، اطلاعات خاصی را در اختیار دارد و برای دریافت اطلاعات از آنها روش های مختلفی استفاده می شود.

منابع اطلاعاتی جهت تحلیل خواسته های کاربران و شناخت نیازها : ۱. کاربران سیستم ۲. فرم ها و مستندات ۳. برنامه های کامپیوتری ۴. گزارش ها

منابع اطلاعاتی: ۱. کاربران سیستم ۲. فرم ها و مستندات ۳. برنامه های کامپیوتری ۴. گزارش ها (report)

۱- تعریف کاربران سیستم: مهمترین منبع اطلاعات کاربران سیستم هستند که می توان خواسته های کاربران را از این طریق بدست آورد. راههای مختلفی برای این کار وجود دارد: **(الف) مصاحبه (View Inter ب) تهیه پرسش نامه ج) مشاهده رفتار کاربران**

(الف) تعریف مصاحبه (View Inter) : یکی از بهترین و متداول ترین تکنیک های جمع آوری اطلاعات و تحلیل مصاحبه است. مصاحبه معمولاً با مدیریت شروع می شود و اینکه همه افراد را باید در نظر گرفت (از مدیران تا کاربران جزء). مصاحبه ابزاری برای تعیین خواسته ها است، بنابراین بایستی با کاربران تماس گرفت و مشکلات و اولویت های آنها را مورد بررسی قرار داد. مصاحبه بهترین روش برای تحلیل سیستم های ساخت یافته بزرگ است.

عوامل موفقیت مصاحبه : ۱. افرادی که برای مصاحبه انتخاب می شوند: تحلیل گر باید مطمئن باشد که تمام افراد مورد مطالعه در نظر گرفته شده اند. ۲. روش درست برای انجام مصاحبه: مصاحبه گر باید روابط دوستانه ای را بین مصاحبه کنندگان برقرار کند تا بتواند حقایق مربوط را کشف کند.

ویژگی مصاحبه : در سیستم های بزرگ مصاحبه تا حدی مشکل است. : اطلاعات باید به روش سازمان یافته ای جمع آوری شود تا تضمین شود که تمام جزئیات سیستم مورد بررسی قرار گرفته اند. باید با تمام کاربران مشورت شود تا تضمین گردد که تمام مسئله های سیستم مشخص شده اند.

راهبرد جستجو در مصاحبه: ۱. قبل از تحلیل سیستم تحلیل گر بایستی راهبرد جستجو را تعیین کند. ۲. این راهبرد جستجو با انتخاب منابعی که اطلاعات باید از آنها بدست آید و تعیین روشهای جمع آوری : اطلاعات از هر منبع، مشخص می گردد. ۳. این منابع و روشها در یک رویه جستجو ترکیب می شوند. ۴. رویه جستجو مشخص می کند که جستجو از کجا شروع شود و چگونه ادامه یابد. ۵. در اغلب رویه های جستجوی مصاحبه لازم است تمام منابع اطلاعاتی بررسی شوند تا همه اطلاعات مورد نیاز سیستم جمع آوری گردد. ۶. اطلاعات باید به روش بالا به پایین جمع آوری گردد تا مدل سیستم به تدریج ساخته شود و در غیر اینصورت مدیریت کار مشکل است. (بالا به

پایین : از مدیران سطوح بالا شروع می شود و درباره هزینه ها ، مدت زمان تحویل سیستم ، اطلاعات کلان و... بحث می شود . و تا کاربران و اپراتور ها ادامه می یابد).

مزایا مصاحبه: ۱. بدلیل اینکه پاره ای از مردم به ایجاد ارتباطهای شفاهی علاقه دارند و در صورتی که مصاحبه گر در زمینه مصاحبه مهارت کافی داشته باشد، به سادگی می تواند با فراهم نمودن شرایطی که لازمه مصاحبه است اطلاعات مناسبی را کسب کرد. ۲. حضور داشتن مصاحبه گر در موقعیت مصاحبه سبب می شود که مصاحبه شونده منظور و هدف او را از طرح سؤال های مصاحبه بهتر درک کند و متناسب با سؤال پاسخ های دقیق تری را ارائه نماید. ۳. امکان دریافت بازخورد سریع و صریح را داراست . ۴. می توان با ایجاد رابطه عاطفی و ایجاد انگیزه های مناسب اطلاعات محرمانه و ارزشمندی را بدست آورد . ۵. در مورد افراد بیسواد ، کودکان ، بیماران روانی و افرادی که تمایل به نوشتن ندارند مناسب است.

معایب مصاحبه: ۱. مصاحبه وقت گیر است. ۲. تعبیر و تفسیر و تجزیه در مورد اطلاعاتی که از مصاحبه گرفته می شود بسیار مشکل است. ۳. برای انجام مصاحبه به افراد مجرب و ماهر نیاز است تا بتوان اطلاعات را طبقه بندی شده و قابل استفاده جمع آوری نمود. ۴. تنها انجام مصاحبه کافی نیست. آنالیزگر باید بر روی نتایج مصاحبه کار کند و معمولاً در قالب تهیه فرم های ورودی خروجی و منوها باشد و آنالیزگر سعی می کند محیط کاری آتی را به کاربر نشان دهد. ۵. نسبت به پرسشنامه مخصوصاً بی نام اطلاعات کمتری بدست می آید.

ب) تعریف تهیه پرسشنامه : شامل مجموعه ای از سوالات هستند که توسط مشاورین و کسانی که با اهداف و عملکرد دستی سیستم ها آشنایی دارند تهیه می شود و در یک سازمان پخش می شود و جواب ها مطالعه می گردد.

انواع پرسشنامه از لحاظ اجرا: ۱. پرسشنامه حضوری ۲. پرسشنامه تلفنی ۳. پرسشنامه پستی ۴. پرسشنامه ایمیلی با شبکه های اجتماعی ۵. پرسشنامه اینترنتی یا صفحات وب

مزایا پرسش نامه: ۱. با استفاده از پرسش نامه ها میتوان تعداد افراد زیادی را در بر گرفت. ۲. تحلیل آن آسان تر است. ۳. قابلیت مقایسه نتایج با یکدیگر وجود دارد.

معایب پرسش نامه: ۱. تهیه پرسش نامه نیاز به تجربه زیاد دارد. ۲. عدم دریافت به موقع پاسخ از افراد. ۳. بدون پاسخ گذاشتن برخی از سوالات توسط افراد. ۴. تفاوت در فهم سوالات ۵. دشوار بودن انتقال احساس

ج) تعریف مشاهده رفتار کاربران : با مشاهده چگونگی انجام وظایف شاید بتوان بهتر از مصاحبه بر روی سیستم و یا نیازهای آن شناخت پیدا کرد . باید دقت کنید که معمولاً مشاهده بسیار طولانی می شود و باید هدف را قبلاً مشخص کرد . معمولاً مشاهده در ارتباط با جنبه های مختلف انجام یک کار می باشد. شامل مدت زمان انجام وظیفه توسط فرد و تعداد خطاهایی که فرد در انجام کار خود مرتکب می شود و ... فاکتورهایی که در ارتقاء کارایی کار افراد مهم هستند باید مشخص شوند.

مزایا مشاهده رفتار کاربران: ۱. آشنایی به روشی که سیستم جاری کار می کند. ۲. میزان کارایی سیستم جاری را می توان مشخص نمود.

معایب مشاهده رفتار کاربران: ۱. مشاهده کننده باید فردی با تجربه باشد تا هر چیزی را مورد تأکید قرار ندهد. ۲. افراد هنگامی که احساس می کنند تحت نظر هستند، طبق روال عادی خود عمل نمی کنند. ۳. ممکن است بسیار طولانی و زمانگیر باشد.

۲- تعریف فرم ها و مستندات : معمولا با ارجاع به فرم ها و گزارشاتی که افراد تهیه می کنند و بخصوص در هنگام مصاحبه می توان بخش اعظمی از نیازها را مشخص کرد. و با ارجاع به این فرم ها میزان داده ها و فرمت داده ها و نهایتا ساختار بانک های اطلاعاتی و گزارشاتی که باید از آن استخراج شود مشخص می گردد.

مزایای استفاده از فرم ها و مستندات : ۱. حجم داده ها و اطلاعات مورد پردازش مشخص می گردد. ۲. فرمت داده ها تعیین میشود. ۳. مشکلات و خطاهایی که به واسطه پر کردن دستی مستندات ایجاد می شود تعیین می گردد. ۴. اگر سیستم کلا تغییر پیدا کند ضرر زیادی متحمل نخواهیم شد (صرفا مبادرت به گردآوری اطلاعاتی کرده ایم که ممکن است در آینده نیازی به آن نباشد).

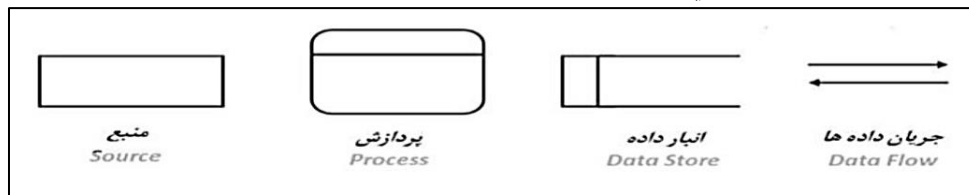
۳- تعریف برنامه های کامپیوتری : برنامه های کامپیوتری می توانند برای تعیین جزئیات ساختمان داده ها (Data Structure) بکار روند. مستندات برنامه ها باید خوانده شوند، برنامه ها باید با داده های آزمایشی اجرا گردند و واسط کاربران مورد بررسی قرار گیرد.

۴- تعریف گزارش ها (Report) : این منبع انواع خروجی های مورد نیاز کاربران را نشان می دهد و می تواند بعنوان مبنایی برای مصاحبه با کاربران بکار رود تا گزارش های جدید شناسایی شوند.

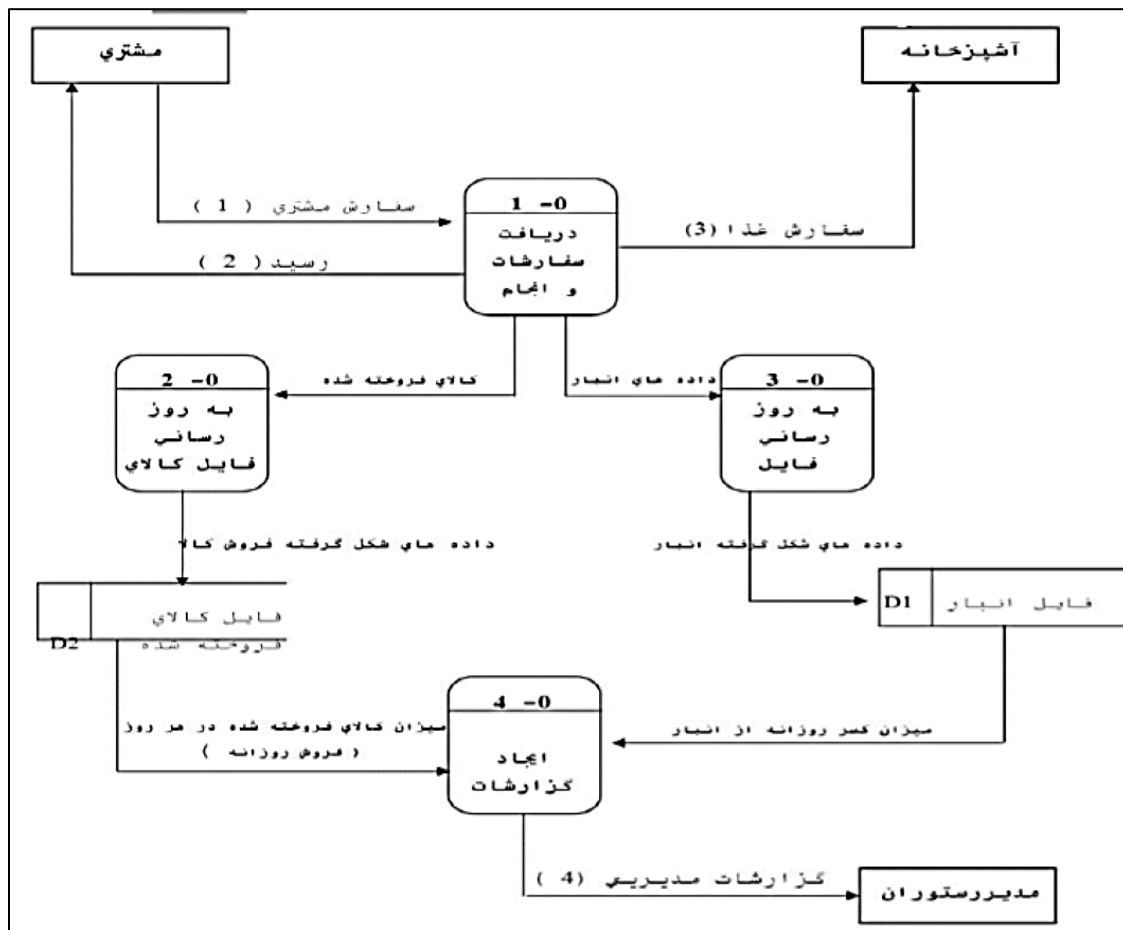
تعریف مدل سازی : جهت به تصویری کشیدن سیستم، مدل سازی یک روش ساختاری است که به کمک آن می توان سیستم را از بالا به پایین (ازکل به جزء) تجزیه و تحلیل نمود.

