****

**ارزیابی معماری سیستم مرکز داده با استفاده از روش تحلیل توازن معماری (ATAM): مطالعه موردی**

**چکیده**

سیستم ترکیبی خوب و قوی نمی تواند از طرح اولیه سیستم منتخب و اجرا شده جدا شود. بنابراین، سیستم موجود یکپارچه سازی سیستم بزرگتری می شود که نیازهای سایر سیستم های متصل که مستلزم ارزیابی و تایدیه کامل می باشد را برآوده کند تا اینکه در زمان اجرای سیستم، تغیرات لازم مورد نیاز در آینده قادر خواهد بود تا یکپارچه اجرا شود. این مقاله ارزیابی نمونه سیستم مرکز داده با استفاده از روش ارزیابی معماری مبتنی بر سناریو به نام ATAM ( روش تحلیل توازن معماری)، برای تحلیل رفتار نشان داده شده توسط سیستم توسعه یافته نشان می دهد. نتایج تجزیه و تحلیل نشان می دهد که برخی بهبودها بویژه براساس امنیت، روایی و انطباق به منظور برآورده کردن الزامات مربوطه قوی و سهولت انجام حفظ و افزایش سیستم می باشد.

**کلمات کلیدی**: یکپارچه سازی، مرکز داده، انبار داده، معماری، ارزیابی، ATAM

1. **مقدمه**

دانشگاه پندیداکان گانشا (آندیکشا) سیستم اطلاعاتی متنوعی بویژه برای نیازهای اجرای فعالیتهای آکادمیک مانند مدیریت آموزش، فعالیتهای آکادمیک دانشجویی، مدیریت داده شامل مدرس، کارمند، داده تحقیقاتی و غیره ایجاد کرده است. بویژه در حال حاضر دوازده سیستم اطلاعاتی وجود دارد که در محیط آندیکشا اجرا شده است که هر سیستم بصورت جداگانه ساخته شده و همچنین سیستم داده مجزایی دارند. در ابتدا، سیستم ها با هدف برآورده کردن نیازها جهت تسهیل کردن فرایند ایجاد و بصورت سالیانه اجرا می شود. آنچنان که سیستم های ساخته شده متمرکز بر نیازها در حوزه مطالعه و اکثرا مجزا از وجود سیستم های دیگر می باشد که اطلاعات به هم پیوسته را در میان آنها داشته باشد. علاوه براین، برخی از سیستم های ایجاد شده بطورمجزا اجرا شد و آنها فقط در مدت شروع سیستم مفید بودند و سپس به علت مشکلات مختلف اطلاعات مشترک موجود نادیده گرفته شدند و سیستم اطلاعات نیاز به بررسی بیشتری هر دو براساس تحکیم طرح های خود و تجربه کاربر می باشد. در پایان سال 2016، تیم IT دانشگاه آندیکشا سیستم آینده نگر را به نام مرکز داده (DC) برای برآورده کردن نیازهای اعتباربخشی سازمانی پیشنهاد و طراحی می کند تا اینکه اطلاعات حاصل شده از سیستم موجود سازماندهی تر و هماهنگ تر شود. اگرچه سیستم های زیادی ایجاد شده است، همانطورکه مشاهده شد هیچ شیوه استانداردی در اجرای ارزیابی سیستم معماری اجرا شده از قبل در مرحله طراحی یا در عملیات سیستم وجود ندارد.

سیستم مرکزی پیشنهادی DC نیاز دارد تا بخوبی در یکپارچه سازی داده از کلیه سیستم های حامی توسعه داده شود تا اینکه موضوع های تکثیر و همچنین مشکلات به دلیل وجود ساختارهای مختلف داده می تواند کاهش یابد زیراکه اطلاعات شامل منبع اصلی فعالیت های آکادمیک کلی در آندیکشا می باشد. در مدت توسعه اولیه سیستم DC، تیم فناوری اطلاعات در گروه UPT-TIK Undiksha کشف کننده و طراحی کننده سرور انبار اطلاعاتی برای سیستم DC با استفاده از فناوری یکپارچه سازی برنامه کاربردی شرکت (EAI) بودند که بویژه ساختار مبتنی بر خدمات به نام یکپارچه سازی انواع برنامه‌های کاربردی سازمان (ESB) را اجرا می کنند.

