طراحی سیستم اطلاعاتی خرید با استفاده از متدولوژی شیء گرا و مقایسه آن با متدولوژی ساخت یافته

دکتر محمدرضا تقوا
(عضو هیات علمی گروه مدیریت صنعتی دانشگاه علامه طباطبایی)
امیررضا آیت اللهی
(کارشناسی ارشد مدیریت صنعتی از دانشگاه علامه طباطبایی)

چکیده

در این مقاله پس از ذکر مقدمه‌ای کوتاه، ضمن بیان انواع سیستم‌های اطلاعاتی و طبقه‌بندی انواع مختلف متدولوژی‌ها، دیدگاه و متدولوژی شیء گرا به تفصیل مورد بحث قرار گرفته است. سپس از متدولوژی شیء گرا برای طراحی یک سیستم اطلاعاتی مکانیزه خرید برای دو فرآیند اصلی خرید داخلی و خارجی در صنعت خودرو، استفاده می‌شود. از این رو واحد خرید یکی از

شرکت‌های فعال در زمینه خودرو سازی (شرکت سازه گستر سایپا) انتخاب شده است و تحقیق مورد نظر به صورت مورد کاوی در آن صورت می‌پذیرد.

در پایان نیز با استفاده از پرسش‌نامه، متدولوژی شیء گرا با متدولوژی تحلیل و طراحی ساخت یافته از سه جنبه فنی، بکارگیری و مدیریتی و از دیدگاه صاحب نظران و متخصصان، مقایسه شده و یافته‌های آن مورد تحلیل قرار می‌گیرد.

کلمات کلیدی: متدولوژی، سیستم‌های اطلاعاتی، متدولوژی شیء گراء

۱ـ مقدمه

در دنیای امروز، سازمانهای متوسط به بالا برای ادامهٔ بقاء خود به سیستم‌های اطلاعاتی[1] محتاج هستند. شرکتهای ملی همچون گاز، مخابرات، آب و برق باید هر ماه صورت حساب ملی میلیون‌ها مشترک را حساب کنند تجسم اینکه تمام یا قسمتی از این کار به صورت دستی صورت بگیرد، آدم را شو که می‌کند. واضح است که این موضوع را می‌توان به سازمانهای مشابهی مثل بانک‌ها، ادارهای ثبت احوال و...نیز تعمیم داد.

1- Information Systems (IS)

همگام با این مسأله، به منظور تحلیل و طراحی سیستم‌های اطلاعاتی سال به سال متدولوژی‌های جدیدتری به بازار عرضه می‌شوند، متدولوژی‌هایی که ادعا می‌کنند از انواع قبلی چه در مراحل بعدی (طرحی نرم افزار و برنامه نویسی و چه در حوزه‌های متفاوت به نسبت کاربردی‌تر و قابل استفاده‌تر می‌باشند.

مشکلاتی که در دهه شصت میلادی سبب پدید آمدن بحران نرم‌افزار شدند، عبارت بودند از:

الف- تأخیر در تحویل به موقع نرم‌افزار نهایی

ب - کیفیت پایین نرم‌افزارهای ساخته شده

ج - هزینه بالای مراقبت و نگهداری

د - دوباره کاری‌ها و اتلاف وقت و انرژی

در همان زمان بود که مهندسی نرم‌افزار[1] با هدف حل مشکلات فوق، مطرح گردید. مهندسی نرم‌افزار پایه‌ریزی و استفاده از اصول و قواعد معتبر مهندسی به منظور مقرون به صرفه نمودن تولید نرم‌افزار قابل اطمینان و کارآ است. اما روش‌های مختلفی که به عنوان تکنیک‌های مهندسی نرم‌افزار بکار برده می‌شوند، تنها مسکن‌های موقتی بودند، زیرا با بالا رفتن سطح توقع و تقاضا از پاسخ‌گوئی به نیاز کاربران عاجز ماندند.

1- Software Engineering

مطرح‌ترین این تکنیک‌ها، روش تحلیل و طراحی ساخت یافته است که در اواخر دهه ۷۰ توسط یوردون و دی مارکو[1] پیشنهاد گردید. این تکنیک نیز اشکالات خاص خود را دارد که اهم آنها عبارتند از:

الف ـ عدم تأکید بر استفاده مجدد از برنامه و طرح‌ها: در این زمینه توجه به این نکته ضروری است که پیاده سازی با امکان مصرف مجدد از حجم دوباره کارها خواهد کاست.

ب ـ تغییر مدل زمینه در جریان گذر از تحلیل به طراحی: اجزاء و نمایه‌های تحلیل با باز نمایه‌ها و اجزاء طراحی تفاوت فاحشی دارند. همین مسأله موجب پیچیدگی روند گذر از نتایج تحلیل به طراحی می‌شود.

از اوسط دهه ۸۰ آنچه اصطلاحاً دوران چهارم استفاده از کامپیوتر نامیده می‌شود، آغاز شده است. آنچه این دوران را از دیگر دوران قبل از خود متمایز می‌سازد، اهمیتی است که نرم‌افزار در آن دارد. توجه به بهبود کیفیت و تقلیل هزینه ساخت‌نرم‌افزار از ویژگی‌های ممیزه این دوران است. از مشخصات جالب توجه این دوران ظهور متدولوژی‌های شیءگرا می‌باشد.

دیدگاه شیء گرا اولین بار در اواخر دهه ۱۹۶۰ به عنوان یک شیء برنامه نویسی (در زبان SIMULA) مطرح شد. پس از گذشت چند دهه، در اواسط دهه ۸۰ و

1- Yourdon & Demarc

اوائل دهۀ ۹۰، به جهت ناکار آمدی روش‌های سنتی در مباحثی همچون تحلیل و طراحی سیستم‌های اطلاعاتی و کامپیوتری و نیز ظهور سیستم‌هایی که مدل کردن آنها به روش‌های سنتی بسیار ناقص بود، تحلیل‌گران و طراحان سیستم به این فکر افتادند تا از دیدگاه شیءگرایی علاوه بر برنامه نویسی در زمینه تحلیل و طراحی سیستم نیز استفاده کنند.

مباحث مطرح شده در این علم تا اواخر دهه ۹۰ در حال تغییرات جدی و سیر تکاملی بود و از سال ۹۲ به بعد بود که این دیدگاه‌های مختلف به نقاط مشترک خوبی نزدیک شدند پیش‌بینی می گردد که این متدولوژی‌ها روز به روز به تدریج جایگزین متدولوژی‌های قدیمی شوند.

