مدل تصمیم‌یار پارامتری چند معیاره برای انتخاب سبک معماری سیستم‌های اطلاعاتی(مطالعهٔ موردی: نیروی انتظامی)

تاریخ دریافت: ۸۹/۹/۱۲ تاریخ پذیرش: ۸۹/۱۰/۳۰

غلامرضا شاه محمدی[۱]، سعید جلیلی[۲]

چکیده

نیروی انتظامی(ناجا) برای برقراری نظم و امنیت و انجام مؤثر مأموریت‌ها و خدمات انتظامی، از سیستم‌های اطلاعاتی گسترده، فراگیر و مقیاس بزرگ استفاده می‌کند. بهره برداری از این سیستم‌ها باعث ایجاد تحول عظیمی در انجام مأموریت‌ها و خدمات انتظامی شده است. این سیستم‌ها از یک‌سو باید از سرعت، قابلیت اعتماد و امنیت بالا برخوردار باشند و از سوی دیگر به لحاظ ضرورت تعامل و لزوم یکپارچه سازی آنها با یکدیگر و تعامل آنها با سیستم‌های کلان جامعه مانند سیستم‌های اطلاعاتی ثبت احوال و املاک نیز بایستی گسترش‌پذیر و منعطف باشند.

یادآوری می‌گردد ساخت سیستم های در مقیاس بزرگ، توسعه پذیر و منعطف نیازمند توجه ویژه به ساختار و سازماندهی اجزای آن سیستم است که معماری نرم‌افزار نامیده می‌شود. بنابراین طراحی معماری این سیستم‌ها که زیر بنای انعطاف و توسعه پذیری آینده آنها را پشتیبانی می‌کند، از اهمیت فوق‌العاده‌ای برخوردار است. تجربه متخصصان سیستم های اطلاعاتی بیانگر آن است که در طراحی این نوع سیستم‌ها بی توجهی به معماری آنها عامل اساسی عدم تأمین ویژگی‌های کیفی مورد اشاره بوده است. از آنجا که انتخاب سبک معماری از تصمیم‌های مهم مرحله طراحی محسوب می‌شود و در تأمین صفات کیفی سیستم نقش اساسی دارد، هدف این پژوهش ارائه مدلی برای انتخاب سبک معماری سیستم‌های اطلاعاتی ناجاست، به نحوی که سیستم‌های اطلاعاتی حاصل قابلیت پاسخگویی به نیازهای کیفی(سرعت، قابلیت اعتماد و ...) را داشته و در انجام مأموریت ها مؤثر واقع شوند.

در مدل پیشنهادی ابتدا ۱) مشخصه‌های[۳] مهم سیستم(مانند واکنشی بودن) تعیین می‌شود؛ ۲) با توجه به این مشخصه‌ها، سبک‌های معماری پالایش شده و سبک‌هایی که از این مشخصه‌ها حمایت می کنند، مشخص می‌شوند، ۳) صفات کیفی قابل ارزیابی در سطح سبک‌های معماری تعیین می گردد؛ ۴) سبک‌های معماری از دید این صفات کیفی به طور کمی ارزیابی می‌شوند و ۵) براساس نتایج سنجش سبک‌های معماری و تعیین اندازه تقریبی سیستم(تعداد مؤلفه‌های نرم‌افزار) و تعداد مؤلفه‌های مؤثر در تراکنش، با استفاده از رویکرد فرایند تحلیل سلسله مراتبی و تقدم صفات کیفی مورد نظرسیستم اطلاعاتی، سبک‌های معماری رتبه بندی شده و سبک معماری برتر برای سیستم اطلاعاتی مورد نظر توصیه می‌گردد. با استفاده از یک سیستم اطلاعاتی نمونه، کاربری و قابلیت استفاده از روش پیشنهادی تشریح می گردد.

نتایج این تحقیق نشان می‌دهد که تبدیل فرایند شهودی و مبتنی بر تجربه فعلی انتخاب سبک معماری به روش اصولی و سیستماتیک ارائه شده در این تحقیق است. استفاده از این روش باعث می‌شود سیستم‌های اطلاعاتی حاصل از کیفیت بالا برخوردار باشند و در حفظ و ارتقای امنیت اجتماعی نقش اساسی داشته باشند.

کلید واژه:

سیستم اطلاعاتی ناجا/ سبک معماری نرم‌افزار/ ارزیابی سبک‌های معماری/ انتخاب سبک معماری/ فرایند تحلیل سلسله مراتبی

۱- دکترای مهندسی کامپیوتر دانشگاه تربیت مدرس و عضو هیئت علمی دانشگاه علوم انتظامی

۲- استادیار دانشکده مهندسی برق و کامپیوتردانشگاه تربیت مدرس

3- Characteristics

مقدمه

ناجا سازمانی با گستره عظیم از وظایف در حوزه‌های مختلف مأموریتی و خدمات انتظامی است و مسئولیت خطیر برقراری نظم و امنیت را بر عهده دارد. سیستم‌های اطلاعاتی گسترده، بسیار بزرگ (از لحاظ حجم داده‌ها و پشتیبانی انواع فرایندها) و فراگیر ناجا تأثیر حیاتی در انجام موفق مأموریت‌های ناجا ایفا می‌کنند و گستره این تأثیر به سرعت در حوزه‌های مختلف مأموریتی ناجا در حال گسترش است و نوید بخش تحقق پلیس الکترونیک است. ویژگی‌هایی که خاص سیستم‌های اطلاعاتی ناجاست عبارتند از: ۱) نیاز به قابلیت اعتماد بالا(بیشتر از ۹۵٪) به جهت عملیاتی بودن مداوم سیستم‌ها در تعامل گسترده با مردم (مانند سیستم فوریت‌های پلیسی۱۱۰)، ۲) گسترش پذیری مسئولیت‌ها به دلیل نیازهای جدید و اصلاح وظیفه‌مندی‌های موجود، ۳) تأمین امنیت بالا به لحاظ جلوگیری از سوء استفاده از اطلاعات سیستم‌ها و یا استفاده غیرمجاز از آن‌ها، ۴) کارایی بالا به لحاظ پاسخگویی سریع به مردم (مانند سیستم شماره‌گذاری که باید استعلام از سیستم در کسری از ثانیه پاسخ داده شود)، ۵) داشتن مقیاس بزرگ (حجم زیاد تراکنش‌ها (بیش از ۳۰ هزار تراکنش) و تعداد زیاد رکوردها (بیش از ۳۰ میلیون رکورد)، ۶) نیاز به تعامل سیستم‌های اطلاعاتی ناجا با یکدیگر و با سایر سیستم‌های اطلاعاتی کلان کشور مانند سیستم اطلاعاتی ثبت احوال و ثبت اسناد، و ۵) ضرورت یکپارچگی این سیستم‌ها.

در طراحی سیستم تصمیم‌های متعددی اتخاذ می‌شود که بعضی از این تصمیم‌ها به کنترل پاسخ صفات کیفی سیستم کمک می‌کند. این تصمیم‌ها، تصمیم‌های معماری تلقی می‌شوند(بس، کلمنتس و کازمن[1]، ۲۰۰۳: ۳۲). معماری سیستم مجموعه موجودیت‌ها و ویژگی‌ها و روابط میان آنهاست که ساختار سیستم را تعریف می‌کند. معماری نرم‌افزار توصیف زیر سیستم‌ها و مؤلفه‌های یک سیستم نرم‌افزاری و روابط میان آنهاست (باشمن و همکاران[2]، ۱۹۹۶ :۳۹۴).