در زمان افزایش این سیستم، مطالعه در ابتدا در رابطه با انتخاب کلی معماری سیستم اجرا شده انجام شده است. در زمان اجرای توسعه سیستم، منابع انسانی در دسترس همچنین دترمینان اصلی برای تکمیل و برآورده کردن الزامات سیستم می باشد. این مقاله بر نحوه انتخاب معماری انجام شده در چندین گزینه با اولویت بندی نیازهای کیفیت سیستم با استفاده از روشی به نام روش تحلیل توازن معماری (ATAM) تمرکز می کند.

ATAM یک روش ارزیابی معماری رویکرد مبتنی بر سناریو معروف می باشد. در انجام ارزیابی، شخصی باید مجموعه ای از سناریوها را بررسی کند که بر سیستم غیرعملکردی یا کیفیت نشان داده شده توسط سیستم تمرکز دارد.

ATAM تکنیک اثباتی برای ارزیابی معماری نرم افزار بطور سیستماتیک برای تناسب اهداف [2], [3] می باشد. سیستم های معماری مشابه زیادی با استفاده از روشی که توسط گامبو و همکاران در (4) انجام شده وجود دارد که سیستم اطلاعات بهداشت را بررسی و بیان می کند که بررسی عملکرد کیفیت در میان سایرین مهم و حیاتی است. با شیوه ATAM، نویسندگان تولید ارزیابی مبتنی بر سناریو را نشان دادند، موضوع عملکرد با ویژگی کیفیت برای تحلیل تجزیه شد. سایرموارد همانند 5، نویسنده ارزیابی الزامات امنیت برای سیستم تجارت الکترونیک مبتنی بر SOE را بررسی می کند که راه حل معماری موجود SOA را در یکپارچه سازی مکانیسم مخفی نهادینه می کند. مطالعه موردی دیگری از ATAM استفاده کرده است و بطورموفقیت آمیزی برنامه کاربردی آنلاین را ارزیابی و بهبود می بخشد همانطورکه در (6) توصیف شد.

زمانیکه نتایج ارزیابی این مطالعه موردی حاصل شد، سیستم جایگزین مشابه معماری و همچنین بهبود عملکرد اتخاذ شده را نشان می دهد.

بخش های مختلف این مقاله به شرح ذیل می باشد: بخش 1 مقدمه، بخش 2 خلاصه ای از روش ارزیابی ATAM را ارئه می دهد، بخش 3 خلاصه ای در مورد سیستم DC و شیوه های ارزیابی با نتایج حاصل شده را توصیف و بخش 4 نتیجه گیری می کند.

**2. روش**

ATAM هدفی برای حذف و اصلاح ویژگی کیفیت معماری دارد که تصمیم های طراحی معماری ویژگی کیفیت الزامات رضایت بخش (7) را بررسی می کند. ویژگی کیفیت شامل واکنش های مختلف سیستم مانند عملکرد، امنیت، در دسترس بودن، انعطاف پذیر بودن و غیره می باشد. همانطورکه در بخش (6) شرح داده شد، ATAM از رویکردهای معماری بجای سبک های معماری استفاده می کند. سبک معماری حداقل شامل مجموعه ای از انواع عناصر، رابط یا مکانیسم تعامل و هزینه غیر رسمی و توصیف مزایا می باشد. مراحل خلاصه شده شامل ATAM وصف شده در شکل 1 می باشد.



شکل 1: ATAM

ATAM شامل 9 مرحله می باشد که می تواند به 4 گروه به شرح ذیل دسته بندی شود:

الف) معرفی که شامل فعالیتهایی مانند (1) معرفی ATAM. در مدت این فعالیت، توصیف روش ارزیابی برای سیستم سهامداران لازم است که معمولا شامل نمایندگان مشتری، تیم معماری سیستم، نماینده های کاربر، مجریان، توسعه دهنده گان، تست کنندگان، یکپارچه سازان و غیره می باشد (2) ارائه محرک های بازرگانی. در این فعالیت، مدیران پروژه باید هدف تجارت را توضیح دهند که تلاش زیاد را تشویق می کند که محرک اصلی معماری مثلا دسترسی بالا، زمان آزادسازی، عملکرد سیستم، امنیت و غیره می باشد (3) معرفی معماری. در این فعالیت، سیستم معماری نیاز به شرح معماری پیشنهادی دارد که بر راه حل برای بررسی محرک های تجاری تمرکز می کند.