همانطور که از عنوان این مقاله "طراحی سیستم اطلاعاتی خرید با استفاده از متدولوژی شیءگرا و مقایسه آن با متدولوژی ساخت یافته" پیدا است، قصد نویسندگان ضمن بیان چکیده‌ای از یک پروژه تحقیقاتی، طراحی و مدلسازی یک سیستم اطلاعاتی مکانیزه برای فعالیت‌های خرید (شامل دو فرآیند اصلی خرید داخلی و خارجی) با استفاده از متدولوژی شیءگرا است. در گام بعدی نیز با استفاده از مجموعه‌ای از معیارهای ارزیابی، دو متدولوژی شیءگرا و ساخت یافته که دو شاخه اصلی متدولوژی‌های طراحی سیستم در زمان حاضر می‌باشند، از جهات گوناگون با هم مقایسه می‌شوند.

به همین منظور واحد خرید شرکت سازه گستر سایپا که قدیمی‌ترین شرکت طراحی مهندسی و تأمین قطعات خودرو در ایران بوده و یکی از شرکت‌های فعال و رو به رشد در این زمینه می‌باشد، انتخاب شده است. شرکت سازه گستر

سایپا (طراحی مهندسی و تأمین قطعه) با توجه به طبیعت وظایف، نمونه نسبتاً کاملی از یک سیستم خرید می‌باشد.

۲ ـ ادبیات تحقیق

ادبیات این پژوهش در بخش‌های زیر از نظرتان می‌گذرد:

۱ـ۲ـ سیستم

کتاب تحلیل ساخت یافته نوین به نقل از فرهنگ و بستر تعاریف سیستم را چنین ذکر می‌کند (یوردون ۱۳۷۶):

۱ـ اجزایی که با تعامل منظم یا همبستگی درونی در یک گروه، یک کل یگانه را شکل می‌دهند.

۲ـ مجموعه سازمان یافته‌ای که از دکترین‌ها، ایده‌ها یا قواعدی که اغلب برای بیان یک نظام یا کاری به عنوان یک کل سیستماتیک به کار گرفته می‌شوند. مانند سیستم مکانیک نیوتنی. این موضوع دو جنبه دارد:

الف) یک رویه سازمان یافته، مانند سیستم لمس جهت تایپ.

ب) الگویی برای طبقه‌بندی، نمادگذاری یا نمایش شماتیک؛ مثل سیستم ده دهی.

۳ـ ترتیب یا الگوی هارمونیک

۴ـ یک جامعه سازمان یافته یا وضعیت و حالت یک اجتماع

۲ـ ۲ـ سیستم‌های اطلاعاتی

سیستم‌های مکانیزه اطلاعاتی سیستم‌های ساخته شده به دست بشر بوده و با یک یا چند کامپیوتر کار کرده یا کنترل می‌شوند. بدون شک شما نمونه‌های مختلفی از اینگونه سیستم‌ها را در زندگی روزمره مشاهده کرده‌اید. این طور به نظر می‌رسد که کامپیوتر تقریباً تمام جنبه‌های مدرن جامعه امروز را تسخیر کرده است.

برای درک بیشتر موضوع سیستم‌های اطلاعاتی چند تعریف مقدماتی زیر ضروری می‌باشد:

الف ـ فناوری اطلاعات[1]: سخت‌افزار و نرم‌افزاری که بوسیله سیستم‌های اطلاعاتی استفاده می‌شود.

ب ـ سیستم اطلاعاتی: نوعی سیستم عملیاتی است که برای پشتیبانی یک یا چند سیستم عملیاتی دیگر از فناوری اطلاعات به منظور انتقال، ذخیره، باز یافت و نمایش اطلاعات استفاده می‌کند.

ج ـ سیستم عملیات[2]: سیستمی است که در آن انسان با استفاده از اطلاعات، فناوری و دیگر منابع در اجرای یک فرآیند کسب و کار، به منظور تولید محصول یا خدمت برای مشتری بیرونی یا درونی، مشارکت می‌کند.

1- Information Technology (IT)
2- Work System

د ـ عملیات[1]: بکارگیری منابع انسانی و فیزیکی مانند انسان، تجهیزات، زمان، تخصص و پول به منظور ایجاد ستاده مورد نیاز مشتری درونی یا بیرونی را گویند.

ه ـ موسسه تجاری[2]: مجموعه‌ای از چند سیستم عملیات به هم پیوسته که متفقاً برای تولید محصول یا خدمت برای استفاده مشتری بیرونی در محیط تجاری فعالیت می‌کنند.

و ـ محیط تجاری[3]: محدوده‌ای شامل خود موسسه تجاری و هر چیز دیگری که می‌تواند روی موفقیت آن اثر گذار باشد. مانند رقبا، حامیان، مشتریان و شرایط اقتصادی، اجتماعی و سیاسی(Alter; 1999).

در این تحقیق، در قدم اول یک سیستم مکانیزه اطلاعاتی خرید طراحی می‌گردد. طبیعی است که این سیستم از آنجا که به صورت «منفرد» طراحی می‌شود، بیشتر به سیستم پردازش تراکنش[4] نزدیک است، اما با طراحی برخی قابلیت‌های گزارش‌گیری، می‌توان آن را به سمت یک سیستم اطلاعات مدیریت[5] نزدیک کرد.

1- Work
2- Firm
3- Business Environment
4- Transaction Process System (TPS)
5- Management Information System (MIS)

۳ـ ۲ـ متدولوژی

فرهنگ آکسفود روش[1] را "شکل ویژه‌ای از رویهٔ بخشی از یک فعالیت فکری" تعریف کرده است. برای دقیق‌تر شدن در مفهوم متدولوژی، باید به مفهوم سایر کلمات با پسوند «اولوژی» بیشتر دقت کرد. به طور نمونه، زیست‌شناسی[2]، "علم توسعه زندگی فیزیکی" جامعه شناسی[3]، "علم توسعه و ماهیت قوانین اجتماعی "و روانشناسی، "علم ماهیت و کارکرد پدیده‌های روحی و مغزی انسان" تعریف شده‌اند.