1- Bass, Clements and Kazman
2- Buschmann et al

با توجه به تأثیر شگرف معماری نرم‌افزار در پشتیبانی انواع صفات کیفی سیستم (بس، کلمنتس و کازمن، ۲۰۰۳: ۵۲)؛ طراحی این سیستم‌ها، یعنی سیستم‌هایی با ویژگی‌های اشاره شده در بالا، باید به گونه‌ای باشد که معماری سیستم از ویژگی‌های کیفی مورد نظر به نحو مطلوب حمایت کند. از طرف دیگر، تصمیمات معماری اتخاذ شده در فرایند طراحی، یک عامل مهم در توسعه موفق سیستم می‌باشند و به طور خاص، انتخاب یک سبک معماری مناسب، تأثیر قابل توجهی روی پشتیبانی انواع صفات کیفی سیستم دارد (سیعو و همکاران[1]، ۲۰۰۹: ۱) و اولین تصمیم بنیادی طراحی از نظر معمار است (بس، کلمنتس و کازمن، ۲۰۰۳: ۴۸).

صفت کیفی، علائقی است که به توسعه سیستم، نحوه فعالیت آن یا هر جنبه دیگری از سیستم که برای تعدادی از دست اندرکاران سیستم مهم است، مربوط می‌شود، مانند کارایی، قابلیت اعتماد (آی تریپل یی[2]-۱۴۷۱، ۲۰۰۰).

تاکنون روشی برای انتخاب سبک معماری با اتکاء زیربنای علمی و فنی ارائه نشده است. این مسئله ناشی از عدم بررسی تأثیر کمی سبک‌های معماری بر روی صفات کیفی مختلف (هریسون و اودو[3]، ۲۰۰۷: ۳) است و باعث گردیده که به کارگیری سبک‌های معماری در عمل روش‌مند (اودو و ایزدان[4]، ۲۰۰۵: ۱) نباشد و در حال حاضر انتخاب سبک معماری مبتنی بر شهود و تجربه توسعه دهنده نرم‌افزار می‌باشد (اسونبرگ و والن[5]، ۲۰۰۳: ۳). بنابراین، معماری حاصل از حیث پشتیبانی ویژگی‌های کیفی مورد نظر که تأثیر بالایی در پذیرش و بکارگیری سیستم اطلاعاتی دارد، مطلوب نیست.

با توجه به موارد فوق و ویژگی‌ها و صفات کیفی خاص سیستم‌های اطلاعاتی ناجا که در انجام مؤثر مأموریت های ناجا تأثیر برجسته‌ای دارند، به لحاظ عدم توجه کافی به معماری نرم‌افزار در فرایند طراحی آن‌ها، به مرور زمان به مرحله‌ای می رسند که

1- Seo et al
2- IEEE
3- Harrison and Avgeriou
4- Zdun
5- Svahnberg and Wohlin

قابل گسترش نیستند و یا از سرعت پاسخگویی و قابلیت اعتماد کافی و ... برخوردار نخواهند بود و بالطبع ناجا با مشکل جدی در توسعه و نگهداری این سیستم‌ها مواجه خواهد بود و در مواردی مجبور به طراحی مجدد آن‌ها می‌شود.

این تحقیق در پی روش‌مند کردن فرایند انتخاب سبک معماری نرم‌افزار است که طبق موارد اشاره شده نقش اساسی در تأمین ویژگی‌های کیفی مورد نظر سیستم اطلاعاتی دارد. بهره گیری از روش پیشنهادی در سازمانی به گسترگی نیروی انتظامی که وظیفه خطیر برقراری نظم و امنیت در جامعه را عهده دار است و کارکردها و مأموریت‌های آن به طور گسترده به سیستم‌های اطلاعاتی وابسته است، کمک مؤثری به ایجاد سیستم‌ها با ویژگی‌های کیفی مطلوب می‌کند.

ما(جلیلی و شاه محمدی ۱۳۸۵) روشی را برای انتخاب سبک معماری از میان چند سبک معماری کاندید برای سیستم اطلاعاتی ارائه کردیم. این مقاله کار قبلی را از طریق: ۱) در نظر گرفتن ۸ سبک رایج معماری (شاع و گارلان[1]، ۱۹۹۶: ۳۱-۲۰؛ باشمن و همکاران، ۱۹۹۶: ۳۹۴)؛ ۲) تعیین صفات کیفی قابل ارزیابی در سطح سبک‌های معماری؛ ۳) ارزیابی سبک‌های معماری از دید این صفات کیفی و ۴) ارائه روشی برای انتخاب سبک معماری سیستم تعمیم می دهد.

یادآوری می‌گردد سبک معماری با توجه به ارزیابی و رتبه بندی سبک‌های معماری کاندید برای یک سیستم اطلاعاتی و صفات کیفی مورد نظر آن سیستم توصیه می‌شود. در مدل پیشنهادی برای توصیه سبک معماری، با توجه به: ۱) مشخصه‌های مهم سیستم و پالایش سبک‌های معماری با توجه به این مشخصه‌ها؛ ۲) اولویت نسبی صفات کیفی سیستم مورد نظر؛ ۳) نتایج ارزیابی سبک‌های معماری از دید صفات کیفی؛ ۴) اندازه تقریبی سیستم(تعداد مؤلفه‌های نرم‌افزار) و تعداد مؤلفه‌های مؤثر در تراکنش، سبک‌های معماری کاندید رتبه‌بندی شده و بهترین گزینه توصیه می‌شود. بنابراین، در این مسئله با یک تصمیم‌گیری چند معیاره مواجه هستیم، که معیارها، صفات کیفی و گزینه‌ها سبک‌های معماری می‌باشد.

1- Shaw and Garlan

فرایند تحلیل سلسله مراتبی[1] از جامع‌ترین روش‌های تصمیم گیری چند معیاره است که نوع سازماندهی آن با تفکر انسان مطابقت دارد. این روش: ۱)معیارهای کمی و کیفی را در مسئله، در نظر می‌گیرد؛ ۲) امکان تحلیل حساسیت روی معیارها را دارد؛ ۳) میزان سازگاری و ناسازگاری تصمیم‌ها را نشان می‌دهد که از مزایای ممتاز آن است. این روش از یک مبنای تئوریک قوی برخوردار بوده و براساس اصول بدیهی[2] بنا شده است (ساعتی و وارگاس[3]، ۲۰۰۱). از روش AHP در حل مسائل متعددی در حوزه های مختلف استفاده شده است.

هدف اصلی این پژوهش، ارائه مدلی برای انتخاب سبک معماری سیستم‌های اطلاعاتی ناجاست، به نحوی که سیستم‌های اطلاعاتی، قابلیت پاسخگویی به نیازهای کیفی (سرعت، قابلیت اعتماد و...) را داشته و نقش مؤثری در انجام مأموریت‌های ناجا داشته باشد.