**ب) بررسی و تحلیل**

بررسی و تحلیل که شامل فعالیتهایی مانند (4) شناسایی رویکردهای معماری می باشد. در مدت این فعالیت، سیستم معماری رویکردهای معماری در دسترس را توسعه می دهد یا شناسایی می کند (5) تولید مطلوبیت کیفیت درخت سودمند. در این فعالیت، مطلوبیت کیفیت سیستم در دسته و سطح سناریوها شناسایی می شود (6) تحلیل رویکردهای معماری. در این فعالیت، هدف شیوه ها برآورده کردن الزامات غیر-مالی تحلیل بیشتر است و نکته های مانند ریسک، حساسیت و توازن شناسایی می شوند.

ج) تست که شامل فعالیتهایی مانند (7) سناریو اولویت بندی و فکر ناگهانی می باشد. در این فعالیت، مجموعه گسترده تری از سناریو ها از کل گروه سهامداران حذف، رای و انتخاب می شوند (8) تحلیل رویکردهای معماری. در این فعالیت، مرحله 6 تکرار می شود اما شامل مجموعه گسترده تری از سناریو های تولید شده در مرحله 7 برای اشاره به رویکردهای سنتی، ریسک ها، حساسیت و امتیازهای توازن می باشد که سپس مستند می شود. گزارش که شامل (9) ارائه نتایج می باشد. در این فعالیت، تیم یافته ها را در امتداد هر استراتژی مهاجرت پیشنهادی برای سهامداران گزارش می دهد.

**3. نتایج**

به منظور توصیف اجرای ATAM، نویسندگان مراحل بررسی انجام شده برای کل فرایند همانند (6) را به دو فاز تقسیم می کنند، اولین فاز، شامل کلیه مراحل از 1 تا 9 روش ATAM (همانند شکل 1) می باشد، فاز دوم مراحل 4 تا 9 را تکرار می کند تازمانیکه سیستم معماری بالغ تری تولید شود. اما در این مقاله، نویسندگان ترجیح می دهند تا از اصطلاح تکرار برای اجرای ATAM و اصطلاح فاز برای هر مراحل ATAM استفاده کنند همانطورکه توسط کازمان در 8 شرح داده شد.

**3.1 تکرار 1**

**الف) معرفی**

آندیکاشا موسسه آموزشی است که دانشکده های اجرایی متعددی و هر دانشکده چندین گروه مانند دانشکده فنی و مهندسی، گروه آموزش مهندسی انفورماتیک، گروه آموزش مهندسی برق و غیره دارد. قبلا، تقریبا کلیه گروه ها در سرتاسر آندیکشا برنامه های کاربردی مبتنی بر وب با پایگاه داده تنظیم شده خودشان توسط بخش فناوری اطلاعات داشتند. آنها به یکدیگر متصل نمی شوند، بنابراین به منظور ارائه اطلاعات تکمیل شده برای موسسه و همچنین برای فعالیت های اعتباری گروه ، سهامداران باید اطلاعات لازم را بصورت جداگانه از هر برنامه کاربردی استخراج کنند و برخی فعالیتها بصورت دستی انجام شده است، برای مثال، استفاده از نسخه خطی پایگاه داده سوالات جدید که در مکان مورد نیاز ایجاد می شود که می تواند منجر به اختلاف داده در مدت زمان تثبیت شود.

به منظور غلبه بر این مشکلات، توسعه سیستم DC برای ارائه خدمات متمرکز پیشنهاد شد که به کلیه خدمات موجود و همچنین برنامه های کاربردی جدید در سرتاسر آندیکشا برای برقرای ارتباط با یکدیگر و حفظ تبادل اطلاعاتی در همگام سازی را اجازه می دهد. اکثر برنامه های کاربردی مبتنی بر وب به زبان برنامه نویسی PHP کدگذاری می شوند و مثالهای پایگاه داده منبع باز یعنی MySQL استفاده می کنند. سیستم DC به عنوان ستون فقرات برای کلیه برنامه های کاربردی عمل خواهد کرد که نیاز به ارتباط با یکدیگر دارد و منبع اولیه خواهد بود که وظایفی مانند داده کاوی برای اهداف آکادمیک در آینده مورد نیاز است.