تعریف و ویژگی نمودار جریان داده DFD (Data flow diagram) : نمودار جریان داده ها به صورت نمایی نقشه مانند و جهت دار برای نمایش جریان داده های ورودی و خروجی با هدف درک بهتر سیستم نرم افزاری مورد نظر در چند سطح طراحی و پیاده سازی می گردد. در این نمودار به صورت کامل مشخص خواهد بود که داده ها از چه بخش وارد سیستم می شوند، در کدام قسمت ها فرایند تعریف شده بر روی آنها اعمال می شوند، از کدام بخش ها عبور می کنند، و در نهایت در کدام بخش ذخیره می شوند. ابزاری است که به شما اجازه می دهد چگونگی جریان داده را، در یک سیستم اطلاعاتی مدل سازی نمایید. از آنجایی که جریان داده روی جابجایی داده بین پردازش ها تمرکز دارد نمودار پردازش نیز نامیده می شود.

نمایش سیستم به شکل DFD : بطور کلی از چهار سمبل زیر استفاده میشود:

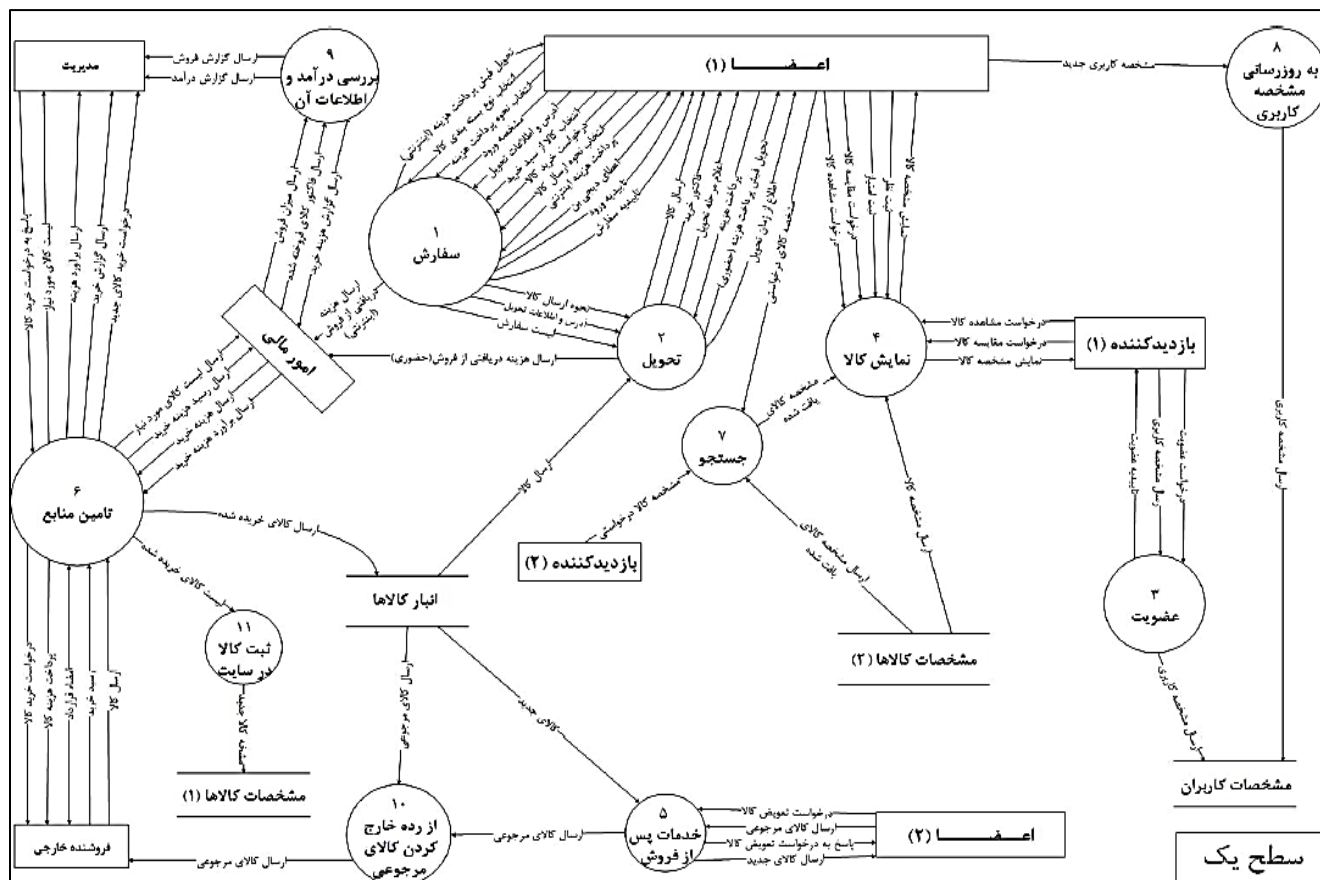
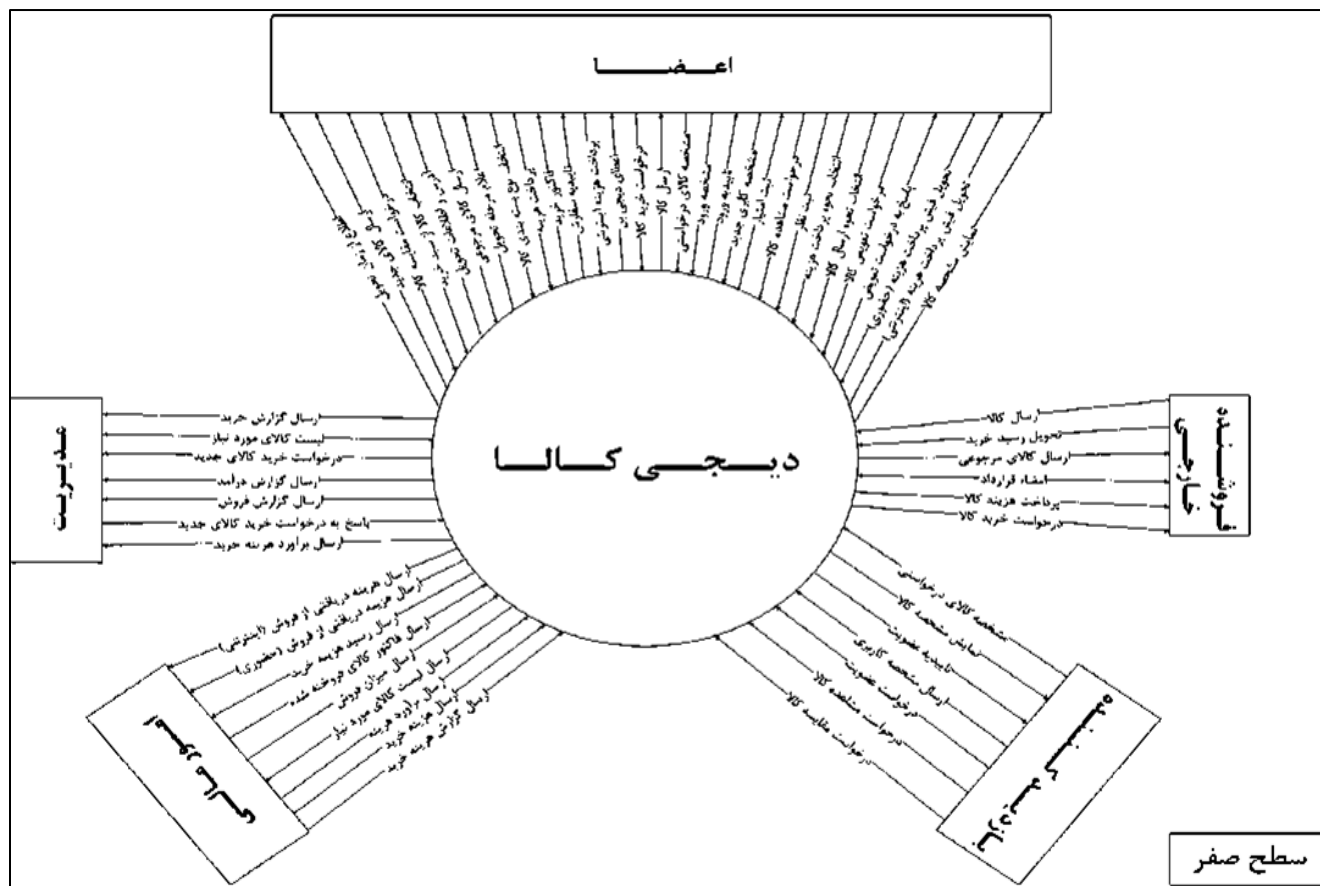