مبانی نظری و پیشینه تحقیق

۱- معماری نرم‌افزار

با گذشت زمان، نقش و تأثیر سیستم های نرم افزاری در عرصه های مختلف زندگی بشر، خصوصاً در امور حیاتی، بیشتر شده است و متناسب با آن بر اندازه و پیچیدگی سیستم ها افزوده شده است. با افزایش اندازه و پیچیدگی سیستم ها، معماری نرم‌افزار اهمیت اساسی پیدا می‌کند. با توجه به نقش بنیادی معماری نرم افزار در نیل به اهداف کیفی نرم‌افزار، طراحی معماری نرم افزار و ارزیابی آن از اهمیت ویژه‌ای برخوردار است. معماری مناسب، یک عامل حیاتی در موفقیت طراحی و ساخت سیستم های نرم افزاری محسوب می‌شود. معماری نرم‌افزار ساختاری از یک نرم افزار است که مؤلفه های نرم‌افزاری، مشخصه‌های قابل رویت این مؤلفه ها و روابط میان آن‌ها را بیان می‌کند (بس، کلمنتس و کازمن، ۲۰۰۳: ۴۵).

1- Analytic hierarchy Process
2- Axioms
3- Saaty and Vargas

۲- سبک معماری نرم‌افزار

سبک‌های معماری، مدل‌هایی برای حل مسایل طراحی معماری نرم‌افزار ارائه می‌دهند به نحوی که هر مدل، مؤلفه‌های تشکیل‌دهنده، مسئولیت مؤلفه‌ها، روابط و نحوه همکاری آنها را توصیف می‌کند (باشمن و همکاران، ۱۹۹۶: ۳۹۴). شاع و همکاران (۱۹۹۶: ۱۹۳-۲۵) هفت سبک معماری رایج را معرفی کرده‌اند. باشمن همکاران (۱۹۹۶: ۱۹۳-۲۵) نیز در کتاب خود، سبک‌های معماری را تشریح کرده‌اند. با توجه به اینکه در ارزیابی سبک‌های معماری از دید قابلیت اعتماد و کارایی، به مفهوم تراکنش و تعبیر آن در سطح سبک‌های معماری نیاز می باشد، ضمن اشاره به مفهوم تراکنش، هشت سبک معماری رایج بطور اجمالی معرفی می‌شود:

تراکنش: تراکنش، مجموعه‌ای از اعمال است که یک واحد منطقی(مانند یک سناریو از یک مورد کاربری) از کار را تشکیل می‌دهد (سیلبراسگات و همکاران[1]، ۱۹۹۹: ۱۳).

الف) سبک مخزن[2] (RPS). سبک مخزن شامل یک انباره مرکزی به نام مخزن و مجموعه‌ای از مؤلفه‌هاست که اعمال ذخیره، بازیابی و به روز رسانی اطلاعات را بر روی انباره مرکزی انجام می‌دهند (شاع و گارلان، ۱۹۹۶: ۲۱). در سبک مخزن تعامل هر مؤلفه با مخزن یک تراکنش محسوب می‌شود.

ب) سبک تخته سیاه[3]. مؤلفه‌های این سبک، تخته‌سیاه، منابع‌دانش[4] (متخصص‌ها) و مؤلفه کنترل است. مولفه کنترل، یک حلقه را اجراء و ضمن نظارت بر تغییرات تخته‌سیاه و ارزیابی متخصص‌ها، یکی از آنها را برای اجراء، فعال می‌کند(باشمن و همکاران، ۱۹۹۶: ۷۱). بنابراین، انجام هر تراکنش در این سبک مستلزم اجرای برنامه مؤلفه کنترل و بررسی وضعیت مؤلفه تخته‌سیاه توسط مؤلفه کنترل و انتخاب متخصص مناسب و تعامل متخصص با تخته‌سیاه می‌باشد.

1 Silberschatz et al
2- Repository (RPS)
3- Blackboard (BKB)
4- Knowledge Resources

ج) سبک لوله و فیلتر[1] (P/F). این سبک از تعدادی مؤلفه تشکیل شده است. هر مؤلفه در نقش یک صافی (فیلتر) است و تعدادی ورودی و تعدادی خروجی دارد. خروجی هر مؤلفه، ورودی مؤلفه بعدی است (شاع و گارلان، ۱۹۹۶: ۲۱). در انجام یک تراکنش کلیهٔ فیلترها مؤثر می‌باشند.

د) سبک لایه‌ای[2] (LYD). در سبک لایه‌ای، تمرکز روی سطوح مختلف انتزاع در نرم‌افزار است. سبک لایه‌ای به صورت سلسله مراتبی سازماندهی می‌شود. هر لایه خدمتی را برای لایه بالاتر فراهم و از خدمت لایه پایین‌تر استفاده می‌کند (شاع و گارلان، ۱۹۹۶: ۲۵). در انجام یک تراکنش کلیهٔ لایه‌ها مؤثر می‌باشند.

و) سبک فراخوانی ضمنی[3] (I/I). این سبک شامل تعدادی مؤلفه (مولد رویداد، علاقه‌مند به رویداد یا هم مولد و هم علاقه‌مند به رویداد) است و یک مؤلفه در نقش هماهنگ‌کننده یا توزیع کننده[4]، با وقوع رویداد، از طریق جدول نگاشت رویداد به مؤلفه علاقه‌مند مراجعه، ضمن تعیین مؤلفه علاقه‌مند، پیامی به او ارسال یا رویه‌ای از آن مؤلفه را فراخوانی می‌کند (شاع و گارلان، ۱۹۹۶: ۲۳). تراکنش در این سبک با وقوع رویداد شروع شده و مؤلفه توزیع کننده، مؤلفه علاقه‌مند را فعال می‌کند. با اتمام فعالیت مؤلفه علاقه‌مند تراکنش خاتمه می‌یابد.

ر) سبک متقاضی/سرویس‌گر[5] (C/S). مؤلفه‌های این سبک متقاضیان، سرویس‌گر(ها) و مخزن می‌باشند. متقاضیان برای استفاده از خدمات سرویس‌گرها، باید از نام و خدمات ارائه شده توسط آن‌ها اطلاع داشته باشند (شاع و گارلان، ۱۹۹۶: ۳۱). تراکنش در این سبک با ارسال درخواست متقاضی به سرویس‌گر آغاز می‌شود و به طور معمول تعدادی از مؤلفه‌های سرویس‌گر در پردازش تراکنش مؤثر بوده و معمولاً به مخزن مراجعه می‌شود. در نهایت با ارسال نتیجه توسط سرویس‌گر به متقاضی به پایان می‌رسد.

1- Pipe and Filter (P/F)
2- Layered (LYD)
3- Implicit Invocation
4- Dispatcher
5- Client/Server (C/S)

ز) سبک کارگزار[1] (BRK). مؤلفه‌های این سبک متقاضیان، سرویس‌گر(ها)، کارگزار، نماینده سمت متقاضی[2]، نماینده سمت سرویس‌گر[3] و مخزن است. کارگزار مسئول هماهنگی ارتباط و مجزا کردن متقاضیان از سرویس‌گرهاست. سرویس‌گرها خود را به کارگزار معرفی و سرویس‌های خود را از طریق واسط متد، در دسترس متقاضیان قرار می‌دهند. متقاضیان از خدمات سرویس‌گرها از طریق کارگزار استفاده می‌کنند. کارگزار، محل سرویس‌گر مناسب را تعیین، درخواست را به او ارسال، و نتیجه را به متقاضی برمی‌گرداند (باشمن و همکاران،۱۹۹۶: ۹۹). تراکنش در این سبک مشابه سبک متقاضی/سرویس‌گر است، اما مؤلفه‌های بیشتری در آن موثر می‌باشند.