مدیریت همچنین قصد دارد تا مطمئن شود که یکپارچه سازی سیستم به سختی و به موقع انجام می شود همانطورکه آن توسط منافع سازمانی برای مشارکت در اعتبارگذاری موسسه در سطح ملی انگیزه دار می شود.

در مدت طرح اولیه، هدف یکپارچه سازی سیستم های موجود، بررسی منابع در دسترس مانند زیرساختارها، منابع انسانی و متخصصان فنی بطورساده می باشد و انتظار می رفت که اصلاحات بیشتری در آینده مورد نیاز خواهد بود. برای رسیدن به این هدف، تیم IT قصد دارد فناوری ساختار در دسترس را با معیارهای از پیش تعین شده مانند هزینه کم و منابع-باز به منظور کاهش زمان توسعه استفاده کند و براحتی حفظ می شود و همچنین برآورد الزامات سهامداران می باشد.

باتوجه ضروریت نیازهای مدیریت، سیستم DC تحت برنامه زمانبندی دقیق با منابع بسیار کمی توسعه یافت. بنابراین زمان کافی برای تحلیل نقاط قوت و ضعف سیستم معماری منتخب وجود نداشت.

برای سیستم DC، نمونه در تحقیقات قبلی در آندیکشا برای طراحی انبار داده (DWH) ایجاد شده است همانطورکه در شکل 2 توصیف شد. در تحقیق، جریان معماری داده NDS + DDS همانطورکه در [9] توصیف شد، اجرا و استفاده شد.



شکل2: معماری سیستم DC

به منظور یکپارچه سازی سیستم های موجود به DWH، سه گزینه سیستم کلی معماری شناسایی شده در مدت بحث تیم IT وجود داشت: (1) معماری ارتباط مستقیم، جاییکه DWH به هریک از منابع سیستم برای تبادل داده به ترتیب از طریق ارتباط مستقیم-سیم ارتباط داده می شود (2) معماری مبتنی بر خدمات وب، که هر سیستم منابع، داده خود را از طریق خدمات وب منتشر می کند که توسط DWH به هنگام نیاز استفاده می شود و (3) معماری مبتنی بر خدمات کارآفرینی، که هر سیستم منابعی داده های خود را از طریق ESB به اشتراک می گذارد که به DWH در سیستم DC ارتباط داده می شود.



شکل3: معماری ارتباط مستقیم (گزینه اول)



شکل4: معماری مبتنی بر خدمات وب (گزینه دوم)



شکل5: معماری مبتنی بر یکپارچه سازی انواع برنامه‌های کاربردی سازمان (ESB)

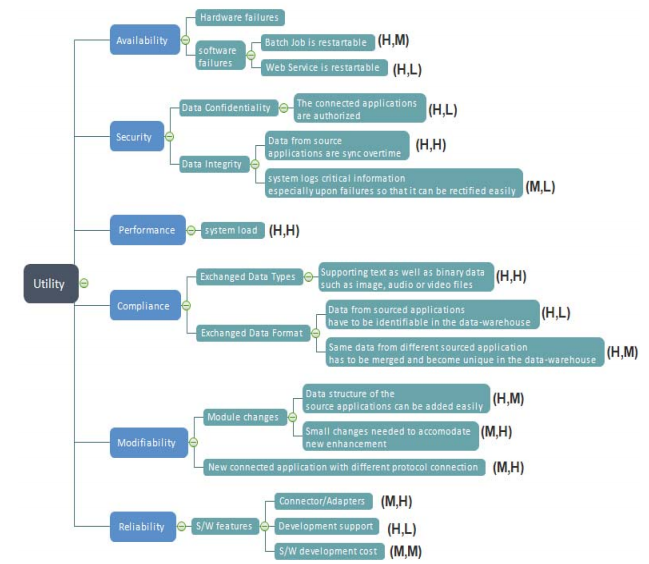
**ب) بررسی و تحلیل**

همانطورکه در (10) بیان شد، اولین وظیفه برای تحلیل معماری، دقیقا حذف اهداف کیفیت معماری خاص می باشد. برای این هدف، مکانسیم مورد استفاده از طریق ایجاد سناریو های الزامی سیستم مانند آنچه که در جدول 1 فهرست شد، می باشد.