با این تفسیرها مشخص است که «اولوژی» به معنی «علم» است، علم هم به معنی بخشی از دانشی است که روی موضوع خاصی تمرکز می‌کند.
بنابراین تعریف متدولوژی عبارت است از: علم روش‌ها یا بخشی از دانش که روی روشها تمرکز می‌کند.

متدولوژی مجموعه‌ای از رویه‌ها، تکنیک‌ها و ابزارهاست که مشتمل بر چند فاز است... با این حال متدولوژی بیشتر از یک مجموعه چیزهاست. متدولوژی معمولاً بر یک دیدگاه فلسفی استوار است ... و گرنه فقط یک روش بود... مانند یک دستورالعمل.

1- Method
2- Biology
3- Society

متدولوژی یک طرح جنگی یا یک دستورالعمل برای رسیدن به اهداف مورد نظر است (Tudor, 1997).

۱ـ ۳ـ ۲ـ انواع متدولوژی

متدولوژی‌ها از جهات متفاوتی قابل تقسیم شدن هستند. و آنها عبارتند از:

الف ـ طبقه بندی متدولوژی‌های مختلف از نظر نسل‌ها:

۱ـ نسل اول (متدولوژی‌های مبتنی بر چرخهٔ عمر)[1]: این متدولوژی مبتنی بر چرخهٔ عمر توسعه سیستم[2] بوده و استانداردهایی را برای تحلیل و برنامه‌نویسی پیشنهاد می‌کنند. اما بر روی ابزار مدل سازی خاصی تأکید نمی‌کنند.

۲ ـ نسل دوم (متدولوژی‌های ساخت یافته)[3]: این متدولوژی‌ها برای اولین بار تکنیک‌های متنوعی را برای مدل سازی ارائه می‌کنند. همچنین متدولوژی‌های این نسل مانند متدولوژی تحلیل و طراحی ساخت یافته[4] بر روی فرآیند توسعه بسیار تمرکز دارند.

۳ ـ نسل سوم (متدولوژی‌های نمایه ساز)[5]: در این نسل برای اولین بار رویکرد نمایه سازی[6] مطرح شد. با استفاده از نمایه و ارائه آن به کاربران، تجدید

1- Formal life – cycle Approaches
2- System Development
3- Structural Approaches
4- Structured System Analysis and Design Methodology (SSADM)
5- Prototyping and evolutionary approaches
6- Prototyping

نظرهای لازم در مقتضیات صورت می پذیرد. در این نسل بود که بحث مهندسی نرم‌افزار به کمک کامپیوتر[1] مطرح شد.

۴ ـ نسل چهارم (متدولوژی تکنیکی / اجتماعی مشارکتی)[2]: در این نسل، هدف اصلی از توسعه سیستم اطلاعاتی، بهبود کیفیت شغل کاربران است. این متدولوژی‌ها مانند ETHICS معتقدند باید کاربران را در فرآیند توسعه سیستم درگیر کرد.

۵ـ نسل پنجم (متدولوژی‌های حل مساله)[3]: این متدولوژی‌ها معتقدند باید به مسأله با یک دید چندگانه نگاه کرد. به عبارت دیگر این متدولوژی‌ها مانند Mutinies, SSM بر روی تصور کردن مسأله و داشتن احساس قوی نسبت به مشکل تأکید دارند (Hirschheim, 1995).

ب ـ تقسیم بندی دیگری نیز وجود دارد که به طور کلی متدولوژی‌ها را به دو دسته ساخت یافته و شیءگرا تقسیم می‌کند. از این نظر دارای دو دسته متدولوژی هستیم:

۱ ـ متدولوژی‌های ساخت یافته: عمده‌ترین متدولوژی‌های ساخته یافته عبارتند از:

1- Computer Aided Software Engineering Tools (CASE)
2- Socio – Technical participatory
3- Sensemaking and problem formulation

۱ـ ۱ـ SSADM [1981]

۱ـ ۱ـ STRADIS [1979] [1]

۱ـ ۱ـ IE [Martin, 1989] [2][Martin, 1989]

۱ـ ۱ـ YSM [3][Yourdon, 1993]

۱ـ ۱ـ MERISE [Quang, Chartier and Kastser, 1991]

۱ـ ۱ـ Euro Method [CCTA, 1994]

۲ ـ متدولوژی‌های شیءگرا: این متدلوژی‌ها نیز عبارتند از (ساعدی، ۱۳۷۹):

۲ـ۱ـ متدولوژی کود و یوردن (Coad, 1991)

۲ـ۲ـ متدولوژی BON (Nerson, 1992)

۲ـ۳ـ متدولوژی OSA [Embley, 1992]

۲ـ۴ـ متدولوژی Hodge / Mock (Hodge, 1992)

۲ـ۵ـ متدولوژی بوچ (Booch, 1994)

۲ـ۶ـ متدولوژی OMT (Rumbaugh, 1991)

۲ـ۷ـ متدولوژی طراحی مبتنی بر مسئولیت‌ها (Jacobson, 1992) (Writ-Brock,1990)

۲ـ ۸ـ متدولوژی Fusion (Coleman, 1994)

1- Structured Analysis, Design and Implementation of Information Systems
2- Information Engineering
3- Yourdon System Method

در این مقاله تقسیم بندی آخر مورد تأکید قرار می‌گیرد و سیستم مورد نظر با استفاده از یکی از متدولوژی‌های شیءگرا طراحی گردیده و در نهایت متدولوژی های شیءگرا و ساخت یافته از نظر پارامترهای مختلف مقایسه خواهند شد.

۱ـ۴ـ شیءگرایی و متدولوژی شیءگرا

برنامه نویسی شیءگرا اولین بار در سال ۱۹۶۰ بوسیله کسانی که با زبان SIMULA کار می‌کردند، شرح داده شد. در سال ۱۹۷۰ به عنوان بخش مهمی از زبان Smaltalk در شرکت زیراکس[1] توسعه یافت. پس از گذشت چند سال و در اوائل دهه ۹۰ میلادی به جهت ناکار آمدی روش‌های سنتی در مباحث تحلیل و طراحی سیستم‌های اطلاعاتی و کامپیوتری، تحلیل گران و طراحان به فکر افتادند تا از دیدگاه شیءگرا علاوه بر برنامه نویسی در زمینه تحلیل و طراحی سیستم نیز استفاده کنند.