ه) سبک شی گرا[4] (OO). در سبک شی‌گرا، نمایش داده و عملیات مرتبط با آن در یک شی محصور می‌شود. مؤلفه‌های این سبک، اشیا و تعامل اشیا از طریق فراخوانی تابع می‌باشد (شاع و گارلان، ۱۹۹۶: ۲۲). در سبک شی‌گرا هر مورد کاربری یک تراکنش تلقی می‌شود و زنجیری از اجرای متدی از تعدادی شی را شامل می‌شود.

۳- صفات کیفی نرم‌افزار

هدف هر نرم‌افزار، ارائه خدماتی است که بر عهده آن نرم‌افزار است، اما این خدمات بایستی با کیفیت تعیین شده، ارائه گردد و این کیفیت با معماری نرم‌افزار رابطه مستقیم دارد. بنابراین، طراحی مجدد نرم‌افزارها اغلب به سبب نقص در انجام وظایف محوله نیست، بلکه به دلیل عدم برخورداری از کیفیت لازم می‌باشد (بس، کلمنتس و کازمن، ۱۹۹۸: ۱۷).

سازمان بین‌المللی استاندارد، مدل ISO 9126 (ISO 9126, 2001) را برای سنجش کیفیت نرم‌افزار ارائه داده است. این مدل، کیفیت نرم‌افزار را در قالب شش صفت کیفی سطح بالا تعریف می‌کند. در ادامه ضمن تعریف هر صفت و زیر صفت‌های آن در مدل ایزو ۹۱۲۶، صفات کیفی قابل ارزیابی در سطح سبک‌های معماری تعیین می‌گردد.

1- Broker (BRK)
2- Client-Side Proxy
3- Server-Side Proxy
4- Object Oriented

وظیفه‌مندی[1]: توانایی سیستم نرم‌افزاری در تأمین وظایف مورد نیاز، در شرایط مشخص شده را توصیف می‌کند. زیرصفت‌های وظیفه‌مندی عبارتند از:

- مناسب بودن[2]: وجود و تناسب مجموعه‌ای از کارکردها برای وظایف مشخص شده را نشان می‌دهد.
- دقت: تولید نتایج صحیح را نشان می‌دهد.
- امنیت: قابلیت نرم‌افزار برای جلوگیری از دسترسی غیر مجاز، اعم از تصادفی و عمدی، به برنامه ها و داده ها را نشان می‌دهد.
- قابلیت همکاری: قابلیت نرم‌افزار برای تعامل با سیستم های مورد نظر را نشان می‌دهد.

بررسی‌ها نشان داده که "وظیفه‌مندی‌ها" با استفاده از ساختارها یا به عبارت دیگر، سبک‌های مختلف قابل تأمین است. به علاوه، "وظیفه‌مندی" نرم‌افزار طبیعت وابسته به معماری نرم‌افزار ندارد (بس، کلمنتس و کازمن، ۲۰۰۳: ۸۹). به عبارت دیگر، با هر ساختاری می‌توان وظیفه‌مندی مورد نظر را پیاده سازی کرد. بنابراین، نیازی به ارزیابی زیرصفت‌های "وظیفه‌مندی" در سطح سبک‌های معماری نمی‌باشد. اما با توجه به اهمیت امنیت در سیستم های نرم‌افزاری، ارزیابی سبک‌های معماری از دید امنیت مهم است و با توجه به اطلاعات موجود از سبک‌های معماری، امکان این ارزیابی وجود دارد.

قابلیت اعتماد[3]. مجموعه‌ای از زیرصفت‌هاست که قابلیت نرم‌افزار برای حفظ کارایی نرم‌افزار تحت شرایط مورد نظر در دوره زمانی معین، را نشان می‌دهد. این زیرصفت‌ها عبارتند از:

- بلوغ[4]: این زیرصفت، فرکانس شکست در اثر بروز عیب در نرم‌افزار، را مشخص می‌کند(ایزو ۹۱۲۶، ۲۰۰۱). بدیهی است که هرقدر فرکانس شکست نرم‌افزار کمتر باشد، میزان بلوغ نرم‌افزار بیشتر است.

1- Functionality
2- Suitability
3- Reliability
4- Maturity

• تحمل خطا: این زیر صفت، با قابلیت حفظ سطح کارایی مشخص شده در صورت بروز عیب در نرم‌افزار، مرتبط است.

• قابلیت ترمیم: این زیر صفت قابلیت برقراری مجدد سطح کارایی و ترمیم داده‌ها در صورت بروز شکست، و زمان و تلاش مورد نیاز برای آن را مشخص می‌کند (ایزو ۹۱۲۶، ۲۰۰۱).

در انتهای مرحله تحلیل نرم‌افزار اطلاعی از کیفیت، فرکانس شکست، قابلیت ترمیم و تحمل خطای مؤلفه‌های نرم‌افزار در سطح سبک های معماری در دست نیست. بنابراین، این زیر صفت‌ها در سطح سبک‌های معماری قابل ارزیابی نمی‌باشند. اما "قابلیت اعتماد" سبک‌های معماری بدون نگاه به زیر صفت‌های آن با استفاده از نمودار بلوکی قابلیت اعتماد[1] (شاه محمدی و جلیلی، ۱۳۸۶) و قابلیت اعتماد معماری نرم‌افزار (تراپتی و مال، ۲۰۰۵) نیز با استفاده از این رویکرد قابل ارزیابی است.

قابلیت استفاده[2]. عبارت است از: قابلیت درک، یادگیری، استفاده و جذاب بودن سیستم نرم‌افزاری برای کاربر، در هنگام استفاده از نرم‌افزار در شرایط مورد نظر. زیر صفت‌های آن عبارتند از:

• قابلیت درک: این زیر صفت، تلاش کاربر برای شناسایی مفهوم منطقی و قابلیت اجراء آن را منعکس می‌کند.

• قابلیت یادگیری: این زیر صفت، تلاش کاربر برای یادگیری سیستم را نشان می‌دهد.

• قابلیت عمل[3]: این زیر صفت، تلاش کاربر برای عمل و کنترل عمل سیستم را نشان می‌دهد.

اغلب زیر صفت‌های "قابلیت استفاده" به معماری نرم‌افزار وابسته نیست (ایزو ۹۱۲۶، ۲۰۰۱). به عبارت دیگر، این صفت و زیر صفت‌های آن بیشتر به مرحله پیاده سازی

1- Reliability Block Diagrams
2- Usability
3- Operability

نرم‌افزار مربوط است و با دید کاربر یعنی کیفیت محصول نهایی مرتبط می‌باشند. استاندارد، قابلیت استفاده را کیفیت در استفاده[1] در نظر می‌گیرد(ایزو ۹۱۲۶، ۲۰۰۱). از طرفی این صفت تأثیر مستقیمی روی معماری نرم‌افزار و در نتیجه سبک معماری ندارد (لوزویو و همکاران، ۲۰۰۴: ۴). بنابراین در سطح سبک‌های معماری قابل ارزیابی نمی‌باشد.

کارایی[2]: مجموعه‌ای از زیرصفت‌هاست که توانایی سیستم نرم‌افزاری در تأمین کارایی مناسب، از نظر میزان منابع استفاده شده، در شرایط بیان شده، را نشان می‌دهد. این زیر صفت‌ها عبارتند از:

- رفتار زمانی: بیانگر پاسخ و زمان پردازش و نرخ توان عملیاتی در انجام کارکردهای نرم‌افزار است.
- رفتار منبع: بیانگر مقدار منابع و مدت استفاده از منابع در انجام کارکردهاست.