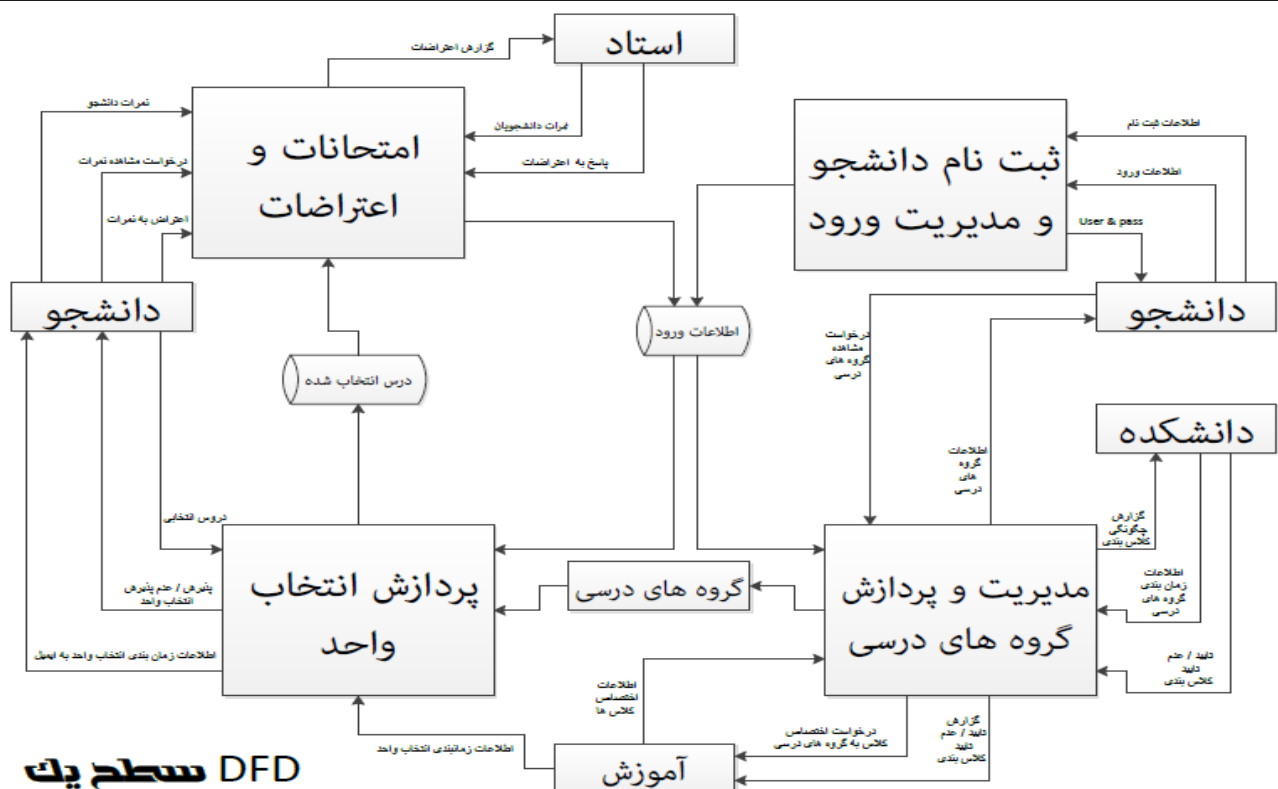
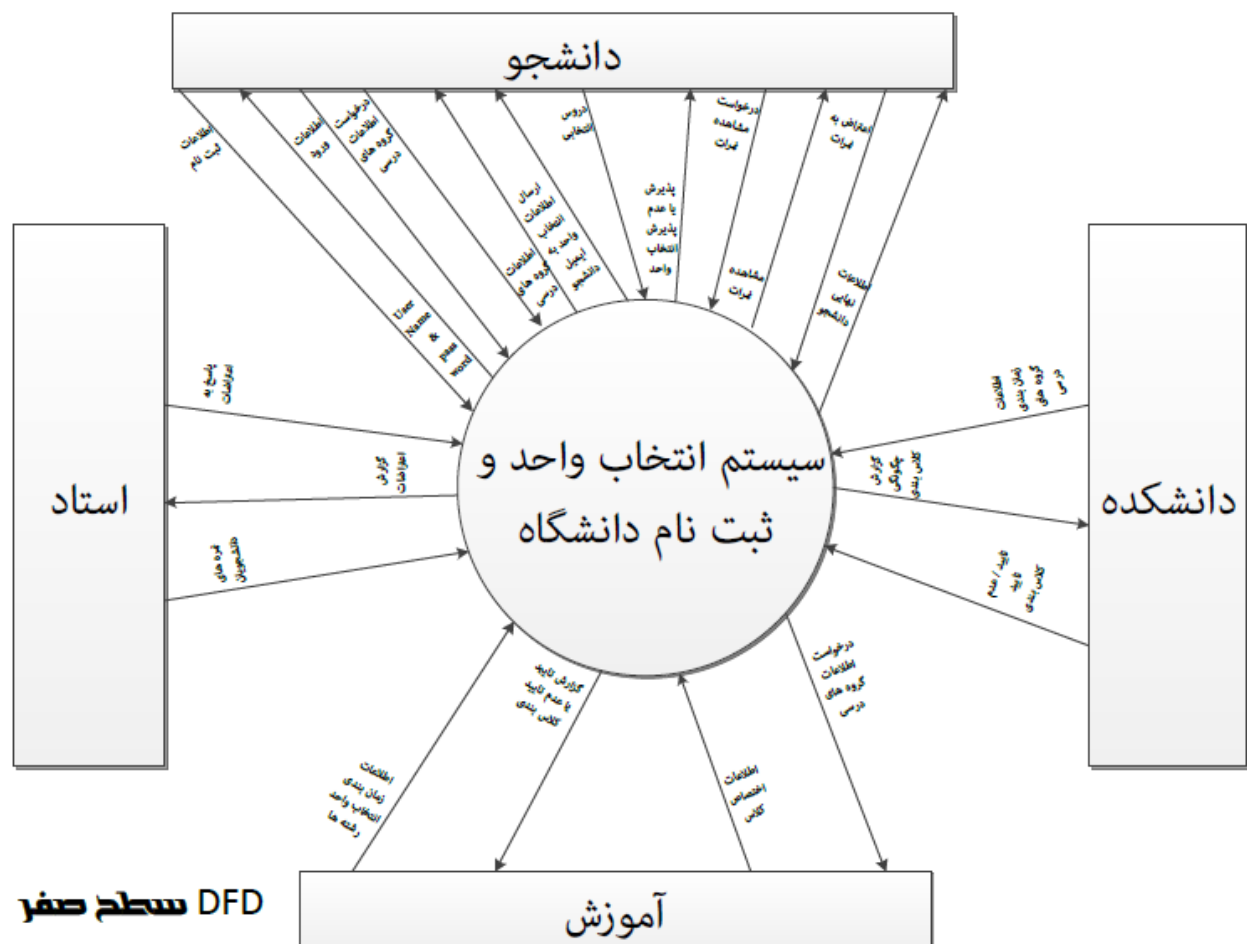
چگونگی تهیه DFD ها (با مثال) : فرضاً مدل منطقی یک سیستم سفارش غذا مد نظر است. بالاترین سطح مشاهده این سیستم، دیاگرام سطح صفر یا دیاگرام متن (Context Diagram) است که در شکل زیر نشان داده شده است:



* دیاگرام سطح صفر بیانگر پردازش های اصلی سیستم، جریان داده و مخازن داده در سطح بالایی از جزئیات می باشد که در بسیاری از اوقات بدون نشان دادن مخازن داده می باشد. و هر DFD سطح n نشان دهنده یک پروسه از DFD سطح n+1 است. *

نمونه های DFD سطح صفر و سطح یک :



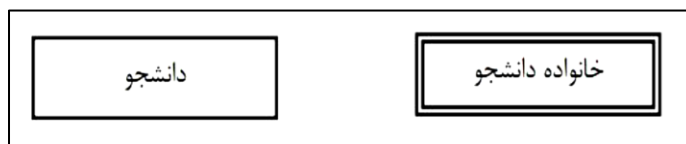


تعریف نمودار رابطه ای ER یا ERD (Entity Relation Diagram): نمودار ER اولین بار در سال ۱۹۷۶ جهت طراحی بانک های اطلاعاتی معرفی شد. یکی از چالش هایی که در آن زمان وجود داشت بیان ساده ای از بانک های اطلاعاتی به دور مدل خاصی بود. این مدل به دلیل سادگی و قابلیت فهم بالا، اهمیت فراوانی پیدا کرد و به تدریج استفاده از آن گسترش یافت.

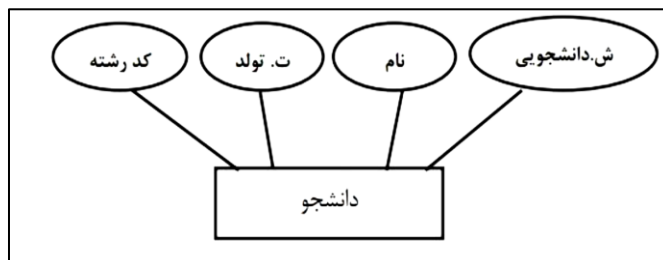
اجزا نمودار ER: موجودیت یا پدیده - صفت - ارتباط

تعریف موجودیت یا پدیده (اجزا نمودار ER): موجودیت عبارت از پدیده هایی است که در بانک اطلاعاتی وجود خارجی دارد. برای مثال پدیده هایی همچون استاد، دانشجو، درس، دانشکده از جمله موجودیت های هستند که در بانک اطلاعاتی دانشگاه وجود دارند.

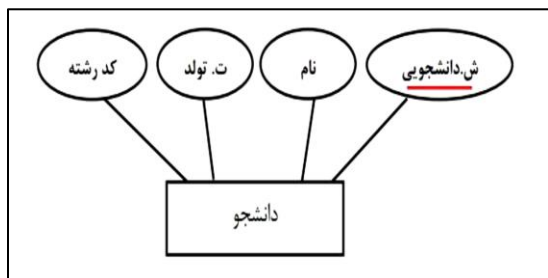
تعریف انواع موجودیت یا پدیده: ۱. **مستقل:** موجودیت هایی هستند که مستقل از هر موجودیت دیگر هستند و به خودی خود معنی خواهند داشت. ۲. **وابسته:** موجودیت هایی هستند که وجودشان وابسته به یک نوع موجودیت دیگر است. برای مثال فرض کنید در پایگاه داده ای می خواهیم اطلاعات دانشجو و اطلاعات خانواده او را ذخیره نماییم. در اینجا دانشجو یک موجودیت مستقل و خانواده دانشجو که وابسته به وجود دانشجو است یک موجودیت وابسته به شمار می آید. موجودیت های مستقل و وابسته را با شکل های زیر نمایش می دهند.



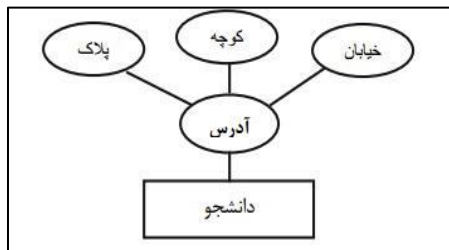
تعریف صفت (اجزا نمودار ER): هر موجودیت دارای ویژگی هایی است که آن را نسبت به موجودیت های دیگر متمایز می سازد، به این ویژگی ها صفت می گویند. برای مثال برای موجودیت دانشجو می توانیم صفت هایی همچون شماره دانشجویی، نام، تاریخ تولد و کد رشته تعریف نماییم.



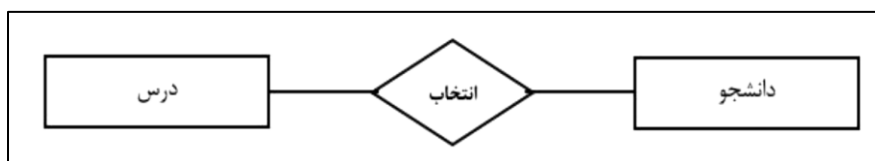
تعریف انواع صفت ها: ۱. **صفت کلید:** صفت کلید عبارت است از یک یا چند صفت از هر موجودیت که مقادیر آن منحصر به فرد است. برای مثال موجودیت دانشجو را در نظر بگیرید صفت شماره دانشجویی برای این موجودیت مقادیر منحصر به فردی خواهد داشت پس می تواند صفت کلید باشد اما صفت کد رشته نمی تواند کلید باشد؛ زیرا ممکن است چند دانشجو در یک رشته در حال تحصیل باشند. صفت کلید را به شکل زیر نمایش می دهند:



۲. **صفت ساده و مرکب:** برخي از صفاتي که در بانک اطلاعاتي داريم ممکن است از تشکيل چند صفت کوچکتر به وجود آمده باشند، به چنين صفاتي که هم خودشان به تنهائي و هم بخش هاي آن ها معني دار هستند صفات مرکب گوييم. براي مثال صفت آدرس یک صفت مرکب است زیرا آدرس به بخش هاي کوچکتری مانند خيابان کوچه و پلاک تقسيم مي شود. اکثر صفات موجود در بانک اطلاعاتي تجزيه پذير نيستند. به صفاتي که از بخش هاي کوچکتری تشکيل نشده باشند صفات ساده گوييم.



تعريف ارتباط (اجزا نمودار ER): به تعامل میان دو يا چند موجوديت مختلف با یکدیگر ارتباط مي گويند. براي مثال موجوديت هاي درس و دانشجو با یکدیگر ارتباط دارند.

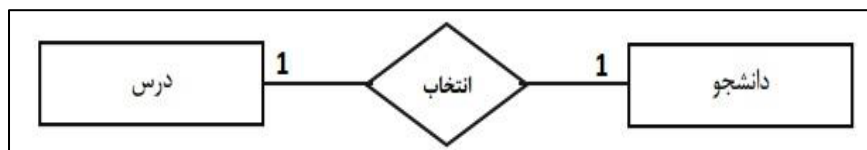


خصوصيات هر ارتباط : ۱. هر ارتباط یک نام دارد. ۲. هر ارتباط معنای مشخصي دارد که آن ارتباط را از ساير ارتباطات متمایز مي سازد. ۳. هر ارتباط نمونه هايي دارد.