۱ـ۱ـ۴ـ دیدگاه شیءگرا

متدولوژی شیءگرا بر دیدگاه شیءگرا استوار است. دیدگاه شی،گرا نگرشی جدید به دنیا و سیستم‌هاست. این دیدگاه سعی دارد تا با نگرش خود به عناصر یک سیستم، کل آن سیستم را مدل سازی کند. دیدگاه شیء گرا بر مباحثی نظیر

1- Xerox

شیء[1] کلاس[2]، مسئولیت[3]، عمل[4] و سناریو[5] استوار است و جهان را به صورت مجموعه‌ای از اشیاء مرتبط به هم می‌بیند.

برای ساخت یک مدل شیءگرا کاربرد به تکه‌ها یا شیءهای کوچکی تقسیم می‌شود، که به یکدیگر وابسته نیستند. بدین ترتیب کاربرد مورد نظر با کنار هم قرار دادن این شیءها بوجود می‌آید. یک کاربرد را می‌توان به یک ساختمان تشبیه کرد. پس از مشخص کردن مراحل ساخت و ایجاد مدل، اولین قدم برای ساخت، تهیه قطعات اساسی و بلوک‌های مختلف است که با کنار هم قرار دادن این بلوک‌ها می‌توان ساختمان مورد نظر را ساخت. شیءها در دنیای کامپیوتر به مثابه این بلوک‌ها هستند.

شیءهای اصلی فقط یکبار ساخته می‌شوند و می‌توان از آنها در ایجاد کاربردهای متفاوت استفاده کرد.

یکی از فوائد مهمی که در مدل شیءگرا وجود دارد، استفاده مجدد از جزءهای موجود در مدل است. بدین معنی که جزءها برای یکبار ساخته می‌شوند و به طور مکرر در مدل موجود یا حتی به طور مجدد در ایجاد کاربردهای دیگر

1- Object
2- Class
3- Responsibility
4- Operation
5- Senario

استفاده می‌گردند. درست شبیه این است که در ساختن قطعه یا ساختمان می‌توان از یک بلوک ساخته شده، چندین بار استفاده کرد. بنابراین می‌توان یک جزء شیءگرا را در یک سیستم حسابداری یا یک سیستم موجودی مورد استفاده مجدد قرار داد.

قبل از بوجود آمدن دیدگاه شیءگرا، برای ایجاد یک سیستم با استفاده از روش‌های قدیمی نیز لازم بود اطلاعاتی را در مورد سیستمی که می‌خواهیم پیاده سازی کنیم داشته باشیم. بنابراین اطلاعات مورد نیاز از توضیحات کاربر جمع‌آوری شده و پایگاه داده ای برای نگهداری این اطلاعات طراحی می‌شد.

همین طور که مشخص است در این روش، کمتر به رفتار سیستم توجه خواهد شد و بیشتر ذخیره اطلاعات مدنظر قرار می‌گیرد. این روش به جامعیت داده معروف است در سالهای اخیر هزاران سیستم با چنین روشی بوجود آمده‌اند. مدل سازی با روش جامعیت داده برای طراحی پایگاه داده‌ها و جمع‌آوری اطلاعات روش بسیار مفیدی است. اما استفاده از این روش برای ساخت سیستم‌های تجاری مشکلاتی را بوجود می‌آورد.

سیستم‌های تجاری به طور دایم در حال تغییر هستند. اگر چه به راحتی می‌توان در پایگاه داده سیستمی که بر اساس روش جامعیت داده بوجود آمده است تغییراتی را اعمال کرد، ولی اگر این تغییرات، عوض شدن قانون تجاری یا تغییر در رفتار سیستم باشد، اعمال تغییرات و پیاده سازی آنها کاری بسیار دشوار و گاه غیر ممکن می‌باشد و نیاز به ایجاد سیستم دیگری خواهد بود.

مدل شیءگرا برای حل چنین مشکلاتی بوجود آمده است. با استفاده از مدل شیءگرا می‌توان توجه خود را هم به اطلاعات[1] و هم به رفتار[2] سیستم معطوف کرد. طبق این روش می‌توان سیستم‌ها را به گونه‌ای ایجاد کرد که در مقابل تغییرات اطلاعاتی و رفتاری قابل انعطاف باشند(ساعدی، ۱۳۷۹).

۲ـ۱ـ۴ـ تشریح عناصر دیدگاه شیءگرا

در دیدگاه شیءگرا هر شیء رفتار مخصوص به خود و مسئولیت خاص خود را دارد. تعریف مختصر این عناصر به شرح زیر است:

الف ـ شیء: انسان یا مکان یا هر چیزی نظیر دانشجو، استاد، وسیله نقلیه، خانه، اطاق، میز، خودکار و یا یک بسته اطلاعاتی و یک شرح از پیاده سازی آن است.

ب ـ مسئولیت: چیزی که به شیء اختصاص داده می‌شود.

ج ـ کلاس: کلاس مجموعه یا گروهی از اشیاء است که خصوصیات یکسان دارند.

۳ـ۱ـ۴ـ زبان مدل سازی یکپارچه[3]

مدل سازی یک سیستم نرم افزاری قبل از ساخت یا نوسازی آن بسیار ضروری است. همانطوریکه برای یک ساختمان بزرگ، طرح اولیه می‌سازند، می‌توان برای سیستم‌های نرم‌افزاری نیز مدل تهیه کرد. اگر ما بخواهیم بخش

1- Information
2- Behavior
3- Unified Modeling Language (UML)

جدیدی را به خانه‌مان اضافه کنیم، بدون در نظر گرفتن مدل آن اقدام به خرید وسائل مورد نیاز مثل تخته و میخ نمی‌کنیم. بطور یقین ابتدا طرحی را که می‌خواهیم تهیه می‌کنیم و قبل از ساختن بخش جدید، طرحی از آنچه که می‌خواهیم، در دست داریم. درست است که زمان بیشتری را صرف می‌کنیم ولی مطمئناً نمی‌خواهیم که باران تمام آن چیزی که درست کرده‌ایم، یکباره خراب کند.

مدل‌ها نیز همین کار را در دنیای نرم افزار انجام می‌دهند. آنها طرح‌هایی برای سیستم هستند. یک طرح کمک می‌کند تا بخش جدیدی قبل از ساخت آن طراحی شود. در این صورت می‌توان مطمئن بود که طراحی‌ای که انجام شده بدون نقص است و بدین ترتیب قبل از ساخت سیستم، می‌توان تغییر نیازمندی‌های آن را نیز در طراحی سیستم اعمال کرد. یک زبان مدل سازی باید شامل موارد زیر باشد:

الف ـ عناصر مدل سازی[1]: مفاهیم معانی اساسی را مدل سازی می‌کند.