استاندارد ایزو ۹۱۲۶، کارایی را کیفیت در استفاده از نرم‌افزار، در نظر می‌گیرد (ایزو ۹۱۲۶، ۲۰۰۱). اما با توجه به اطلاعات موجود از سبک‌های معماری امکان ارزیابی مبتنی بر سناریو آنها وجود دارد. با افزایش ظرفیت حافظه‌ها، تأثیر رفتار منبع در مقایسه با رفتار/کارایی زمانی نرم‌افزار قابل توجه نیست. در نتیجه کارایی سبک‌های معماری از دید زمانی ارزیابی می‌شود.

قابلیت نگهداری: توانایی سیستم نرم‌افزاری برای اصلاح را نشان می‌دهد. اصلاح شامل تصحیح، بهبود یا تطبیق نرم‌افزار با تغییر محیط، خواسته‌ها و وظایف است. زیر زیرصفت‌های قابلیت نگهداری عبارتند از:

- قابلیت تحلیل: تلاش لازم برای تشخیص نقایص یا سبب شکست ها یا تلاش لازم برای شناسایی بخش های اصلاح شده نرم افزار را نشان می‌دهد.

1- Quality-In-Use
2- Efficiency

• قابلیت تغییر: تلاش لازم برای اصلاح، برطرف کردن عیب یا تغییر محیطی نرم افزار، را نشان می‌دهد.
• پایداری: ریسک اثر اصلاحات غیر منتظره در نرم‌افزار را منعکس می‌کند.
• قابلیت آزمون: تلاش لازم برای اعتبارسنجی نرم‌افزار اصلاح شده را منعکس می‌کند.
قابلیت نگهداری در سطح سبک‌های رایج معماری با استفاده از روش مبتنی بر سناریو قابل ارزیابی است (شاه محمدی و جلیلی،۱۳۸۷).

قابلیت حمل: توانایی سیستم نرم‌افزاری برای انتقال از یک محیط(سازمانی، سخت افزاری یا نرم‌افزاری) به محیط دیگر را نشان می‌دهد. زیر صفت‌های قابلیت حمل عبارتند از:
• قابلیت تطبیق[1]: منظور از قابلیت تطبیق این است که نرم‌افزار بدون استفاده از وسایل معمول در نظر گرفته شده برای آن، می‌تواند با محیط‌های مختلف مشخص شده، تطبیق پیدا کند.
• قابلیت‌نصب: تلاش لازم برای نصب نرم‌افزار در محیط مشخص شده را نشان می‌دهد.
• سازش[2]: مقدار تطبیق نرم‌افزار را با استاندارد یا قراردادهای مرتبط با قابلیت حمل نشان می‌دهد.
• قابلیت‌جایگزینی[3]: توانایی محصول نرم‌افزاری برای استفاده از آن در مکان سایر محصولات نرم‌افزاری مشخص شده برای هدف واحد در محیط واحدی است.
زیر صفت‌های قابلیت نصب و سازش به مرحله پیاده سازی نرم‌افزار توجه دارند و با اطلاعات موجود در سطح سبک‌های معماری قابل سنجش نمی‌باشند. اما با توجه به این که قابلیت حمل به عنوان اصلاح سکو، تلقی می‌شود (بس، کلمنتس و کازمن، ۲۰۰۳: ۱۰۹). نتایج ارزیابی صفت کیفی قابلیت نگهداری، برای ارزیابی قابلیت حمل

1- Adaptability
2- Co-existence
3- Replaceability

نرم‌افزار نیز قابل استفاده است.
به این ترتیب از مدل کیفی ایزو ۹۱۲۶، صفات کیفی: ۱) قابلیت اعتماد، ۲) قابلیت نگهداری، ۳) کارایی، ۴) قابلیت حمل(براساس قابلیت نگهداری) و ۵) امنیت در سطح سبک‌های معماری قابل ارزیابی می‌باشند.

۴- سیستم اطلاعاتی

سیستم اطلاعاتی، سیستمی است که داده یا اطلاعات مورد نیاز کاربران سازمان را فراهم می‌کند. برخی سیستم اطلاعاتی را از دید سیستمی تعریف کرده‌اند. با این دیدگاه، سیستم اطلاعاتی مجموعه ای از اجزای مرتبط به هم است که اطلاعاتی را برای پشتیبانی از تصمیم در سازمان جمع آوری، پردازش، ذخیره و توزیع می‌کند، ضمن اینکه به مدیران و کارکنان در تحلیل مسائل و تجسم موضوعات پیچیده و خلق محصولات جدید کمک می‌کنند. به تعبیر دیگر، یک سیستم اطلاعاتی، یک سیستم نرم‌افزاری است که داده ها و تراکنش های مرتبط با هدف کاری[1] را جمع‌آوری، پردازش و ذخیره می‌کند.

نقش سیستم های اطلاعاتی در برقراری نظم و امنیت.

در چند سال اخیر استفاده از سیستم‌های اطلاعاتی در ناجا ضمن افزایش دقت، سرعت و صحت انجام مأموریت‌ها، توان ناجا را در انجام ماموریت ها افزایش داده و حتی فرصت پرداختن به امور راهبردی در برخی حوزه‌ها را نیز فراهم کرده است. این امر در افزایش امنیت اجتماعی و احساس امنیت شهروندان نقش مؤثری داشته است. سیستم فوریت‌های پلیسی ۱۱۰ و سیستم جامع انتظامی نمونه هایی از این سیستم‌ها می باشند.

سیستم فوریت‌های پلیسی ۱۱۰: این سیستم نیازهای فوری شهروندان به پلیس را در کوتاه‌ترین زمان ممکن پاسخ می دهد و عملکرد پلیس در پاسخ به این نیازها به

1- Business

منظور بررسی عملکرد پلیس در پایگاه‌های مختلف ثبت و ضبط می‌شود. با راه‌اندازی بانک اطلاعات جامع ۱۱۰، کلیه نیازهای پلیسی شهروندان و چگونگی عملکرد پلیس بطور یکپارچه ذخیره می‌گردد و جستجوهای تحلیلی و آماری متنوعی فراهم می‌گردد و راه بهبود و اصلاح کیفیت عملکرد پلیس را هموار می‌سازد. با وقوع جرم و تماس شهروندان با کاربران این سیستم، با استفاده از سیستم اطلاعات جغرافیای متصل به این سیستم، نزدیک‌ترین کلانتری به محل ارتکاب جرم شناسایی و حضور در محل و برخورد مناسب با مجرم/مجرمان، به آنان ابلاغ می‌شود(بکارگیری فناوری اطلاعات در ناجا، ۱۳۸۶). ایجاد بانک اطلاعاتی درخواست‌های مردم از پلیس، دستیابی به جغرافیای جرم، توزیع زمانی آن، از اهداف این سیستم است (به‌کارگیری فناوری اطلاعات در ناجا، ۱۳۸۶: ۱۹).

سیستم جامع انتظامی: این سیستم امور مختلف وظایف انتظامی را عهده دار است و امکان نظارت و کنترل بر عملکرد مأموران و یگان‌های انتظامی دارد و تأثیر قابل توجهی در صحت، دقت و سرعت اقدامات پلیس داشته است.

پیشینه تحقیق

تحقیقات و مطالعاتی که مرتبط با موضوع تحقیق انجام شده است به طور مختصر به آن‌ها اشاره می‌کنیم.