* ارتباط نیز مي تواند خود داراي صفاتي باشد.

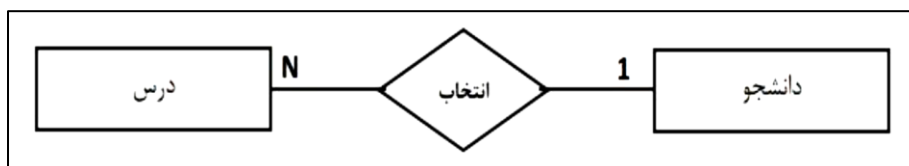
انواع ارتباط : چندي (ماهيت) ارتباط بيانگر چگونگي ارتباط میان مجموعه نمونه هاي دو موجوديت مختلف است. به طور کلي سه نوع چندي داريم: ۱. یک به یک ۱:۱ ۲. یک به چند ۱:N ۳. چند به چند N:M

مثال از انواع ارتباطات : ۱. یک به یک ۱:۱ :



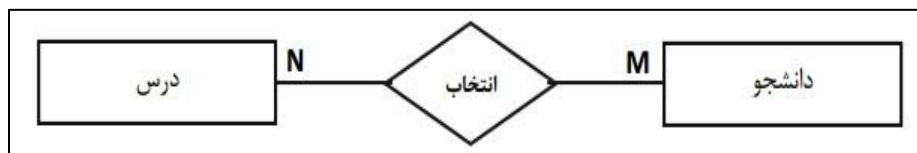
در مثال فوق ارتباط بين دانشجو و درس یک به یک مي باشد يعني هر دانشجو مي تواند یک درس را انتخاب نمايد و هر درس نیز مي تواند توسط یک دانشجو انتخاب شود.

۲. یک به چند ۱:N :



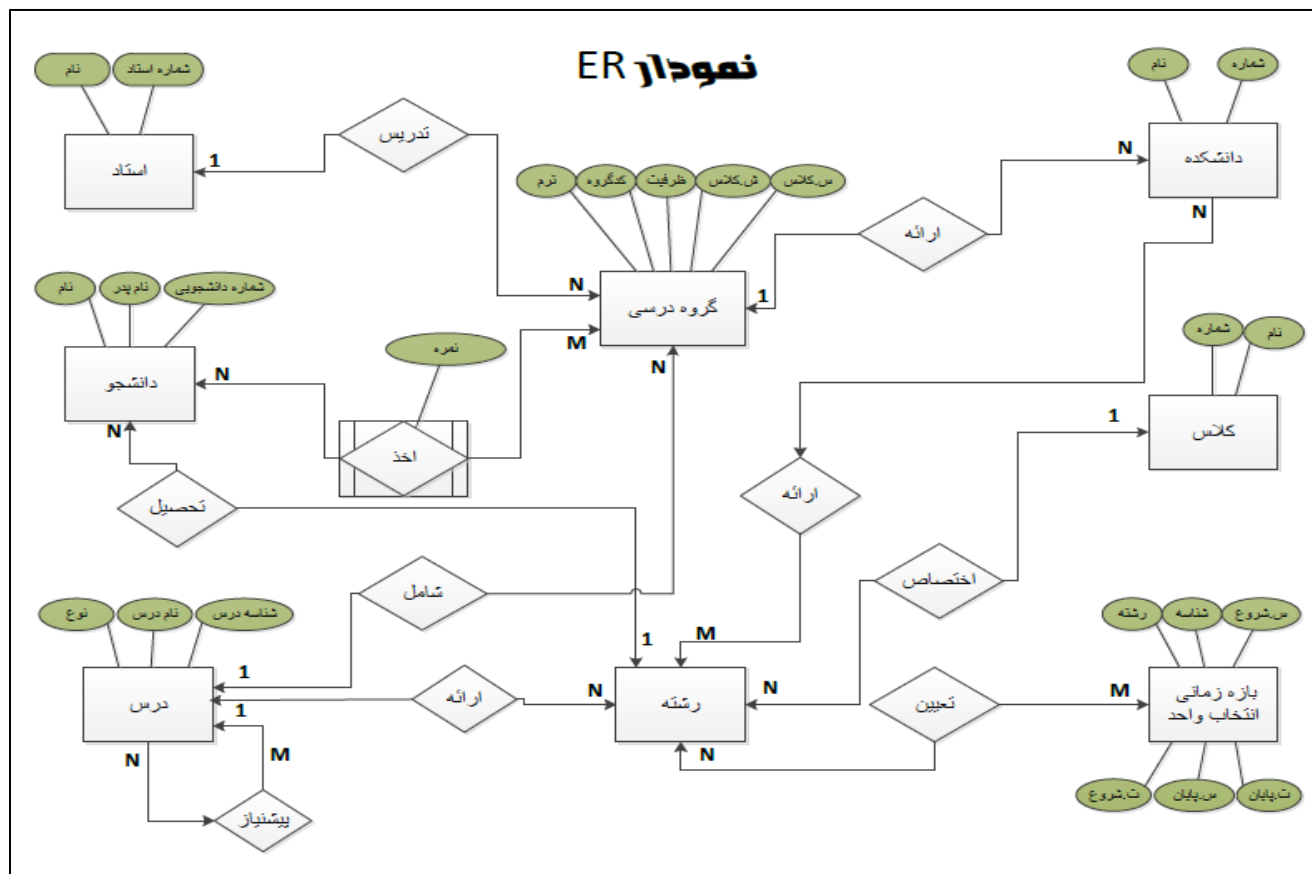
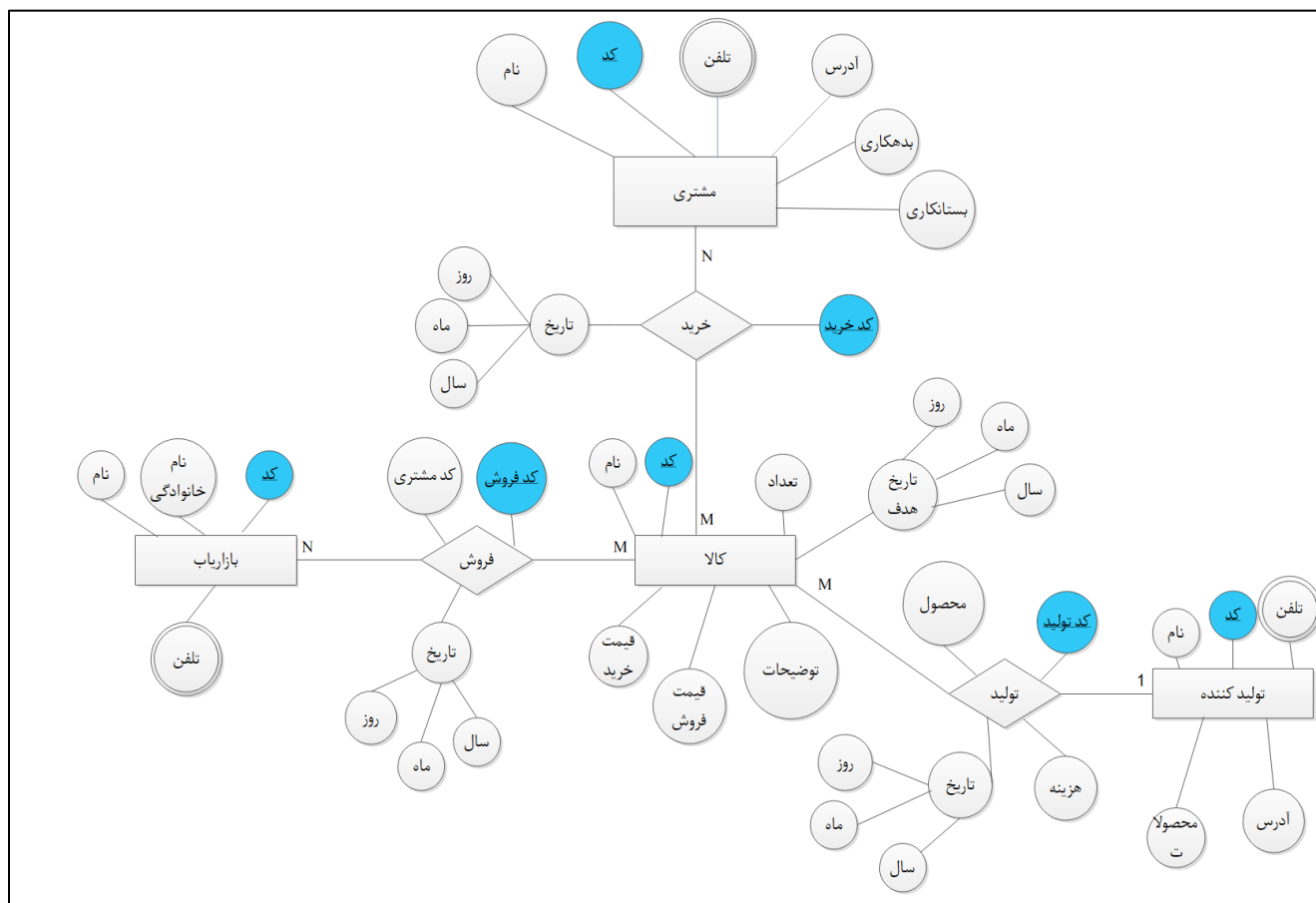
در بالا یک ارتباط یک به چند را مشاهده می کنیم با توجه به این چندی هر دانشجو می تواند چند درس را انتخاب نماید و هر درس می تواند فقط توسط یک دانشجو انتخاب شود.

۳. چند به چند N M :



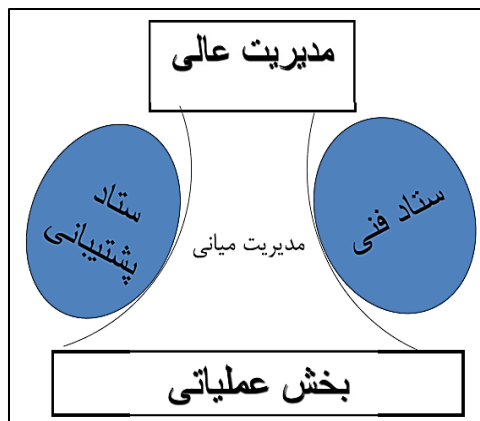
با توجه به ماهیت چند به چند رابطه بالا، هر دانشجو می تواند چند درس را انتخاب نماید و در مقابل هر درس می تواند توسط چند دانشجو انتخاب شود.

نمونه ای کامل از نمودار ER :



تعریف سازمان : سازمان عبارت است از سیستمی متشکل از اجزاء به هم پیوسته و مبتنی بر نظم و انضباط که در جهت رسیدن به هدف های خاصی فعالیت می کند .

ساختار کلی سازمان از دیدگاه هنری مینتزبرگ : سازمان دارای پنج بخش اساسی است : ۱- بخش مدیریت عالی : (مسئولیت نهایی اداره امور سازمان را دارد) ۲- بخش میانی : (نظارت بر کار واحدها و هماهنگی فعالیت آنها را بر عهده دارد) ۳- بخش عملیاتی : (مسئولیت تولید کالاها و خدمات را دارد) ۴- بخش فنی : (از متخصصان فنی تشکیل و در موارد لزوم نظرات فنی و تخصصی می دهد) ۵- ستاد پشتیبانی : (نقش کمک و یاری دهنده را برای سایر واحدها ایفاء می کند .