ب ـ علائم[2]: برای بیان عناصر مدل سازی نیازمند علائم مناسب می‌باشیم. این علائم تعبیر بصری یا تجسمی[3] از عناصر مدل‌سازی هستند. در روبرو شدن با سیستم‌های پیچیده، تجسم بصری در مدل سازی اهمیت ویژه ای می‌یابد.

1- Model Elements
2- Notation
3- Visual

ج ـ رهنمودها[1]: مدل ساز را برای ترسیم مدل هدایت می‌کند و اصطلاحات کاربردی را برای او بیان می‌دارد.

زبان مدل سازی یکپارچه تلاشی برای تعریف زبان استاندارد مدل سازی است که مورد پذیرش همگان باشد. UML یک زبان مدل‌سازی یکپارچه بصری برای ایجاد سیستم‌های شیءگرا و مبتنی بر اجزاء[2] است (Booch, 1999).

الف ـ تعریف

زبان مدل سازی یکپارچه زبانی برای مشخص سازی[3]، مجسم سازی[4]، ساخت[5]، مستندسازی[6] دست آوردهای سیستم نرم‌افزاری و مدل سازی کسب و کار و دیگر سیستم‌های غیر نرم‌افزاری است (Stevens, 2000)

زبان مدل سازی یکپارچه مجموعه‌ای از بهترین تجربیات مهندسی را که موفقیتشان در مدل‌سازی سیستم‌های بزرگ و پیچیده به اثبات رسیده است، عرضه می‌دارد.

ب ـ ابزارهای مدلسازی

زبان مدل سازی یکپارچه یک زبان مدل سازی را مشخص می‌کند که اتفاق نظر جماعت شیءگرا بر مفاهیم اساسی مدل سازی است.

1- Guidlines
2- Component Based
3- Specifying
4- Visualizing
5- Construction
6- Documenting

درUML برای ایجاد مدل‌ها و نمودارهای حوزه مسأله هیچ توصیه‌ای نمی‌شود و این تجربیات و یادگیری افراد است که تشخیص استفاده از کدام نمودارها و مدل‌ها را به ایشان یاد می‌دهد.

در یک دیدگاه UML شامل مدل‌های زیر است:

نمودار مورد کاربرد[۱]

نمودار کلاس[۲]

نمودارهای رفتار[۳]

نمودار حالت[٤]

نمودار فعالیت[٥]

نمودارهای تعامل[٦]

نمودار توالی[۷]

نمودار همکاری[۸]

1- Use Case Specification Diagram
2- Class Diagram
3- Behavior Diagram
4- State Chart Diagram
5- Activity Diagram
6- Interaction Diagrams
7- Sequence Diagram
8- Collaboration Diagram

نمودارهای پیاده سازی[1]

نمودار اجزاء[2]

نمودار استقرار[3]

لازم به ذکر است که نرم‌افزار Rational Rose برای استفاده از روش UML بوجود آمده است.

۴ـ۱ـ۴ـ زبان مدل سازی یکپارچه[4]

زبان مدل سازی یکپارچه یک زبان مدل سازی است و نه یک فرآیند و بر این اساس هیچ گونه علامت‌گذاری نیز برای فرآیند توسعه و ایجاد نرم‌افزار ارائه نمی‌دهد. سه مبدع UML فرآیندی را که در ابتدا به Objectory هم اکنون به Unified Process معروف است، اجرا کرده‌اند. این فرآیند در شرکت Rational از سال‌ها قبل در حال اجرا است.

بنابراین برای تکمیل این قسمت، کمی هم از Unified Process می‌گوییم. فرآیند توسعه فرآیندی تکراری و افزایشی است و در چهار مرحله[5] به انجام می‌رسد. هر مرحله می‌تواند از چند تکرار[6] تشکیل شود. در هر تکرار قدم‌های

1- Implementation Diagrams
2- Component Diagram
3- Development Diagram
4- Rational Unified Process (RUP)
5- Phase
6- Iteration

چرخه عمر وجود دارد. به عبارت دیگر قدم‌های تعیین نیازمندی‌ها، تحلیل، طراحی، پیاده سازی و تست در هر تکرار انجام می‌شود. مراحل به شرح زیر است:

الف ـ آغازش[1]

ب ـ جزئیات[2]

ج ـ ساخت[3]

د ـ انتقال[4]

۳ـ مدل سازی سیستم

۱ـ۳ـ ساختار فرآیندی

سیستم خرید شرکت سازه گستر دارای دو فرآیند یا دو زیر سیستم اصلی خرید داخلی و خرید خارجی می‌باشد که در ادامه تشریح می‌شوند:

الف ـ زیر سیستم خرید داخلی: خرید ملزومات (برحسب فرم درخواست تأمین ملزومات و تجهیزات اداری) یا قطعات (بر حسب فرم درخواست تأمین قطعات) از فروشندگان داخلی را بر عهده دارد. لازم به ذکر است درخواست

1- Inception
2- Collaboration
3- Construction
4- Transition

خرید ملزومات می‌تواند از سوی تمام واحدهای شرکت صادر شود، درخواست تأمین قطعات نیز باید توسط واحد تولید/ مهندسی صادر و پس از تأیید واحد لجستیک برای مدیریت خرید ارسال شود.

ب ـ زیر سیستم خرید خارجی: خرید مواد اولیه مورد نیاز سازندگان (برحسب فرم درخواست تأمین مواد) از فروشندگان خارجی را انجام می‌دهد. بسته به مورد چنانچه سازنده طرف قرارداد واحد تولید شرکت سازه گستر سایپا باشد یا طرف قرارداد واحد مهندسی، درخواست توسط این دو واحد صادر و پس از تأیید واحد لجستیک برای مدیریت خرید ارسال می‌شود.