جدول1: الزامات سناریو

|  |  |
| --- | --- |
| دسته | سناریو |
| استفاده از سناریو موردی | کلیه برنامه های کاربردی نیاز به ارتباط و اتصال با یکدیگر دارند و به گونه ای تایید شده اند که از هرگونه داده اضافی در میان آنها اجتناب می شود. |
|  | هر کدام از سایر برنامه های کاربردی متصل باید قادر باشند تا به انبار - داده مرکزی ارتباط داده شوند و داده را در صورت لزوم استخراج کنند. |
|  | سیستم DC باید قادر باشد تا هر داده اصلی کپی شده را از برنامه های کاربردی مختلف همزمانی یا همگام سازی کند |
|  | سیستم DC باید قادر باشد تا هویت نرم افزار کاربردی را تایید کند که قبل از دسترسی مجاز به آن ارتباط داده شود. |
|  | باید مکانیسمی جهت نظارت بر وظایف EOD باشد که بصورت موفقیت آمیزی انجام شده و شیوه ای که وظایف ناموفق EOD را مجدد اجرا می کند.  وظیفه پایانی EOD باید لاگ اطلاعات حسابرسی را در سیستم ایجاد کند. |
|  | سیستم منابع نباید بار زیاد را تجربه کند زمانیکه سیستم دیگری برای بدست آوردن اطلاعات متصل می شود |
| رشد سناریو | اضافه کردن ساختار داده های زیادی به برنامه های کاربردی متصل موجود و منعکس کردن تغیرات به انبار داده |
|  | سیستم DC باید هرگونه فرمت انتقال داده را شامل فرمت های دو دویی مانند تصاویر، فایل های صوتی و ویدئوئی را تسهیل کند |
|  | داده اضافه شده از کاربردهای منابع به انبار داده باید قابل شناسایی باشد. |
|  | اطلاعات مشابه داده اصلی از برنامه های کاربردی منابع باید بصورت یکپارچه و تک در انبارداده حفظ شود. |
| سناریوی اکتشافی | تغیر اندک یا بدون تغیر در هر یک از برنامه های کاربردی متصل به منظور حفظ راه حل مورد نیاز است. |
|  | هر افزایش رایجی مانند اعمال تغیرات از جداول برنامه های کاربردی منبع نیاز به اصلاحات اندک و سریع در سیستم DC دارد. |
|  | سایر برنامه های کاربردی قادر به اتصال از طریق شبکه های امن و غیرامن بدون مشکل می باشد. |
|  | سیستم باید قابلیت توسعه و تنظیم آسان را داشته باشد. |
|  | هر رابط مورد نیازی برای سیستم های متصل مانند اتصال به پروتکل متفاوت باید به سادگی فضا داده شود و در سیستم DC ادغام و یکپارچه شود. |
|  | هزینه ها بر اساس توسعه نرم افزار مورد نیازباید حداقل باشد و ترجیحا از نرم افزارهای منبع-باز استفاده کنند. |
|  | حمایت انجمن برای هر نرم افزار منبع باز مورد استفاده تا حد زیادی در دسترس و فعال است. |

سناریوهای اهداف ایجاد شده کیفیتی سپس به ویژگی خصوصیات درختان جمع می شوند که می توانند در اولویت بندی قرار گیرند ، همانطور که در شکل 6 ذکر شده است



شکل6: ویژگی کاربرد درختان

از سناریوهای فهرست شده و اولیت دارترین ویژگی کیفیت درختان کاربرد، فکر و بکر ناگهانی سناریو انجام شد و متوجه شدند که مهمترین ویژگی شامل امنیت، عملکرد، انطباق و اصلاح پذیری می باشد. اگرچه کلیه ویژگی های کاربردی معرفی شده نیاز به بررسی و تحلیل دارند، این مقاله فقط در مورد 4 ویژگی مذکور برای نشان دادن نحوه انتخاب معماری و نحوه حساسیت و همچنین امتیازهای توازن تحلیلی بحث خواهد کرد.