مبنای سازماندهی : ۱- سازمان بر مبنای تعداد ۲- سازمان بر مبنای وظیفه (هدف) ۳- سازمان بر مبنای نوع عملیات (تخصص) ۴- سازمان بر مبنای مشتری (ارباب رجوع) ۵- سازمان بر مبنای قلمرو عملیاتی (محل جغرافیایی) ۶- سازمان بر مبنای محصول (نوع تولید) ۷- سازمان بر مبنای پروژه ۸- سازمان ماتریسی ۹- سازمان با ساخت آزاد ۱۰- ساخت سازمانی مدولار ۱۱- ساختهای ترکیبی

تعریف سازمان بر مبنای تعداد : در این سازماندهی ، گروهبندی و تقسیم وظایف بر مبنای تعداد صورت می گیرد و شامل تقسیم بندی افراد بطور اتفاقی و به دسته های مساوی و مأمور کردن هر دسته برای انجام یک قسمت از وظایف سازمانی است .

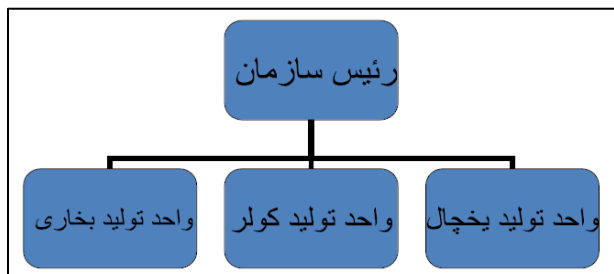
تعریف سازمان بر مبنای وظیفه (هدف) : در این نوع سازماندهی ، واحدها بر حسب وظیفه و کاری که قرار است انجام دهند تقسیم بندی می شوند در این روش هدف اصلی سازمان به فعالیت های اصلی و فعالیت های اصلی به فعالیت های فرعی و فعالیت های فرعی به وظایف اصلی و وظایف اصلی به وظایف فرعی تقسیم می شوند .

تعریف سازمان بر مبنای نوع عملیات (تخصص) : در این روش مبنای تقسیم کار نوع فعالیت ، تخصص و آگاهیها و حرفه افرادی است که کارها را انجام می دهند .

تعریف سازمان بر مبنای مشتری (ارباب رجوع) : در این سازماندهی به دریافت کنندگان خدمات توجه گردیده و سازماندهی بر اساس گروههای مختلف مشتریان و ارباب رجوع صورت می پذیرد .

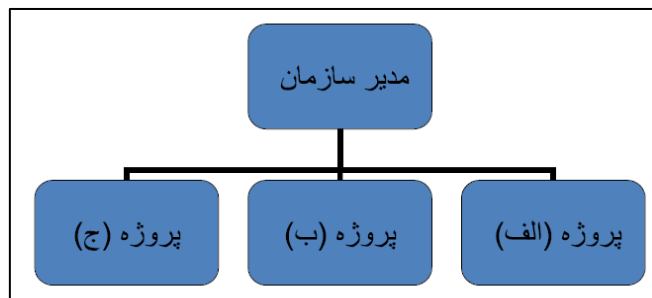
تعریف سازمان بر مبنای نوع محصول (تولید) : در این روش مبنای گروه بندی وظایف و تقسیم کارها نوع کالا یا محصولی

است که قرار است تولید شود .



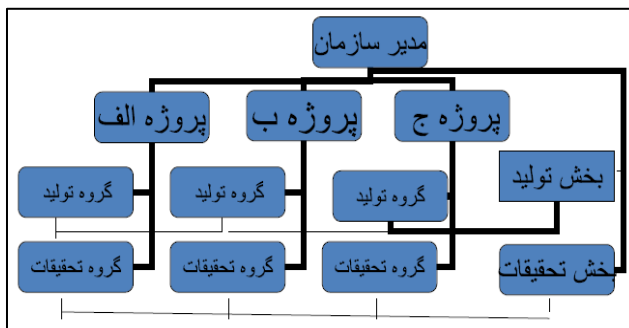
تعریف سازمان بر مبنای پروژه : در سازمانهایی که هدف آنها در قالب پروژه ها و برنامه های تقریباً مستقل اجرا می شود ، می توان

به تعداد پروژه های موجود ، واحد های مستقلی را ایجاد نمود.



تعریف سازمان ماتریسی : این سازمان تلفیقی از سازمان بر مبنای وظیفه و سازمان بر مبنای پروژه است بدین ترتیب که جریان اختیار

در واحدهای تخصصی بصورت عمودی و در واحدهای اجرایی بصورت افقی است و از تلاقی این دو محور یک ماتریس بوجود می آید.



تعریف سازمان با ساخت آزاد یا ادھوکراسی : این نوع سازماندهی ساختی موقت ، فی البداهه و استعجالی دارد و برای رسیدن

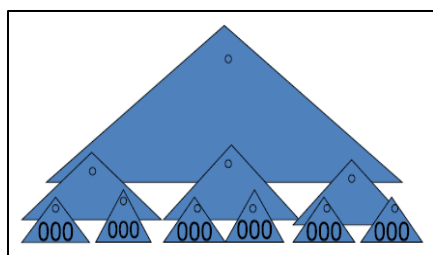
به هدف معینی بدون طرح و نقشه قبلی بوجود می آید . این ساخت شباهت زیادی با ساخت ماتریسی دارد که از نیروهای اطلاعاتی و

تخصصی بنا به مقتضای موارد مطرح شده مناسب ترین استفاده های تخصصی و عملیاتی را به عمل می آورد .

تعریف سازمان با گروههای متداخل یا سازمان حلقوی : این ساخت در قالب گروههایی که بوسیله اعضای مشترک با هم

در ارتباط می باشند شکل می گیرد . یک فرد در گروهی بعنوان مدیر ، در گروه دیگر به عنوان مرئوس و در گروه سوم به عنوان مشاوره

می تواند انجام وظیفه نماید.



تعریف ساخت سازمانی مدولار (ماجولار) : در این سازماندهی ، سازمان از واحدهای مختلفی تشکیل می شود که هر واحد با ضوابط اجرایی مشخص و مأموریتی که خود مینیاتوری از مأموریت کل سازمان است بطور مستقل ، غیر متمرکز و خود کفا به فعالیت ادامه می دهد.

تعریف ساختهای ترکیبی : هر یک از روشهای سازماندهی دارای معایب و محاسنی هستند که با توجه به شرایط زمانی و مکانی و مقتضیات موقعیتی آنهایی را بر می گزینند که بیشترین سود و کمترین زیان را داشته باشند . به همین دلیل در بسیاری از سازمانها ممکن است از ترکیبی از روشهای سازماندهی فوق بهره گرفته شود .

چگونگی تنظیم نمودار ستونی : استفاده از روش های : ۱- تحلیلی ۲- تلفیقی ۳- ترکیبی

تعریف روش تحلیلی در تهیه نمودار سازمانی : در این روش امر گروه بندی فعالیتها و وظایف از بالا به پایین انجام می گیرد . بدین معنی که هدف سازمان به هدفهای فرعی ، فعالیتها ، وظایف و عملیات تقسیم می شود و اجرای مجموعه ای از وظایف و عملیات تحت عنوان یک شغل به شاغل واگذار می گردد.

تعریف روش تلفیقی در تهیه نمودار سازمانی : در این روش کار گروه بندی از پایین به بالا انجام می گیرد . بدین معنا که ابتداء مشاغل مختلفی که باید در سازمان انجام شوند مشخص گردیده و آنگاه مشاغل مشابه و مرتبط در یک گروه قرار داده می شوند و به یک واحد سازمانی احاله می گردند و این کار ادامه می یابد تا سلسله مراتب سازمانی کامل گردد.

تعریف روش ترکیبی در تهیه نمودار سازمانی : این روش ترکیبی از روشهای تلفیقی و ترکیبی است و در عمل هم بیشتر از این روش استفاده می گردد و در عین حال به محتوای مشاغل نیز توجه شده و ارتباط لازم بین وظایف ، فعالیتها و هدفها مد نظر قرار می گیرد.

نکات مهم تهیه نمودار : ۱- تبادل نظر با مدیریت در زمینه تمرکز ، عدم تمرکز ، حیطه مدیریت و سطح یا بلند بودن سلسله مراتب ۲- تعادل اختیار و مسئولیت ۳- توزیع مناسب اختیارات بین واحدهای صف و ستاد ۴- توجه به ارتباط هر یک از مشاغل با هدف سازمان ۵- بررسی قلمرو هر یک از واحدها و در صورت لزوم ترکیب آنها.

تعریف نمودار توالی : این نمودار، از جمله نمودارهای تبادلی (interaction) است و برای نمایش ارتباط بین عامل ها و اشیاء مختلف در یک توالی زمانی و مبادله پیام بین آنها استفاده میشود. در. هر نمودار توالی باید حداقل یک عامل وجود داشته باشد به عملیات را راه اندازی کند.

کاربرد نمودار توالی : ۱. نمودار توالی چگونگی عملکردهای یک شی را با دیگر اشیا نشان می دهد و یک جنبه مهم به نام زمان را نیز در بر دارد. ۲. ترتیب رویدادها با گذشت زمان به انتها می رسد. ۳. نمودار توالی حاوی اشیایی است که با مستطیل نام گذاری شده نشان داده می شود و پیغام ها توسط فلش مشخص می شود. ۴. اشیا در بالای نمودار و از چپ به راست قرار می گیرند. ۵. گسترش زمان از طرف هر شی به سمت پایین توسط خطوط تیره (Life Line) مشخص می شود. ۶. مستطیل باریک روی خطوط تیره ، فعال سازی

(Focus of control) نامیده می شود و نمایانگر مدت فعال بودن و اجرای یک عملکرد توسط یک شی است ۷. از نمودار Sequence برای نمایش جریان عملیات برای Use Case استفاده میشود.

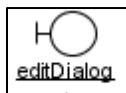
نحوه ترسیم دیاگرام توالی : برای ترسیم نمودار توالی باید از روی سناریو اقدام نمائیم یعنی از ابتدای سناریو شروع می کنیم و هر کجا به شی برخورد کردیم که با اشیا دیگر در رابطه است، آن شی را رسم و نحوه ارتباط آن را با دیگر اشیا نیز مشخص می کنیم. در پایان این دیاگرام ما باید مدلی تصویری از آنچه در Use case اتفاق می افتد را تولید کرده باشیم. هر Actor و اشیا مورد نیاز برای انجام عملیات Use Case در بالای نمودار نمایش داده میشود عملیات بصورت ارسال پیام بین اشیا انجام می گیرد. ترتیب ارسال پیام از بالا به پایین می باشد. هر پیام دارای یک نام بوده و می تواند پارامتر و اطلاعات کنترلی نیز داشته باشد. دو نوع اطلاعات کنترلی میتواند در کنار پیام نوشته شود، یکی شرط ارسال است که با بر آورده شدن آن پیام ارسال می شود و دیگری علامت تکرار است که مشخص می کند پیام چند مرتبه به اشیا فرستاده می شود. در نمودار پیکان نشانگر ارسال پیغام بین یک شی و شی دیگر یا شی و Actor میباشد.

Object ها دارای Stereotype های مختلفی هستند که به معرفی آنها می پردازیم :



Actor : همان عامل می باشد.

Boundary: به معنای User Interface هستند. یعنی هرکجا می خواستیم از واسط کاربر به کار ببریم، باید از این گزینه استفاده کنیم اجزای لازم برای برقراری ارتباط سیستم با یک بازیگر را در خود دارند. این کلاس ها 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.



Control: این Object ها همان اشیا کنترلی هستند یعنی هر جا در تحلیل قصد نمایش اشیا کنترلی را داشتیم از این گزینه استفاده می کنیم و معمولاً اشیا دیگر و رفتارهای تعبیه شده در یک use case را کنترل می کنند.

Entity: اشیا یی هستند که در سیستم وجود دارند و 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 مثلاً شی کتاب را در سیستم فروش کتاب با این شکل نمایش می دهند.



Table: اگر از میان اشیا از جدولی از پایگاه داده استفاده می کنید. می توانید برای نمایش آن از این گزینه استفاده کنید.

انواع پیغام : ساده (simple) - همگام - (synchronous) - ناهمگام (Asynchronous)

ویژگی پیغام : پیغام ساده کنترل را از شیئی به شی دیگر انتقال می دهد. اگر شیئی پیام همگامی را ارسال کند، منتظر پاسخ آن پیام می ماند، قبل از اینکه آن را بکار گیرد. در صورتیکه شیئی پیام ناهمگامی را بفرستد، دیگر منتظر آن نمی شود.