پس از برگزاری جلسات متعدد، بررسی فرم‌ها، رویه‌ها و دستور العمل‌ها (که در بعضی مواقع با وضع موجود تفاوت اساسی دارند)، هر کدام از این دو فرآیند اصلی به زیر فرآیندهای ذیل تقسیم می‌شوند:

الف ـ فرایند خرید داخلی

۱ـ دریافت و بررسی درخواست خرید

۲ـ استعلام

۳ـ عقد قرارداد

۴ـ تأیید نمونه

۵ـ انجام خرید

۶ـ تحویل کالا به انبار

۷ـ تسویه حساب

۸ـ ارزیابی

ب ـ فرایند خرید خارجی

۱ـ دریافت و بررسی درخواست خرید

۲ـ استعلام

۳ـ اخذ مجوز تا دریافت پروفرما

۴ـ تأیید نمونه

۵ـ ثبت سفارش

۶ـ گشایش اعتبار

۷ـ بیمه و بازرسی کالا

۸ـ ترخیص

۹ـ دریافت کالا

۱۰ـ ارزیابی

همانطور که پیشتر نیز اشاره شد برای پاسخ به سوال اول تحقیق، با استفاده از سه ابزار مدل سازی UML (یعنی نمودارهای مورد کاربرد، کلاس و فعالیت) جنبه‌های فعالیتی، داده‌ای و رفتار دینامیکی سیستم را مدل سازی یا به عبارت دیگر طراحی می‌شود.

اما نمودار مهم دیگری که به عنوان «مواد اولیه» از آن استفاده می‌کنیم، نمودار گردش عملیات[1] می‌باشد که ترسیم آن ما را بیش از پیش به جزئیات و ماهیت سیستم آشنا می‌سازد. این نمودار هر چند جزء نمودارهای UML به حساب نمی‌آید اما کمک می‌کند تا ارتباط سیستم را با سایر سیستم‌ها و نهادهای بیرونی بدست آوریم.

۲ـ۳ـ ساختار داده‌ای

مولفهٔ دیگری که برای طراحی یک سیستم اطلاعاتی بسیار مهم است، داده‌ها و اطلاعات در گردش سیستم مورد مطالعه می‌باشد. در خلال مصاحبه و بررسی مدارک، فرم‌ها و گزارش‌های مورد استفاده در این سیستم شناسائی شدند. بخشی از این فرم‌ها مربوط به داخل سازمان و تحت پوشش رویه‌های تضمین کیفیت بوده و بخش دیگر مربوط به نهادها و موسسات خارج از سازمان (نظیر بانک، گمرک، بیمه و...) می‌باشند.

لازم به ذکر است به علت ورود و خروج اطلاعات مربوط به همین نهادهای خارجی، سیستم خرید (بخصوص در بخش خرید خارجی) جزء سیستم‌های پر داده سازمان به شمار می‌رود و این در حالی است که اطلاعات مورد استفاده سایر سیستم‌ها (انبار، فروش و...) منحصر به فرم‌های تضمین کیفیت می‌باشد.

1- Workflow Diagram

پس از شناسائی کلیه فرم‌ها، آن‌ها را در فرم «فهرست فرم‌ها» فهرست می‌کنیم: برای هر فرم نیز یک فرم «شناسنامه» که حاوی اطلاعات فرم از قبیل محل صدور، نحوهٔ گردش، توزیع نسخ، توضیحات و... می‌باشد، تهیه و به خود فرم الصاق می‌کنیم.

۳ـ۳ـ مورد کاربرد

مورد کاربرد، تعاملی نوعی ما بین کاربر و سیستم مورد نظرمان است. به عبارت دیگر مورد کاربرد در اصل همان نیاز کاربر است. به عنوان مثال در سیستم بانک، مشتری از سیستم انتظار دارد که چک خود را به پول تبدیل کند. این انتظار یا نیاز یک مورد کاربرد است (شکل ۱).

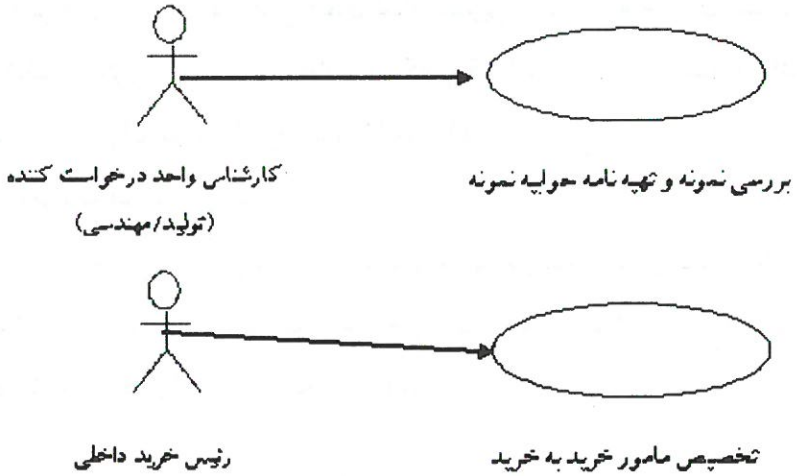
۱ـ۲ـ۳ـ تعریف مورد کاربرد

مورد کاربرد مشخص کننده رفتاری از سیستم یا زیر سیستم بوده و مجموعه‌ای از توالی عملیاتی است که سیستم جهت در اختیار قرار دادن یک نتیجه ملموس مورد نظر کاربر یا کنشگر آنها را انجام می‌دهد.

شکل ۱ـ مورد کاربرد و مثالی از آن

۲ـ۳ـ۳ـ کنشگر

نقشی که کاربر در ارتباط با سیستم ایفا می‌کند، کنشگر[1] نامیده می‌شود. در این نمودار توجه به نقش کاربر است و نه شخص کاربر (شکل ۲). به عنوان مثال یک فرد ممکن است هم مدیر عامل باشد هم عضو هیئت مدیره. این فرد دو نقش دارد و بر اساس هر یک از این نقش‌ها نیز تعامل خاصی با سیستم دارد. یک کنشگر را با علامت آدمک نمایش می‌دهند (شکل ۳).



شکل ۲- نمودار مورد کاربرد

1- Actor

انواع مختلفی از موجودیت‌های خارجی را می‌توان نام برد که به عنوان کنشگرهای سیستم می‌توانند عمل کنند:

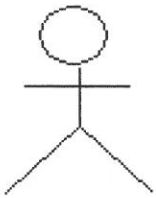
۱ـ افرادی که نیاز به انجام عملیات خاصی از سوی سیستم دارند (اکثر کاربران سیستم در این دسته قرار می‌گیرند).

۲ـ افرادی که سیستم به آنها نیاز دارد.