- شاع و گارلان از بنیانگذاران طراحی معماری نرم‌افزار مبتنی بر سبک معماری بودند و معتقد بودند که سبک معماری نه تنها بر توصیف نرم‌افزار، بلکه برتجزیه آن به مؤلفه ها و کارایی آن اثر دارد (شاع و گارلان، ۱۹۹۶: ۳۱-۲۰). این روش تنها به ارائه رهنمودهای کلی در انتخاب سبک معماری اکتفا می‌کند و به عنوان یک روش اصولی برای انتخاب سبک معماری قابل استفاده نیست.
- شاع و همکاران سیستمی برای انتخاب سبک معماری ارائه داده اند (شاع و فار، ۲۰۰۴)، امّا صفات کیفی سبک‌های معماری رتبه بندی نشده و شیوه استدلال و انتخاب سبک معماری مشخص نیست. در مقابل، مدل پیشنهادی، مبتنی بر ارزیابی و ترکیب نتایج ارزیابی صفات کیفی سبک های معماری با استفاده از روش AHP

برای تعیین رتبه نهایی سبک‌های معماری برای یک سیستم خاص است. در مدل پیشنهادی به اندازه نرم‌افزار و تعداد مؤلفه‌های مؤثر در انجام تراکنش نیز توجه شده است.

• رضا و همکاران روشی برای تعیین ارتباط ۶ سبک معماری و صفات کیفی پیشنهاد داده‌اند (رضا و گرانت، ۲۰۰۵). ارتباط صفات کیفی معماری با اصول طراحی، و برخی از سبک‌های معماری، با استفاده از چارچوب مبتنی بر درخت تعیین شده است. سبک‌ها در ریشه درخت(ریشه برای هر سبک یک شاخه دارد)، اصول و تصمیمات طراحی در گره‌های میانی و صفات کیفی در برگ ها قرار گرفته‌اند. ریشه درخت دارای ۶ شاخه است. در این روش، تعیین ارتباط سبک‌های معماری با صفات کیفی نرم‌افزار خیلی تقریبی بوده و مبتنی بر ارزیابی سبک‌های معماری نیست. در مقابل روش پیشنهادی مبتنی بر ارزیابی است و نسبت به این روش از دقت بالایی برخوردار است، ضمن اینکه ۸ سبک معماری را در نظر می‌گیرد، و به اندازه نرم‌افزار و تعداد مؤلفه‌های مؤثر در انجام تراکنش نیز توجه دارد.

• لیو و همکاران چارچوبی برای استخراج تصمیمات معماری از نیازها، و یک اجراء مبتنی بر قانون با توانمندی استدلال خودکار، برای توصیه سبک معماری ارائه داده‌اند (لیو و ایستربروک، ۲۰۰۳). مشکل چارچوب پیشنهادی این است که به صفات کیفی سبک‌های معماری در موقع انتخاب آنها، توجه نمی کند.

• معاون و همکاران (۲۰۰۹)، روشی را برای انتخاب سبک معماری از میان چند سبک کاندید با استفاده از روش انتگرال فازی ارائه داده‌اند. مزیت روش پیشنهادی توجه به تعامل صفات کیفی در تعیین رتبه سبک‌های معماری است. اما استفاده از این روش منوط به شناخت تأثیر کمی سبک‌های معماری بر روی صفات کیفی مختلف است که در آن تحقیق انجام نشده است، ضمن اینکه در نظر گرفتن تعامل میان صفات کیفی نیازمند شناخت تعامل صفات کیفی با یکدیگر است که مبنای علمی یا تجربی برای تعامل صفات کیفی ارائه نشده است و به صورت موردی عمل شده است.

روش‌شناسی پژوهش

۱- مدل پیشنهادی برای انتخاب سبک معماری

در این مدل سبک معماری در پایان مرحله تحلیل خواسته‌ها از نرم‌افزار و براساس متدلوژیRUP[1] توصیه می‌شود. در این مرحله شناخت کافی از مشخصه‌ها و نیازهای غیر وظیفه‌ای یا صفات کیفی سیستم درخواستی وجود دارد. مدل پیشنهادی برای انتخاب سبک معماری در شکل ۱ ارائه شده است. این مدل شامل دو مرحله اصلی است: در مرحله اول با توجه به مشخصه‌های مهم سیستم سبک‌های معماری فاقد مشخصه ها حذف می‌شوند. در مرحله دوم با توجه به صفات کیفی مورد نظر سیستم و تقدم انها، سبک‌های معماری رتبه‌بندی می‌شوند. مراحل این مدل عبارتند از:

مرحله اول: این مرحله شامل زیر مراحل زیر است.

۱- اخذ مشخصه‌های مهم سیستم: در این مرحله مشخصه‌های مهم سیستم با شناخت موجود از سیستم تعیین می‌گردد. این مشخصه ها شامل مواردی مانند واکنشی بودن/نبودن سیستم(سیستم واکنشی، سیستمی است که در تعامل مداوم با محیط خود است. درخواست‌هایی را می پذیرد و پاسخ هایی را تولید می‌کند (مانا و نیوعلی، ۱۹۹۲)، محاوره‌ای بودن/نبودن(عوامل انسانی با سیستم تعامل دارند) سیستم است.

۲- پالایش سبک‌های معماری سیستم: در این مرحله با توجه به مشخصه‌های مهم سیستم، سبک‌های معماری فاقد این مشخصه‌ها حذف می‌شوند.

مرحله دوم: در این مرحله سبک‌های معماری باقیمانده از مرحله قبل به شرح زیر رتبه بندی می‌شوند:

۱- اخذ صفات کیفی سیستم: در این مرحله، با توجه به وظایف سیستم که در قالب موردهای کاربری بیان می‌شود، صفات کیفی مهم سیستم استخراج می‌شود.

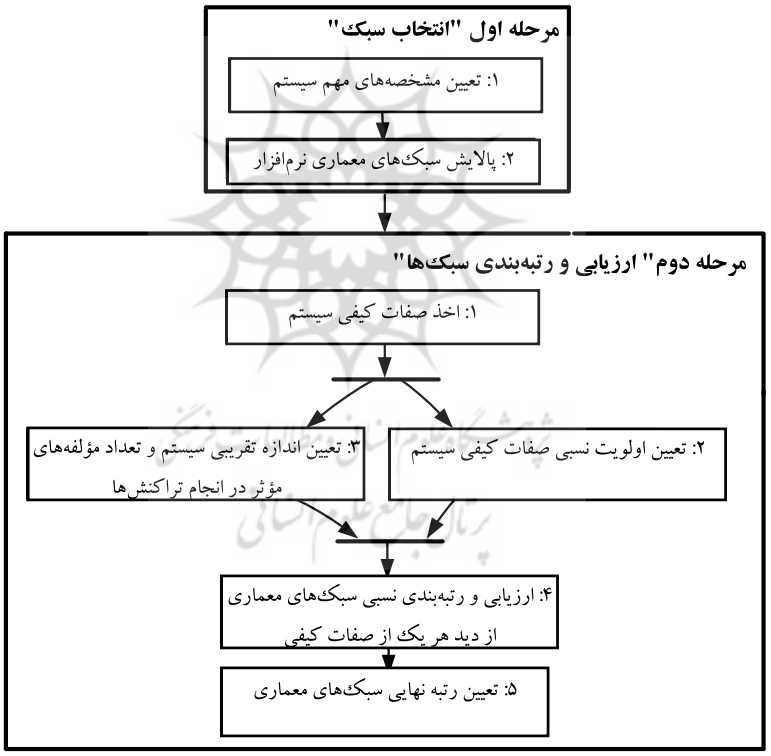
۲-تعیین اولویت نسبی صفات کیفی مهم سیستم: در این مرحله براساس

1- Rational Unified Process

اهمیت صفات کیفی و انجام مقایسه‌های زوجی با استفاده از جدول مقایسه زوجی روش AHP (ساعتی و وارگاس، ۲۰۰۱)، ماتریس مقایسه صفات کیفی را تشکیل داده و پس از انجام محاسبات روش AHP، اولویت نسبی صفات کیفی سیستم درخواستی تعیین می‌شود.