تعریف ایجاد شی : زمانی که یک توالی منجر به ایجاد شی می شود، در یک چهارگوش نامگذاری می کنیم و تفاوت آن با دیگر اشیا این است که در بالای نمودار ترتیبی قرار نمی گیرد و به جای آن در امتداد عمودی در زمان ایجاد قرار می گیرد. موقعیت قرارگیری آن طبق زمان است. پیامی که شی را به وجود می آورد، Create نامیده شده و پراترها به یک عملکرد دلالت می کنند.

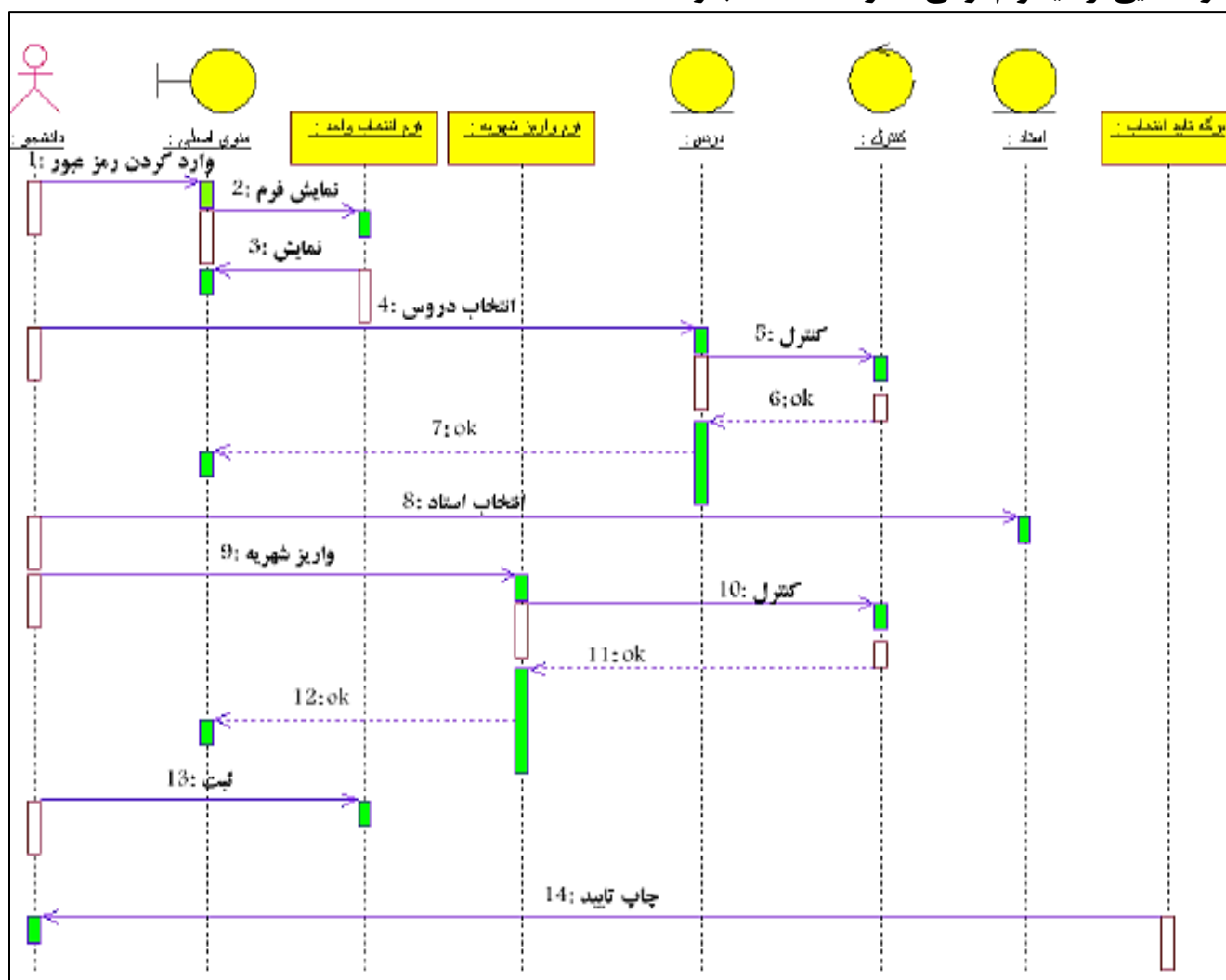
تعریف نمایش بازگشتی : زمانی که یک توالی منجر به ایجاد شی می شود، در یک چهارگوش نامگذاری می کنیم و تفاوت آن با دیگر اشیا این است که در بالای نمودار ترتیبی قرار نمی گیرد و به جای آن در امتداد عمودی در زمان ایجاد قرار می گیرد.

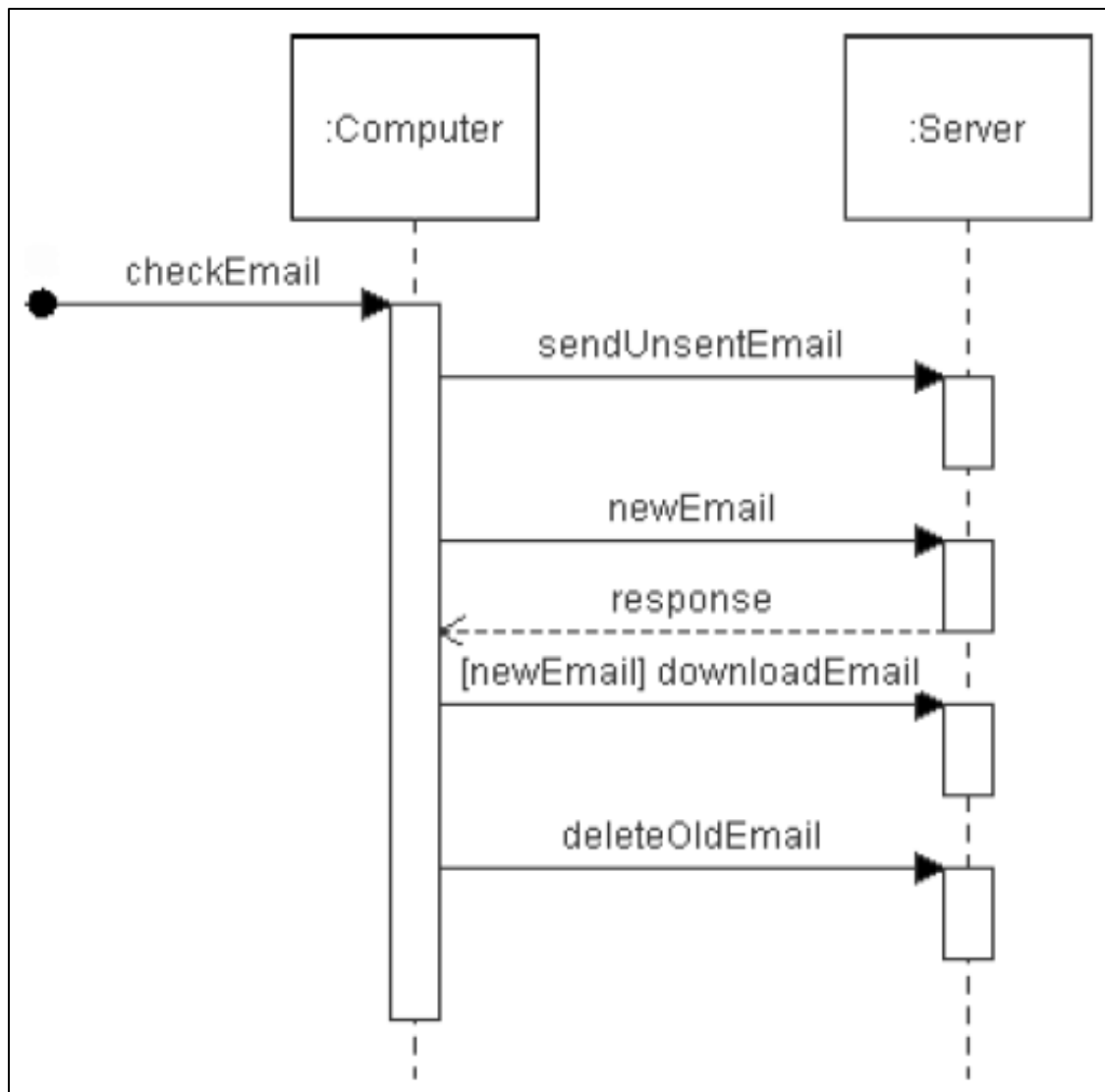
تعریف یادداشت سازی متنی : توضیحات دیاگرام توالی در سمت چپ صفحه قرار می گیرند.

تعریف محدودیت ها : برای نمایش زمانبندی پیام ها استفاده می شود و معمولاً از دو عبارت send time و receive time استفاده می شود.

برآوردی از زمان لازم جهت انجام یک فعالیت : در این حالت حدود زمانی رد و بدل شدن پیام ها مشخص می شود.

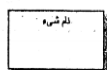
نمونه هایی از دیاگرام توالی : نمونه ۱ : انتخاب واحد





اجزاء نمودار توالی : ۱- عامل (Actor) ۲- خط زندگی (Lifeline) ۳- پیام (Message) ۴- پیام به خود (Self - Message)

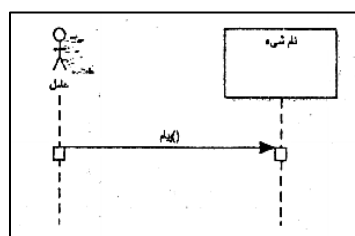
تعریف عامل (Actor): همان عاملی است که در نمودار use case تعریف شده است.



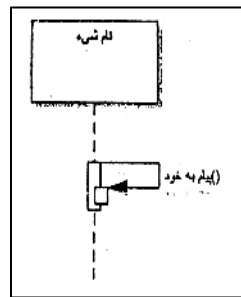
تعریف خط زندگی (Lifeline): نشان دهنده یک شیء در نمودار است و بصورت روبرو نمایش داده میشود:

تعریف پیام (Message): ارتباط بین یک عامل و یک شیء یا ارتباط بین دو شیء را بیان می کند. و با یک پیکان جهت دار نمایش

داده میشود.