**تحلیل امنیت**

بر اساس امنیت، گزینه اول فرض می شود نامن ترین معماری باشد بطوریکه هر سیستمی باید به DC فرصت دهد تا به پایگاه داده خود با ارئه اطلاعات پایگاه داده به DC وصل شود. همانطورکه انتظار می رود آن در برابر حملات دشمن از بیرون آسیب پذیر است. گزینه های دوم و سوم از گزینه اول امن تر است.

**تحلیل عملکرد**

بر اساس عملکرد، گزینه اول فرض می شود که تست سریعی باشد زیراکه آن از اتصال مستقیم استفاده می کند. به هرحال، آن امنیت سیستم را تا حد زیادی تنزیل می دهد. گزینه دوم نیاز دارد تا از سیستم های منبع درخواست کند و اطلاعات به روز را برای سیستم های مقصد متصل همزمان منتشر کند. سیستم های منبع ممکن است درخواست های بیشتری از DC در شرایط خاص دریافت کند همانطورکه می تواند بخوبی عملکرد خود را تنزیل کند. گزینه سوم تاثیر کمتری بر عملکرد با هویت معماری خود دارد که از مکانیسم پیام رسانی و انتشار استفاده می کند. بنابراین سیستم های مقصد فقط نیاز دارد تا داده مورد علاقه را از DC بدون نیاز به ارسال درخواست های اضافی برای DC و برعکس ارسال کند.

**تحلیل انطباق**

براساس انطباق، گزینه های اول و دوم خدمات مجزایی نیازخواهد داشت زیرا اکثر برنامه های کاربردی موجود داده های دو دویی را در پایگاه داده خودشان ذخیره نمی کند. مثلا، فقط مکان تصویر، اطلاعات فایلهای صوتی یا ویدئویی در پایگاه داده موجود است درحالیکه فایلهای فیزیکی در هارد درایو سرور ذخیره می شود. گزینه سوم بهترین گزینه است زیراکه این دستگاه دارای FTP داخلی خود و همچنین قابلیت ارسال پیام های حباب برای انتقال داده های دو دویی از DC به سیستم های دیگر را دارد.

**تحلیل اصلاح پذیری**

بر اساس اصلاح پذیری، گزینه اول از سیستم های دیگر رنج می برد زمانیکه تغیرات در ساختار داده سیستم های منبع رخ می دهد. درحالیکه گزینه های دوم و سوم در برابر تغیرات منبع بسیار سازگارهستند. براساس تغیرات مورد نیاز ، زمانیکه DC نیاز به بهبود اطلاعات جدید از سیستم های منبع دارد، در گزینه اول، DC صرفا سوال و پرسش جدیدی را برای پایگاه داده سیستم های منبع می نویسد. به هرحال، هرگونه تغیرات بیشتر برای سیستم های منبع، همچنین به تغیرات کدها در سیستم DC نیازخواهد داشت که مطلوب نیست. گزینه دوم تغیرات کمتری در سیستم DC نیازخواهد داشت و فقط نیاز دارد تا خدمات جدیدی در سیستم منبع ایجاد کند. آن همچنین راه حل بهتری برای تیم IT در UPTTIK دارد زمانیکه اکثر توسعه دهندگان اکثرا آشنا با روش خدمات وب هستند. در گزینه سوم، ویژگی ESB آن مزایای زیادی بر اساس خدمات داخلی در دسترس می دهد، به هرحال، تغیرات ایجاد شده در سیستم DC ممکن است نیاز به تغیرات اضافی در منابع و همچنین سیستم های مقصد داشته باشد.

**ج) تست کردن**

از سناریوهای فهرست شده و اولویت دارترین ویژگی کیفیت درختان کاربرد، فکر و بکر ناگهانی سناریو برای تعین مناسبترین گزینه برای الزامات خاص مانند آنچه که در جدول 2 فهرست شد را می باشد.