۳- تعیین اندازه تقریبی سیستم و تعداد مؤلفه‌های مؤثر در انجام تراکنش:

در این مرحله با توجه به تعداد موجودیت‌های[1] سیستم و وظایف سیستم (با استفاده از موردهای کاربری سیستم در صورت وجود)، تعداد تقریبی مؤلفه‌های مورد نیاز در سبک‌های معماری و تعداد مؤلفه‌های مؤثر در انجام یک تراکنش، تعیین می‌شود.



شکل ۱. مدل پیشنهادی برای انتخاب سبک معماری

1- Entities

۴- ارزیابی و رتبه بندی نسبی سبک‌های معماری از دید صـفات کیفـی: در بخش قبل، صفات کیفی قابل ارزیابی در سطح سبک های معمـاری، براسـاس مـدل کیفی ایزو ۹۱۲۶ تعیین شد. در این مرحله، سبک‌های معماری ارزیابی و رتبه‌بنـدی می‌شوند. رتبه سبک‌های معماری از دید قابلیت اعتماد و کارایی به تعداد مؤلفه‌هـای مؤثر در انجام تراکنش وابسته است و همان‌گونه که خواهیم دید، رتبـه سـبک‌هـای معماری از دید این صفات کیفی، با توجه به تعداد مؤلفه‌های مؤثر در انجام تـراکنش در هر یک از جداول مرتبط با صفات کیفی ارائه می‌شـود. رتبـه قابلیـت نگهـداری و قابلیت حمل سبک‌های معماری نیز با توجه به اندازه نرم‌افزار ارائه می‌گردد. بنابراین، در این مرحله، با توجه به تعداد تقریبـی مؤلفـه‌هـای سـبک‌هـای معمـاری و تعـداد مؤلفه‌های مؤثر در انجام تراکنش برای سیستم اطلاعاتی درخواستی، کـه در مرحلـه قبل تعیین شد، از جدول‌های مربوط به رتبه سبک‌های معماری از دید صفات کیفی: قابلیت نگهداری، قابلیت حمل، قابلیت اعتمـاد، و کـارایی، رتبـه نـسبی سـبک‌هـای معماری استخراج می‌گردد.

۵- تعیین رتبه نهایی سبک‌های‌معماری.

در این مرحله، با توجه به اولویت نسبی صفات کیفی محاسـبه شـده و رتبـه نـسبی سبک‌های معماری از دید صفات کیفی مختلف، با استفاده از رابطـه (۱) روش AHP، رتبه نهایی سبک‌های معماری محاسبه می‌شود.

پژوهشگاه علوم انسانی و مطالعات فرهنگی
پرتال جامع علوم انسانی

یافته های تحقیق

۱- ارزیابی سبک‌های معماری از دید صفات کیفی

در این مرحله چگونگی ارزیابی سبک‌های معمـاری از دیـد صـفات کیفـی پیـشنهاد می‌شود.

با توجه به اینکه انتخاب سبک‌های معماری منوط به شناخت ابعاد کیفی سبک‌هـای معماری است و با توجه به اینکه سبک‌های معماری از دیـد قابلیـت نگهـداری(شـاه محمدی و جلیلی،۱۳۸۷) و قابلیت اعتمـاد (شـاه محمـدی و جلیلـی، ۱۳۸۶)، قـبلاً

ارزیابی شده‌اند، در این بخش مرور اجمالی بر ارزیابی سبک‌های معماری از دید قابلیت نگهداری، و قابلیت اعتماد، خواهیم داشت. با توجه به اینکه قابلیت حمل به نوعی اصلاح سکو تلقی می گردد؛ بنابراین، نتایج ارزیابی قابلیت نگهداری قابل تعمیم به ارزیابی قابل حمل سبک‌های معماری است. در نهایت، به لحاظ محدودیت فضا از دو صفت کیفی باقیمانده یعنی امنیت و کارایی، تنها صفت کیفی کارایی سبک های معماری ارزیابی می شود.

ارزیابی سبک‌های معماری از دید قابلیت نگهداری

شاه محمدی، جلیلی و هاشمی نژاد (۱۳۸۷)، سبک‌های معماری را از دید صفت کیفی قابلیت نگهداری با استفاده از روش مبتنی بر سناریو ارزیابی کردیم. ابتدا، تأثیر سناریوهای افزودن، حذف و اصلاح وظیفه‌مندی و اصلاح ساختار داده‌ها، بر سبک‌های معماری، شامل تعداد مؤلفه‌های متأثر شده و نوع تأثیر، تعیین شد. سپس با توجه به تعداد مؤلفه‌های متأثر شده و نوع تأثیر، هزینه اعمال هر سناریو بر سبک‌های معماری، تخمین زده شد و در نهایت با توجه به اهمیت سناریوها و با استفاده از روش AHP، ارزیابی مبتنی بر سناریو از کیفی به کمی تبدیل گردید. در کمی‌سازی تأثیر سناریوها بر سبک‌های معماری، تعداد مؤلفه‌های سبک‌ها بطور متوسط ۵ و در سبک شی گرا ۳۵ در نظر گرفت. با توجه به اینکه در سبک شی‌گرا، تعداد کلاس‌ها(n_o) و در سایرسبک‌های معماری تعداد مؤلفه‌ها(n) متناظر با اندازه نرم‌افزار است، برای بررسی تأثیر اندازه نرم‌افزار(تعداد مؤلفه‌ها)، بر رتبه قابلیت نگهداری سبک‌های معماری، تعداد مؤلفه‌های نرم‌افزار به ترتیب ۴،۳، ۵، ۶، ۷، ۸ و ۹ و متناسب با آن تعداد کلاس‌ها در سبک شی‌گرا به ترتیب ۲۱، ۲۸، ۳۵، ۴۲، ۴۹، ۵۶ و ۶۳ درنظر گرفته شد تا تأثیر عامل اندازه نرم‌افزار بر رتبه‌بندی سبک‌های معماری نیز در نظر گرفته شود. آنگاه به روش AHP، رتبه سبک های معماری از دید قابلیت نگهداری محاسبه و در جدول ۲ درج گردید.