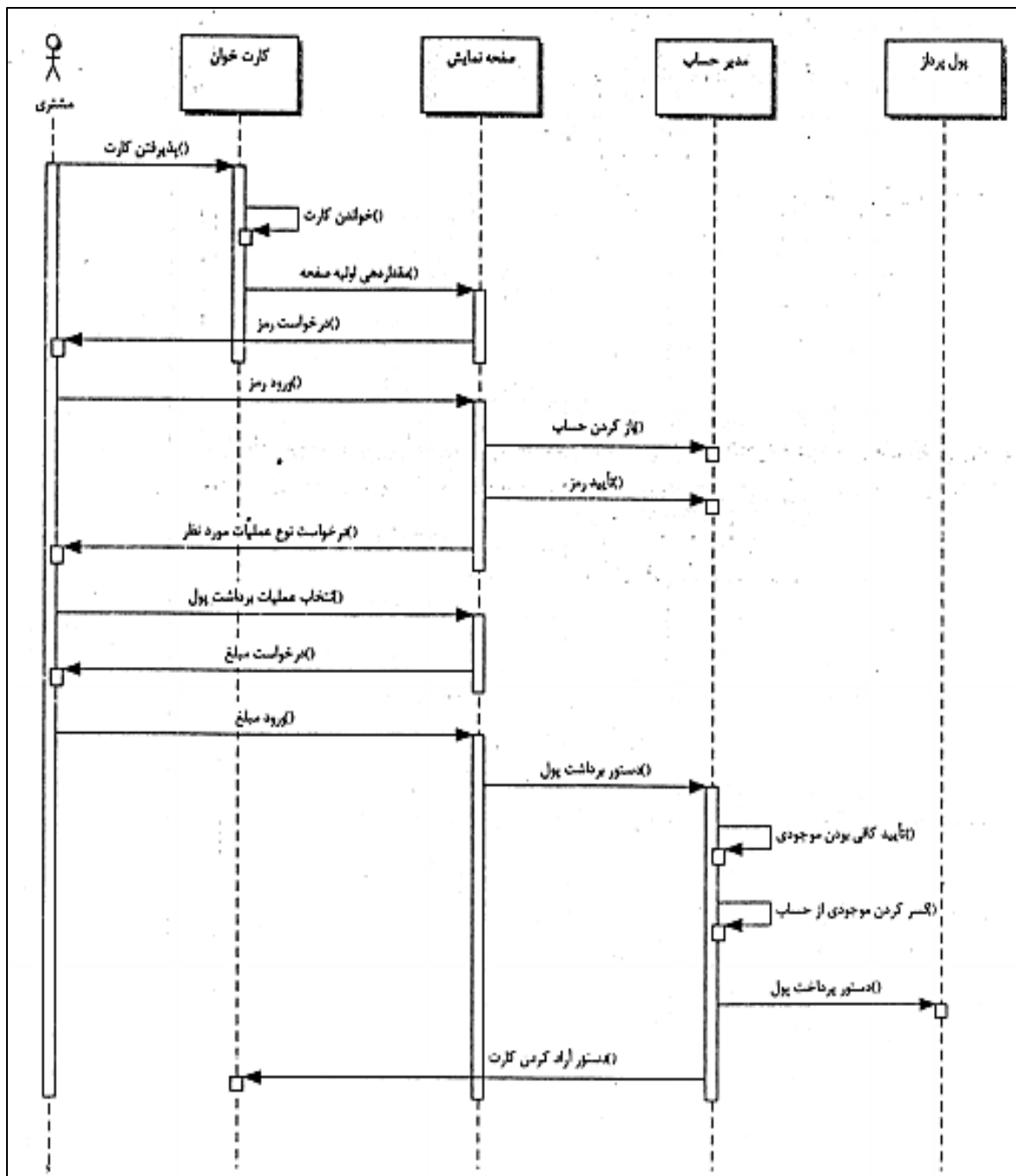


تعریف پیام به خود (Self - Message) : ارتباط بین یک شیء با خودش را بیان می کند و بصورت زیر نمایش داده می شود :

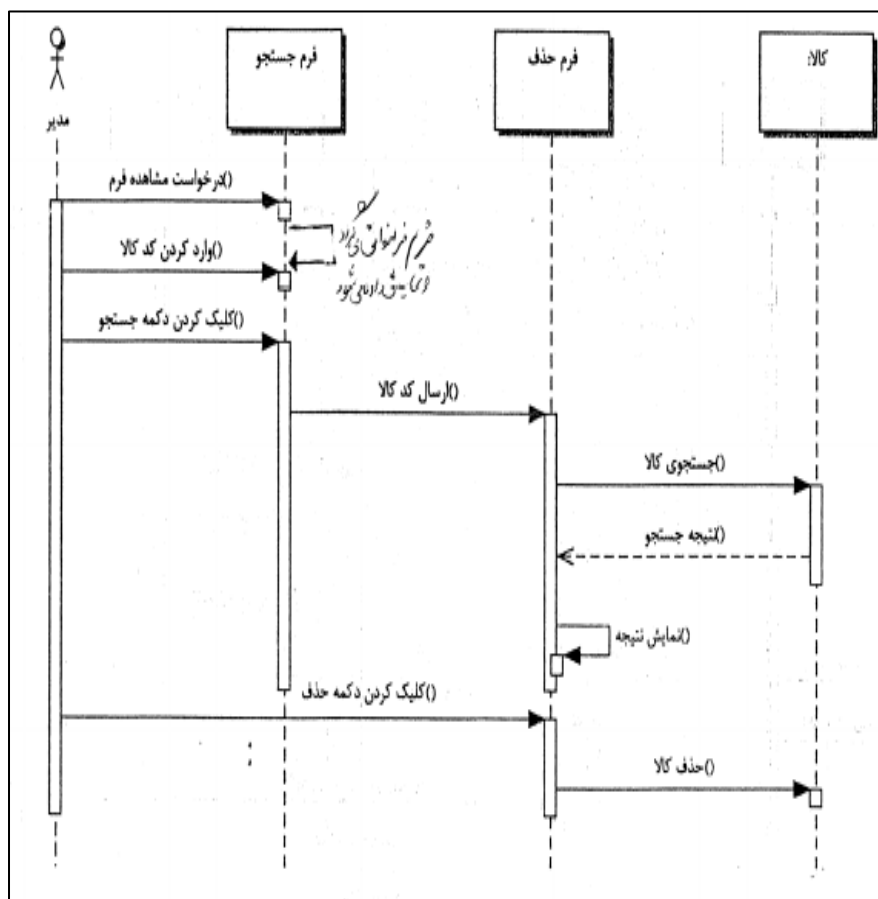


نمونه هایی از use case :

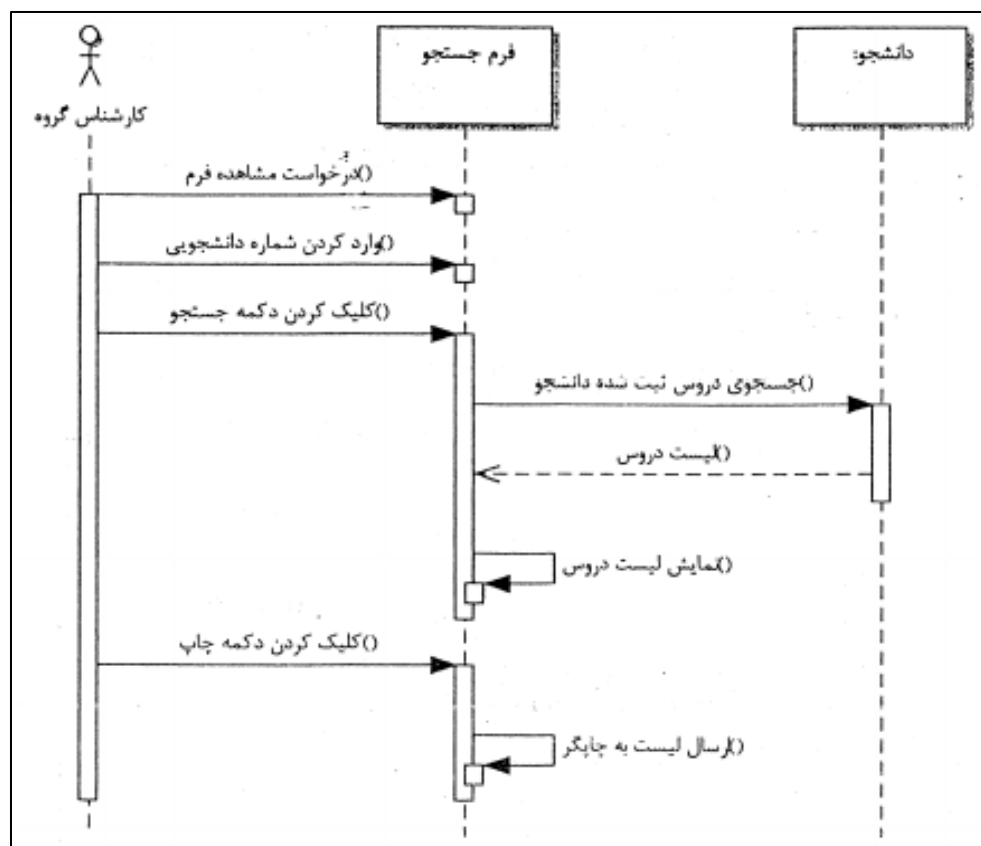
نمونه ۱ : برداشت پول از ATM



نمونه ۲: حذف یک کالا



نمونه ۳: چاپ دروس ثبت شده یک دانشجو (سیستم انتخاب واحد دانشگاه)



جدول تقسیم کار : جدولی است که نشان می دهد کارکنان یک سازمان در یک مدت معین ، چه کارهایی را انجام می دهند و چه مقدار وقت صرف انجام دادن این کارها می کنند .

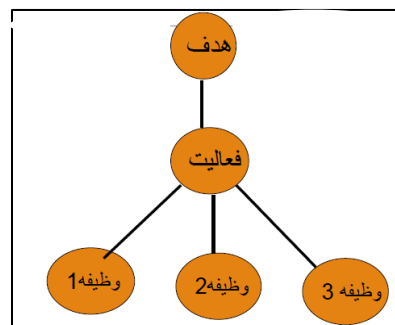
مزایای جدول تقسیم کار : ۱ - کسب آگاهی از بار واقعی کار و نحوه توزیع آن ۲ - شناخت محللهای تراکم کار و تراکم نیروی انسانی ۳ - آگاهی از تکرار ها و تداخلهای وظایف ۴ - آگاهی از نحوه اختصاص وقت به هر یک از وظایف ۵ - کسب اطلاع از اوقات اضافی کارکنان ۶ - آگاهی از تعداد واقعی نیروی مورد نیاز ۷ - ارزشیابی کارکنان

محدودیتهای جدول تقسیم کار : ۱ - چون از اطلاعات کارکنان استفاده می شود ممکن است با واقعیت تطبیق نداشته باشد .
۲ - تعیین زمان در مورد برخی وظایف بطور دقیق میسر نیست .

مقدمات لازم برای تهیه جدول تقسیم کار : ۱ - تشکیل جلسه توجیهی ، کارکنان و سرپرست مستقیم آنها پیرامون جدول تقسیم کار ۲ - کمک گرفتن از اطلاعات تکمیلی برای کسب اطلاعات دقیق تر ۳ - آنالیز بایستی در فرایند کار از سرپرست کمک بگیرد .
۴ - آنالیز بایستی رابطه ای دوستانه با کارکنان و سرپرست در حین تهیه و تنظیم جدول داشته باشد .

مراحل تهیه جدول تقسیم کار : ۱ - انتخاب واحد بررسی ۲ - تهیه لیست وظایف کارکنان ۳ - تهیه لیست فعالیتهای واحد
۴ - تهیه جدول تقسیم کار در وضع موجود ۵ - تجزیه و تحلیل جدول تقسیم کار در وضع موجود ۶ - تهیه جدول تقسیم کار در وضع پیشنهادی

تعریف مرحله انتخاب واحد بررسی : در این مرحله آنالیز کل سازمان را به واحدهای کوچک تقسیم و برای هر یک از واحد ها بطور جداگانه جدول تقسیم کار تهیه و سپس برای کل واحد ها از روی جداول گزارش جامعی تهیه و به مدیریت ارائه می دهد .



رابطه هدف ، فعالیت و وظیفه در جدول تقسیم کار :

تعریف مرحله تهیه لیست وظایف کارکنان : لیست وظایف کارکنان لیستی است که نشان می دهد هر یک از کارکنان یک واحد سازمانی در یک مدت معین چه وظایف و عملیاتی را انجام و چقدر وقت صرف انجام آنها می کنند .

نام و نام خانوادگی	عنوان شغل	نام واحد سازمانی
میزان تحصیلات رشته تحصیلی	تجارب در مجموع در شغل فعلی	نام سرپرست تاریخ
شماره وظیفه	شرح وظیفه	شماره فعالیت
		زمان صرف شده در روز
		جمع کل

نمونه ای از لیست وظایف انفرادی کارکنان :

تعریف مرحله تهیه لیست فعالیت‌های واحد: این لیست صورتی از کلیه فعالیت‌هایی که در یک واحد سازمانی انجام می‌شود می‌باشد که در برگیرنده مجموعه فعالیت‌های اساسی و عمده ای که کارکنان انجام می‌دهند است .