جدول 2: تست ارزیابی معماری

|  |  |  |
| --- | --- | --- |
| ردیف | شرح الزامات | ریسک بله/عدم ریسک خیر |
| 1 | داده از برنامه کاربردی متصل در اکثر اوقات همزمان هستند. | خیر |
| 2 | سیستم DC باید از انتقال داده در متن و همچنین فرمت های دو دویی مانند تصویر، فایل های ویدئویی، صوتی بین انبارداده و برنامه های کاربرد متصل حمایت کند. | بله |
| 3 | برنامه های کاربردی متصل نیاز دارند تا انطباق را به منظور اتصال به سیستم DC نصب کنند. | بله |
| 4 | کار دسته ای برای انتقال داده ها از برنامه های کاربردی منبع به سیستم DC ، مجدداً قابل راه اندازی است. | خیر |
| 5 | داده ها از برنامه های کاربردی منبع یکپارچه می شوند و در انبار داده منحصر به فرد هستند. | خیر |
| 6 | ساختار داده های تغییرات برنامه های کاربردی منبع را می توان به راحتی در انبار داده جای و انعکاس داد. | خیر |
| 7 | هزینه استفاده از S / W حداقل است. | خیر |
| 8 | اجرای خدمات وب باید اضافه کاری در دسترس باشد و در زمان خاموش شدن به راحتی مجدداً راه اندازی شود. | خیر |
| 9 | اتصال برنامه های کاربردی به سیستم DC قبل از هر درخواست مجاز است. | خیر |
| 10 | داده از هر برنامه های کاربردی منبع باید قابل شناسایی باشند. | خیر |
| 11 | پشتیبانی جامعه از S / W تا حد زیادی در دسترس است. | خیر |
| 12 | برای پیشرفت باید تغییرات کوچکی انجام شود. | بله |
| 13 | سیستم DC باید اطلاعات مهم و حیاتی را مانند شناسه کار دسته ای ، شناسه برنامه منبع و موارد دیگر را بر وظایف ناقص وارد کند. | خیر |

**د) گزارش**

از جدول 2، برخی از پاسخ های ویژگی های مهم کیفیت وجود دارد که با افزایش ریسک بر تصمیم معماری بررسی می شود. باگزینه دوم، آن از مکان سناریو 2 و 12 رنج خواهد برد زیراکه آن تغیرات مورد نیاز بیشتری برای افزایش یا انتقال ضروری دودویی داده را الزام می کند. درحالیکه در گزینه سوم، آن از فضای سناریو 3 رنج خواهد برد زیراکه برخی از سیستم های موجود که از نسخه قبلی سخت افزار باقی مانده است، بنابراین تغیرات بیشتری برای رفتار کاربرد امکان پذیرنیست. به منظورحل کردن، ریسک های احتمالی، اکتشاف راه حل جدید برای استفاده برخی از نرم افزار رده عام قابل اصلاح (MOTS) می باشد یا ساختارهای جایگزین انجام شد.

**حساسیت و توازن**

باتوجه به اینکه گزینه های دوم و سوم در اکثر ویژگی های تحلیل شده بهتر هستند، آنها انتخاب و بررسی شدند. به هرحال، ویژگی های انطباق و قابل اصلاح معماری های مذکور برای فضای خدمات فوق العاده مورد نیاز حساس هستند. دراین نکته، آن امتیاز توازن معماری برای اضافه کردن خدمات جدید به سیستم DC مانند خدمات FTP یا توانایی اضافه کردن برای ارسال/اصلاح فایلهای دودویی از سیستم DC به سیستم های دیگر و بالعکس کشف شد. زیرا ویژگی انطباق بطورمثبت بررسی خواهد شد درحالیکه ویژگی های قابل اصلاح بطورمنفی در نظرگرفته می شود. با این اطلاعات، معماری اصلاح شده کشف شد.

**3.2 تکرار2**

باتوجه به ریسک ها که از تصمیم معماری حاصل می شود، حیاتی در نظرگرفته شد، باهم با سهامداران سیستم، در این مورد تیم IT، ، تکرار دوم ارزیابی ATAM انجام شد. تیم راه حل جدیدی را در بالای معماری موجود با جایگزین کردن محصول ESB با گزینه ساختار منبع- باز برای توسعه خدمات از حذف شناسایی کرده بود (آن شامل انتقال داده های بخش از طریق کار-دسته ای می باشد، که داده ها را از برنامه های کاربردی منبع برای سیستم DC خارج می کند و خدمات – REST توسط سایر برنامه های کاربردی متصل در دسترس می باشد).