۳ـ سخت افزار یا نرم افزار.

٤ـ افرادی که وظیفه مدیریت یا مراقبت و نگهداری از سیستم را بر عهده دارند.

۵ـ سیستمی دیگر.



شکل ۳ـ کنشگر

مورد کاربرد فقط آنچه را که یک سیستم (یا زیر سیستم) انجام می‌دهد، توصیف می‌کند و از چگونگی انجام آن صحبتی نمی‌کند. در زبان مدل سازی توجه به این چگونگی مهم است. برای این منظور، توصیف جریان وقایع یک مورد کاربرد، کار مناسبی است (ساعدی، ۱۳۷۹).

۴ـ مقایسه متدولوژی شیءگرا و متدولوژی ساخت یافته

پس از اینکه با استفاده از متدولوژی شیءگرا یک سیستم اطلاعاتی مکانیزه خرید مدل سازی شد، این سئوال اساسی مطرح می‌گردد که آیا متدولوژی شیءگرا در مقایسه با متدولوژی ساخت یافته از جهات گوناگون برتری دارد یا خیر؟

بدین منظور کلیه معیارهای ارزیابی متدولوژی را مورد بررسی قرار داده و جمعاً ۲۸ معیار را در سه دسته کلی زیر طبقه بندی نمودیم:

۱ـ معیارهای فنی

۲ـ معیارهای به کارگیری

۳ـ معیارهای مدیریتی(رامسین، ۱۳۷۳).

در قدم بعد، پرسش نامه‌ای حاوی معیارهای فوق، تهیه و این پرسش‌نامه را در بین جامعه آماری منتشر نمودیم. لازم به ذکر است، مقایسه این دو متدولوژی در این تحقیق با اهداف زیر صورت می‌پذیرد:

الف) مقایسه عام دو متدولوژی فوق الذکر

ب ـ بررسی و ارزیابی نقاط قوت و ضعف هر یک از متدولوژی‌ها در مقایسه با دیگری (از نظر هر یک از سه دسته معیارهای ارزیابی)

ج) تشخیص توانایی‌ها و قابلیت‌های متدولوژی‌ها به صورت منفرد جهت راهنمائی الگوی انتخاب متدولوژی

د ـ ترکیب ویژگی‌های متدولوژی‌های متفاوت به منظور خلق یک متدولوژی جدید.

سطوح ارزیابی معیارها

به منظور ارزیابی هر یک از معیارها، سه سطح پیش‌بینی شده است که با استفاده از آن می‌توانیم نتایج ارزیابی را کمی نمائیم.

سطح A ـ متدولوژی شیءگرا از این نظر برتری محسوسی نسبت به متدولوژی ساخت یافته دارد.

سطح B ـ متدولوژی‌های شیءگرا و ساخت یافته تفاوت محسوسی از این نظر با هم ندارند.

سطح Cـ متدولوژی ساخت یافته از این نظر برتری محسوسی نسبت به متدولوژی شیءگرا دارد.

همانطور که قبلاً ذکر شد، در پاسخ به این سوال که آیا متدولوژی شیءگرا در مقایسه با متدولوژی ساخته یافته از جهات گوناگون برتری دارد یا خیر؟، از روش انتشار پرسشنامه و تجزیه و تحلیل نتایج آن استفاده می‌کنیم. این بخش نمایانگر یافته‌های انتشار این پرسشنامه و تجربه و تحلیل نتایج آن است.

بدین منظور با اغلب شرکت‌های نرم‌افزار واجد شرایط (شرکتهای نرم افزاری واقع در ۲۴ شرکت اول خدمات انفورماتیک) به طور کتبی و شفاهی در رابطه با این موضوع تماس حاصل شد که از این بین ۴ شرکت به درخواست ما پاسخ مثبت دادند و جمعاً ۴۷ پرسشنامه از این شرکت‌ها جمع‌آوری گردید.

۵ ـ تجزیه و تحلیل یافته‌ها

روش تجزیه و تحلیل یافته‌ها، روش آنالیز عوامل خواهد بود. بدین ترتیب که این دو متدولوژی را از نظر هر کدام از سه گروه معیار "فنی"، "بکارگیری "و "مدیریتی،" مورد بررسی قرار خواهند گرفت(جدول ۱).

۶ـ نتیجه‌گیری

این تحقیق در قدم اول به دنبال طراحی سیستمی با استفاده از متدولوژی شیء گرا با متدولوژی ساخت یافته از جهات گوناگون بود.

برای پاسخ به موضوع اول، نخست ساختار سازمانی، فرم‌های در گردش، فرآیندها و فعالیت‌های واحد خرید شرکت سازه گستر سایپا شناسائی شد، سپس در ۴ قالب، سیستم خرید مدل سازی گردید:

۱ـ نمودار کلاس

۲ـ نمودار مورد کاربرد

۳ـ نمودار فعالیت

۴ـ نمودار گردش عملیات

برای پاسخ به موضوع دوم نیز، پرسشنامه‌ای در بین کارشناسان شرکت‌های خدمات انفورماتیک انتشار یافت که از پاسخ‌های آنان می‌توان نتیجه گرفت که بر اساس هر یک از ۲۸ معیار، متدولوژی شیء گرا در مقایسه با متدولوژی ساخت یافته چه وضعیتی دارد.