نمونه ای از لیست فعالیت های واحد :

نام واحد سازمانی		نام سرپرست واحد
نام آنالیز		تاریخ
شماره	شرح فعالیت	ساعات صرف شده در هفته
فعالیت		
		جمع کل

تعریف مرحله تهیه جدول تقسیم کار برای وضع موجود : در این مرحله آنالیز با استفاده از اطلاعاتی که در دو مرحله قبلی بدست آورده نسبت به تهیه جدول تقسیم کار اقدام می نماید. مرحله چهارم در واقع ترکیبی از دو مرحله دوم و سوم می باشد و جدول تقسیم کار نیز برابندی از لیست وظایف انفرادی کارکنان و لیست فعالیتهای واحد می باشد.

تعریف مرحله تجزیه و تحلیل جدول تقسیم کار : در این مرحله به بررسی انتقادی جدول تقسیم کار پرداخته می شود و سوالاتی از قبیل سوالات ذیل درباره اطلاعات جدول پرسیده و پاسخ آنها مشخص می گردد. ۱ - آیا کلیه فعالیتها در این واحد انجام می شود. ۲ - کدام یک از این فعالیتها بیشترین وقت را می گیرند و آیا لازم است این مقدار وقت صرف آنها شود. ۳ - چه مقدار وقت صرف کارهای غیر ضروری می شود. ۴ - آیا از تخصص و مهارت افراد استفاده مناسب به عمل می آید .

تعریف تهیه جدول تقسیم کار پیشنهادی : زمانی که پاسخ سوالات مطرح شده در مرحله قبل مشخص گردیدند معین می شود که در مورد برخی از وظایف تغییراتی ضروری می باشد مثلاً مشخص می شود که تخصیص زمان نسبت به اهمیت وظایف و فعالیتها به درستی صورت نگرفته است . لذا تغییرات مذکور در جدول جدیدی به نام جدول تقسیم کار پیشنهادی اعمال میگردد .

نمونه ای از جدول تقسیم کار :

نمونه‌ای از جدول تقسیم کار برای وضع موجود در فرم شماره (۳) ارائه شده است.

فرم شماره (۳)

جدول تقسیم کار

نام واحد:		نام سرپرست:		☐ جدول در وضع موجود ☐ جدول در وضع پیشنهادی	
شماره فعالیت	شرح فعالیت	نام شاغل:	نام شاغل:	نام شاغل:	نام شاغل:
		عنوان شغل:	عنوان شغل:	عنوان شغل:	عنوان شغل:
		شرح وظایف	شرح وظایف	شرح وظایف	شرح وظایف
		ساعات در هفته	ساعات در هفته	ساعات در هفته	ساعات در هفته

تعریف نمودار جریان کار: نمودار جریان کار عبارت است از تصویری از مراحل مختلفی که برای انجام یک کار از ابتداء تا انتها طی می شود.

علائم نمودار جریان کار: برای ترسیم نمودارهای جریان کار از علامتهایی استفاده می شود که هر کدام دارای مفهوم خاصی هستند این علائم در نمودار سبب تسهیل و تسریع درک نمودار می شوند که در اسلاید های بعدی ارائه شده اند .

تعریف عمل یا اقدام: برای نشان دادن عمل یا اقدام که معرف انجام کار است و زمانی که چیزی بوجود آمده یا تغییری حادث شده و یا اینکه جزئی بر چیزی اضافه شده باشد(مثل نوشتن یک نامه) از علامت دایره استفاده می شود.

تعریف بازرسی یا کنترل: از این علامتها برای نشان دادن بازرسی ، کنترل ، مقایسه و تطبیق آنچه که صورت گرفته است با معیارهای مورد نظر استفاده می شود. مثال: بررسی چکی که صادر شده است .

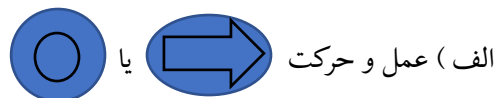
تعریف تصمیم گیری: تصمیم گیری گرچه نوعی عمل است و می توان آن را با دایره بزرگ نشان داد ولی ممکن است برخی از آنالیسرها از علامت لوزی برای نمایاندن آن استفاده کنند. مثال: اتخاذ تصمیم در مورد افزایش تولید

تعریف انبار یا بایگانی: این علامت نشان دهنده نگهداشتن کار بطور موقت یا دائمی است. مثال : ضبط یک نامه یا نگهداری یک کالا در قفسه.

تعریف تأخیر یا انتظار غیرموجه: این علامت نشان دهنده زمانی است که کار در انتظار می ماند تا مرحله بعدی آن صورت گیرد . تأخیر موقعی پیش می آید که به علت شرایط موجود ، اجرای مرحله بعدی امکان پذیر نباشد و قسمتی از کار متوقف شود .

حرکت یا ارسال (جابجایی): این علامت برای نشان دادن حرکت یا نقل و انتقال استفاده می شود که معرف تغییر مکان از جایی به جای دیگر است.

تعریف ترکیبی از علائم :



هرگاه عملی در حین حرکت انجام شود از این علامت استفاده می شود.



این علامت معرف انجام بازرسی در حین عمل است . مثل وزن کردن شیشه های نوشابه

انواع نمودار جریان کار: الف) نمودار عمودی یا یک ستونه ب) نمودار افقی یا چند ستونه

تعریف نمودار عمودی یا یک ستونه : این نمودار در مواردی بکار می آید که کلیه مراحل یک کار در یک قسمت یا یک واحد انجام شود. با کمک این نمودار آنالیز می تواند به ارتباط کاری موجود در یک واحد پی ببرد و از تکرار و تداخلهای وظیفه ای و عملیاتی

نمونه ای از نمودار جریان کار عمودی یا یک ستونه :

عنوان کار				خلاصه			
شروع کار	پایان کار	عنوان مراحل	وضع موجود	وضع پیشنهادی	تغییرات		
نام سرپرست	نام واحد	تعداد مراحل عمل					
		" بازرسی "					
		" تاخیر "					
		" بایگانی "					
		" نقل مکان "					
		زمان بر حسب دقیقه					
		مسافت بر حسب متر					
		وضع موجود ☐ وضع پیشنهادی ☐					
شرح مراحل							
		○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓ ○ □ ▽ ↓					

نمونه ای از نمودار جریان کار افقی یا چند ستونه :

عنوان کار				خلاصه			
شروع کار	پایان کار	عنوان مراحل	وضع موجود	وضع پیشنهادی	تغییرات		
		تعداد مراحل عمل					
		" اخذ تصمیم "					
		" رسیدگی "					
		" بایگانی "					
		" تاخیر "					
		" انتقال "					
		وضع موجود ☐ وضع پیشنهادی ☐					
شماره	عناوین واحدها	شرح مراحل انجام شده					

تعریف نمودار جریان کار افقی یا چند ستونه : نمودار جریان کار افقی یا چند ستونه این نمودار در مواردی بکار می آید که مراحل انجام یک کار در چند قسمت یا چند واحد انجام می شود. این نمودار به آنالیز کمک می کند تا به نحوه ارتباط بین واحدهای مختلف پی ببرد و از تکرارها و تداخلهایی که در آن واحد رخ می دهد آگاهی پیدا نماید.

مراحل بررسی نمودار جریان کار : مرحله اول : تعیین کار مورد نظر مرحله دوم : تعیین مراحل کار مرحله سوم : ترسیم نمودار جریان کار در وضع موجود مرحله چهارم : تجزیه و تحلیل نمودار جریان کار مرحله پنجم : تنظیم نمودار پیشنهادی

تعریف مرحله تعیین کار مورد نظر : در این مرحله آنالیز بایستی کاری که می خواهد مورد بررسی قرار دهد را تعیین و نقطه آغاز و پایان آن را دقیقاً مشخص نماید.

تعریف مرحله تعیین مراحل کار : در این مرحله آنالیز بایستی از کلیه مراحل انجام کار لیستی تهیه نموده و دقت نماید که صرفاً جریان کار مورد نظر را تعقیب و از آن منحرف نشود.

تعریف مرحله ترسیم نمودار جریان کار در وضع موجود : در این مرحله آنالیز با استفاده از اطلاعات کسب شده در مرحله دوم نمودار در وضع موجود را ترسیم می کند و اگر کلیه مراحل در یک واحد انجام شود نمودار عمودی و اگر در چند واحد انجام شود نمودار افقی رسم می گردد .

تعریف مرحله تجزیه و تحلیل نمودار پیشنهادی : در این مرحله آنالیز با دید انتقادی به نمودار وضع موجود نگاه کرده و سوالات ذیل را درباره اطلاعات جدول می پرسد . ۱ - این مرحله چیست یا عبارت دیگر در این مقطع کدام یک از مراحل کار باید انجام شود ؟ ۲ - چرا بایستی این مرحله انجام شود ؟ ۳ - چرا این مرحله توسط این فرد انجام می شود ؟ ۴ - چرا این مرحله در این زمان انجام می گیرد ؟ ۵ - این مرحله چگونه انجام می گیرد ؟ ۶ - چرا این مرحله در این مکان انجام می شود ؟

تعریف فرم : فرم یکی از ابزارهای برقراری ارتباط است که برای دریافت اطلاعات خاصی به صورت مکتوب تهیه و تنظیم می شود.

انواع فرم : ۱ - قلمرو استفاده ۲ - کار و وظیفه ای که انجام می دهند.

انواع فرم بر حسب قلمرو استفاده : الف) فرمهای داخلی - ب) فرمهای استاندارد

تعریف فرمهای داخلی : این فرمها در بخش کوچکی از سازمان استفاده می شوند .

تعریف فرمهای استاندارد : این فرمها در کلیه قسمتهای سازمان استفاده می شوند . (مثل برگه مرخصی)

انواع فرم بر حسب کار و وظیفه ای که انجام می دهند : ۱ - فرمهای پرسنلی ۲ - فرمهای مالی ۳ - فرمهای آموزشی ۴ -

فرمهای تدارکاتی ۵ - فرمهای تعمیرات و نگهداری وسایل ۶ - فرمهای متفرقه

هدف از بررسی و کنترل فرم : هدف از بررسی و کنترل فرم طراحی ، اصلاح و بهبود فرمهای مورد نیاز سازمان و حذف فرمهای اضافی و تهیه فرمهای مورد نیاز با حد اکثر بازدهی و حد اقل هزینه می باشد .

مراحل بررسی و کنترل فرم : ۱ - بررسی فرمهای موجود ۲ - تجزیه و تحلیل فرمهای موجود ۳ - تهیه طرح پیشنهادی برای فرمهای مورد نیاز

تعریف بررسی فرمهای موجود : در این مرحله آنالیز نمونه ای از فرمهای موجود سازمانی را جمع آوری و با دقت در شکل ظاهری آنها و موارد مندرج در آنها و همچنین ترتیب قرار گرفتن و نحوه تخصیص جا برای هر قسمت آن و نوع کاغذ و سایر مشخصات فرم را دقیقاً شناسایی می نماید .

تعریف تجزیه و تحلیل فرمهای موجود : در این مرحله آنالیز با تجزیه و تحلیل اطلاعات موجود به بررسی نقایص ، کاستی ها ، ایرادات و نارسایی های فرمهای موجود می پردازد . برخی از سوالاتی که در این مرحله بایستی پاسخشان مشخص گردند به شرح ذیل هستند .

۱ - چه فرمی مورد بررسی است؟ ۲ - آیا کلیه نسخ این فرم ضروری است؟ ۳ - آیا محتوای فرم برای کسانی که آن را بکار می برند واضح و قابل درک است؟ ۴ - آیا اندازه فرم مناسب است ؟ ۵ - آیا فرم اطلاعات مورد نیاز را تأمین می کند؟

تعریف تهیه طرح پیشنهادی برای فرمهای مورد نیاز : در این مرحله آنالیز با استفاده از اطلاعات مراحل قبل گام های زیر را بر می دارد:

۱ - تعیین هدف فرم ۲ - تعیین عنوان فرم ۳ - تعیین شماره یا کد فرم ۴ - نظر خواهی از تکمیل کنندگان و استفاده کنندگان از فرم
۵ - تعیین اطلاعاتی که در فرم درج می شود ۶ - تهیه طرح فرم ۷ - استفاده آزمایشی از فرم در یک واحد نمونه ۸ - جرح و تعدیل فرم با استفاده مؤثر از باز خور ۹ - تهیه فرم نهایی