شکل7: معماری اصلاح شده از طریق MOTS جدید

با این تغیرات ایجاد شده در سیستم معماری، گام های بررسی و تحلیل بیشتری برداشته شد. با ارزیابی لیست مشابه سناریو در جدول 2، معماری جدید به نظر می رسد تصمیم ریسک را در امتیاز های 2 و 3 حل کرده باشد. اما برخی کاربردهای اضافی موردی سناریو ها در جدول 3 آشکار شد که همه آنها در سطح ریسک و حیاتی برای سیستم بررسی شدند.

جدول3: سناریوهای اضافی در تکرار2

|  |  |
| --- | --- |
| دسته | سناریو |
| سناریو اکتشافی | سیستم DC باید به برنامه های کاربردی منبع به شیوهای امن دسترسی داشته باشد و خدمات آن باید بطورامن در دسترس باشد. |
|  | سیستم توسعه یافته باید توسعه دهنده-دوستانه برای افزایش اهداف باشد. |

**نتایج**

تصمیم معماری جدید به همراه ریسک در موارد ذیل بررسی می شود (1) یک ماژول امنیتی باید توسعه یابد و بخوبی تست شود قبل ازاینکه آن به عنوان ساختار منبع-باز اجرا شود که با ماژول های امنیتی داخلی سرو کار ندارد که می تواند خارج از جعبه تنظیم شود (2) ساختار منبع-باز به عنوان توسعه دهنده-دوستانه بررسی نمی شود بلکه آن منحنی-یادگیری گسترده ای جهت درک توسعه دهنده جدید دارد. این موضوع دیگری است همانطورکه تیم IT موجود اکثرا شامل فارغ التحصیلان جدید و توسعه دهندگان کم تجربه می باشند.

**4. نتیجه گیری**

در این تجزیه و تحلیل سیستم موردی DC، استفاده از روش ATAM روش ارزشمندی بوده است و به کشف نقاط ضعف و قوت گزینه های معماری قابل مقایسه کمک می کند. آن برای متقاعد کردن سهام داران جهت ارزیابی فناوری خاص سخت بوده است که در استفاده برای توسعه دادن سیستم معماری معروف بود زمانیکه برای آنها جدید بود. اکثر مدیریت ها در مورد ما، توسعه نرم افزاری را موضوع مهمی تجربه کردند تا اهداف خود را در سیستم ساختمان در بودجه و در شیوهای به موقع داشته باشند اما کیفیت سیستم را مصالحه می کند. آنچنان که، تعداد زیادی از سیستم ها داخلی آندیکشا عمل کردند اما مفید نبودند و نیاز دارند تا مجدد با اضافه کاری ایجاد و اختراع شوند. بعد از مراحل ATAM، ما تجربه بهتری از گزینه های سیستم معماری بدست آوردیم همانطورکه ویژگی آن می تواند به نسبت کیفیتی خاص تجزیه شود که شامل اهداف و تجزیه و تحلیل می باشد همانطورکه در ریسک/غیرریسک با سهامداران برای تصمیم معماری مواجه می شود. رویکرد همانطورکه در مراحل ATAM راهنمایی شد، عالی بود، آن می تواند برای حاصل شدن نتایج معماری مطلوب بطور مداوم پیگیری و تکرار شود.

با انجام تحلیل از طریق این روش ATAM، آن مشاهده می شود که معماری اولیه و فعلی سیستم DC عالی نیستند و به بررسی و تحلیل بیشتر در آینده نیاز دارد. آن همچنین از دوره حاصل شده اثبات شده است که اجرای روش می تواند انعطاف پذیر باشد، حتی قبل از انجام هرگونه پیشرفت سیستم ، می تواند از قبل استفاده شود.

همانطورکه در مدت فعالیتهای تحقیقاتی مشاهده شد، گزینه های MOTS زیادی در بازار برای یکپارچه سازی سیستم های مجزا به سمت ساختمان مرکز داده مرکزی وجود دارد. تحقیقات آینده تناسب و سازگاری را با هر مطالعه موردی ارائه شده را ارزیابی کنند که ارزش و اهمیت زیادی دارد.

تشکر و قدر دانی

از تیم IT آندیکشا UPT-TIK تشکر می کنیم که به ما در مدت فکر ناگهانی سناریو و همچنین تسهیل کردن جلسات مختلفی در مدت بحث های سیستم کمک کرد.