گروه معیار	ردیف	معیار	A		B		C	
			درصد	فراوانی در کل	درصد	فراوانی در کل	درصد	فراوانی در کل
فنی	۱	ساختار سلسله مراتبی	٪۱۷	۸	٪۶۸	۳۲	٪۱۵	۷
	۲	سازگاری علائم مدلسازی	٪۲۳	۱۱	٪۵۹	۲۸	٪۱۸	۸
	۳	محدودیت تنوع	٪۱۵	۷	٪۶۸	۳۲	٪۱۸	۸
	۴	قابلیت ویرایش	٪۵۹	۲۸	٪۱۸	۸	٪۲۳	۱۱
	۵	قابلیت نگهداری	٪۹۱	۴۳	٪۹	۴	—	—
	۶	قابلیت تبدیل به کد	٪۹۱	۴۳	٪۹	۴	—	—
	۷	امکان ترکیب مراحل مختلف کاری	٪۳۲	۱۵	٪۶۸	۳۲	—	—
	۸	مبنای تولید نرم افزار	٪۲۵	۱۲	٪۶۶	۳۱	٪۹	۴
	۹	تحلیل کارائی	٪۸۵	۴۰	٪۱۵	۷	—	—
	۱۰	انعطاف پذیری	٪۵۱	۲۴	٪۲۵	۱۲	٪۲۴	۱۱
	۱۱	مدلسازی ساختار داده‌ای	٪۱۸	۸	٪۷۶	۳۶	٪۶	۳
	۱۲	مدلسازی وظیفه‌مندی سیستم	٪۶۸	۳۲	٪۲۳	۱۱	٪۹	۴
	۱۳	مدلسازی رفتار سیستم	٪۵۹	۲۸	٪۲۳	۱۱	٪۱۸	۸
	۱۴	محصولات کاری	٪۱۵	۷	٪۶۷	۳۲	٪۱۸	۸
	۱۵	مراحل کاری	—	—	٪۶۵	۳۱	٪۳۴	۱۶
	۱۶	تجرید	٪۸۲	۳۹	٪۱۸	۸	—	—
	۱۷	قابلیت استفاده مجدد	٪۷۵	۳۵	٪۲۵	۱۲	—	—
	۱۸	استقلال از زبان در مراحل مختلف	—	—	٪۷۴	۳۵	٪۲۵	۱۲
	۱۹	سازگاری با بانک اطلاعاتی	٪۵۱	۲۴	٪۴۲	۲۰	٪۷	۳
به کارگیری	۲۰	درک کاربر	—	—	٪۷	۳	٪۹۳	۴۴
	۲۱	حوزه کاربر	٪۶۸	۳۲	٪۳۲	۱۵	—	—
	۲۲	حجم سیستم	٪۲۶	۱۲	٪۶۸	۳۲	٪۶	۳
	۲۳	امکانات تولید	٪۲۵	۱۲	٪۵۷	۲۷	٪۱۸	۸
	۲۴	سابقه به کارگیری	—	—	٪۲۴	۱۱	٪۷۶	۳۶
	۲۵	ملاحظات آموزشی	٪۱۸	۸	٪۱۵	—	٪۶۷	۳۲
مدیریتی	۲۶	ادغام با سایر متدولوژی‌ها	٪۶۸	۳۲	٪۱۴	۷	٪۱۸	۸
	۲۷	مدیریت پروژه	٪۳۴	۱۶	٪۵۷	۲۷	٪۹	۴

جدول ۱ـ فراوانی هر سطح در هر معیار

با این تفاصیل می‌توان گفت از نظر معیارهایی که به ذات خود متدولوژی مربوط است (گروه معیارهای فنی و مدیریتی) نتایج این تحقیق، برتری قاطع

متدولوژی شیءگرا را نشان می‌دهد، اما از نظر مقایسه دو متدولوژی از معیار بـه کارگیری عکس این موضوع صادق است.

در ارزیابی کلی (تمامی عوامل) دو متدولوژی، برتری با متـدولوژی شـیءگرا است. البته لازم به ذکر است موارد فوق از نظر پرسش شوندگان، محدوده جامعه آماری و با توجه به محدودیت‌های موجود بوده‌اند.

منابع و مآخذ

الف ـ فارسی

۱ـ ساعدی، مهدی (۱۳۷۹): ***چکیده UML : زبان مدلسازی یکپارچه***، شرکت توسعه نرم افزار هدا.

۲ـ رامسین، رامان (۱۳۷۳): ***بررسی تفصیلی و ارزیابی متدولوژی موجودیت‌گرای ایجاد نرم افزار***، پایان نامه کارشناسی ارشد رشته کامپیوتر (نرم افزار)، دانشگاه صنعتی شریف.

۳ـ گودلند، ک. (۱۳۷۷): ***روش ساخت یافته تجزیه و تحلیل و طراحی سیستم‌ها***، ترجمه شهناز پیروزفر، نشر سالکان.

۴ـ مه آبادی، امین ا... (۱۳۷۸): ***آنالیز شیءگرا***، نشر آذرخش.

۵ـ یوردون، ادوارد (۱۳۷۶): ***تحلیل ساخت یافته نوین***، ترجمه مرجان رضایی، ماندانا کاویانی‌فر و جواد بهرامی، جلد اول، سازمان مدیریت صنعتی.

۶ـ یوردون، ادوارد (۱۳۷۸): ***تحلیل ساخت یافته نوین***، ترجمه مرجان رضایی، جلد دوم، سازمان مدیریت صنعتی.

ب ـ انگلیسی

1- Alter, Steven (1999). "***Information Systems: A Management Perspective***". Addison Wesley.

2- Booch, Grady & James Rumbaugh & Ivar Jacobson (1999). "***The Unified Modeling Language: User Guide***". Addison Wesley.

3- Clarke, Steve (2001). ***"Information Systems Strategic* Management"**. Routledge Publishing.

4- Conndly, Thomas & Carolyn Begg (1999). ***"Database Solutions: A Step by Step Guide to Building Databases"***. Addison Wesley.

5- Hirschheim, R. & H. K klein & K. Lyyliens (1995). ***"Information Systems Development & Data Modeling"***. Cambridge Press.

6- Hoffer, Jeffrey A. & Joey F. George & Joseph S. Valacich (2000). ***"Modern Systems Analysis and Design"***. Addison Wesley.

7- Kendall, K. (1998). ***"Systems Analysis And Design"***. Prentice Hall.

8- Maciaszek, Leszek A. (2000). ***"Requirement Analysis and Systems Design"***. Addison Wesley.

9- Perdita, Stvens & Rob Pooley (2000). ***"Using UML: Software Engineering with Objects and Components"***. Addison Wesley.

10- Satzinger, Johnw & Robert B. Jackson & Stephen D. Burd (2000). ***"Systems Analysis and Design: In a Changing world"***. Course Technology.

11- Simsion, Graeme C. (2001). ***"Data Modeling Essentials: Analysis, Design and Innovation"***. Coriolis.

12- Tudor, D.J. & I.J Tudor (1997). ***"Systems Analysis And Design: A Comparison of Structured Method"***. Macmillan.

13- Turban, E., E. McLean and J. Wetherbe (2002). ***"Information Technology for Management: Making Connection for Strategic Advantages"***. New York:John Wiley & Sons.

14- Wang, Shouhong (1999). ***"Analyzing Business Information Systems: An Object- Oriented Approach"***. Auerbach.