جدول۲. رتبه سبک‌های معماری از دید قابلیت نگهداری

نام سبک	نماد	n=۳	n=۴	n=۵	n=۶	n=۷	n=۸	n=۹
		no=۲۱	no=۲۸	no=۳۵	no=۴۲	no=۴۹	no=۵۶	no=۶۳
مخزن	RPS	۷۰	۶۹	۶۸	۶۸	۶۸	۶۷	۶۷
تخته سیاه	BKB	۴۴	۴۴	۴۳	۴۳	۴۲	۴۲	۴۲
لوله و فیلتر	P/F	۱۷۵	۱۸۲	۱۸۶	۱۸۹	۱۹۱	۱۹۳	۱۹۴
لایه‌ای	LYD	۲۱۰	۲۱۸	۲۲۳	۲۲۷	۲۲۹	۲۳۱	۲۳۳
فراخوانی ضمنی	I/I	۱۹۶	۱۹۶	۱۹۶	۱۹۶	۱۹۷	۱۹۷	۱۹۷
متقاضی/سرویس‌گر	C/S	۱۰۹	۱۰۳	۹۸	۹۵	۹۳	۹۱	۹۰
کارگزار	BRK	۷۳	۶۴	۵۹	۵۶	۵۳	۵۱	۴۹
شی گرا	OO	۱۲۲	۱۲۴	۱۲۵	۱۲۶	۱۲۷	۱۲۷	۱۲۸

ارزیابی سبک‌های معماری از دید قابلیت اعتماد

شاه محمدی و جلیلی (۱۳۸۶)، سبک‌های معماری را از دید قابلیت اعتماد براساس رویکرد دیاگرام بلوکی قابلیت اعتماد(RBD) با توجه به اندازه نرم‌افزار(تعداد مؤلفه‌ها) مورد ارزیابی قرار دادیم. RBD یک نمایش بلوکی از مؤلفه‌های نرم‌افزار و بررسی قابلیت اعتماد آنها ارائه می‌دهد (تراپتی و مال، ۲۰۰۰). پیش بینی قابلیت اعتماد سبک معماری از طریق بررسی قابلیت اعتماد مؤلفه‌های تشکیل دهنده آن قابل انجام است. برای ساخت یک RBD باید پیکربندی مؤلفه‌های نرم‌افزار تعیین شود. شکل ۲ یک پیکربندی سری یا ترتیبی را نشان می دهد. در پیکربندی ترتیبی، شکست هر مؤلفه باعث شکست نرم‌افزار می‌گردد.

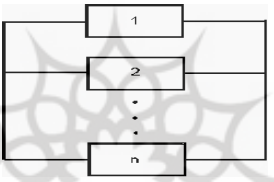
شکل ۲. دیاگرام بلوکی پیکربندی ترتیبی (تراپتی و مال، ۲۰۰۰)

قابلیت اعتماد نرم‌افزار با پیکربندی ترتیبی برابر احتمال این است که همه مؤلفه‌های نرم‌افزار درست عمل کنند. لذا، برای موفقیت نرم‌افزار باید همه n مؤلفه موفق باشند. بنابراین قابلیت اعتماد نرم‌افزار از رابطه (۲) محاسبه می گردد.

$$R_s = P(X_1 \cap X_2 \cap \dots \cap X_n) = P(X_1) \quad (۲)$$
$$P(X_2|\, X_1)\, P(X_3|\, X_1\, X_2) \dots P(X_n|\, X_1\, X_{2\dots}\, X_{n-1})$$

در این رابطه R_s قابلیت اعتماد نرم‌افزار، X_i رخداد فعال بودن مؤلفه i-ام و $P(X_i)$ احتمال فعال بودن مولفه i-ام می‌باشد. اگر مؤلفه‌ها مستقل از یکدیگر باشند، رابطه (۲) به رابطه (۳) تبدیل می‌شود، که R_i قابلیت اعتماد مولفه i-ام می‌باشد. در نرم‌افزارهای با پیکربندی موازی(شکل ۳)، برای موفقیت نرم‌افزار حداقل باید یکی از مؤلفه ها درست عمل کند.

$$Rs = P(X_1)\, P(X_2) \dots P(X_n) = \prod_{i=1}^{n} P(X_i) = \prod_{i=1}^{n} R_i \quad (۳)$$



شکل۳. دیاگرام بلوکی پیکربندی موازی (تراپتی و مال، ۲۰۰۰)

احتمال شکست نرم‌افزاری با n مؤلفه موازی که از نظر آماری مستقل باشند برابر احتمال این است که همه مؤلفه‌های نرم‌افزار دچار شکست شوند. به عبارت دیگر، اگر حداقل یک مؤلفه درست عمل کند، آنگاه نرم‌افزار درست عمل خواهد کرد. عدم اطمینان نرم‌افزار از رابطه (۴) محاسبه می‌گردد. در این رابطه Q_s عدم اطمینان نرم‌افزار، Xi پیشامد شکست مؤلفه i-ام و $P(Xi)$ احتمال شکست مؤلفه i-ام است. اگر مؤلفه‌ها از یکدیگر مستقل باشند، رابطه (۴) به رابطه (۵) تبدیل می‌شود.

$$Q_s = P(X_1 \cap X_2 \cap \dots \cap X_n) = P(X_1)\, P(X_2|\, X_1)\, P(X_3|\, X_1\, X_2) \dots P(X_n|\, X_1\, X_{2\dots}\, X_{n-1}) \quad (۴)$$

در این رابطه Q_i عدم اطمینان مؤلفه i-ام است. با توجه به رابطه (۵)، قابلیت اعتماد نرم افزار با پیکربندی موازی از رابطه (۶) محاسبه می‌شود.

$$Qs = P(X_1)\, P(X_2) \dots P(X_n) = \prod_{i=1}^{n} P(X_i) = \prod_{i=1}^{n} Q_i \quad (۵)$$

$$Rs = 1 - Qs = (Q_1 \cdot Q_2 \dots Q_n) = 1 - [(1-R_1)(1-R_2)\dots(1-R_n)] = 1 - \prod_{i=1}^{n}(1-R_i) \quad (۶)$$

قابلیت اعتماد نرم‌افزارهای با پیکربندی ترکیبی به طور مشابه قابل محاسبه است (تراپتی و مال، ۲۰۰۰).

در توسعه کار قبلی در ارزیابی قابلیت اعتماد سبک‌های معماری (شاه محمدی و جلیلی، ۱۳۸۶)، با انجام تغییرات: ۱) قابلیت اعتماد سبک‌های معماری با توجه به مفهوم تراکنش و تعداد مؤلفه‌های مؤثر در انجام تراکنش به جای اندازه نرم‌افزار، محاسبه شود؛ ۲) مؤلفه سرویس‌گر در سبک‌های متقاضی/سرویس‌گر و کارگزار به جای مؤلفه واحد پیچیده، مرکب از تعدادی مؤلفه در نظر گرفته شود؛ ۳) قابلیت اعتماد سبک شی‌گرا با توجه به مسیرهای مختلف اجرای تراکنش‌ها و مدل سازی آن به جای مدل AND، به صورت ترکیبی AND وOR محاسبه گردد (شاه محمدی و جلیلی، ۱۳۸۶)؛ قابلیت اعتماد سبک شی‌گرا براساس قابلیت اعتماد یک مورد کاربری محاسبه شد. اما از آنجا که سبک شی‌گرا شامل تعدادی مورد کاربری است و با شکست یک مورد کاربری، سیستم با توان کمتر به کار خود ادامه می‌دهد و از طرفی تعدادی از کلاس‌ها در اغلب موردهای کاربری مشترک می‌باشند، قابلیت اعتماد این سبک به صورت سری-موازی یا ترکیبی(AND و OR) است. بررسی قابلیت اعتماد بخش موازی این سبک با تعداد مختلف کلاس‌ها نشان داد که قابلیت اعتماد این بخش نزدیک به ۱ است. در نتیجه قابلیت اعتماد سبک شی‌گرا معادل قابلیت اعتماد بخش سری(AND) می باشد، مجدداً قابلیت اعتماد سبک‌های معماری با رویکرد دیاگرام بلوکی قابلیت اعتماد ارزیابی شد. جدول ۲ رابطه قابلیت اعتماد هریک از سبک‌های معماری را با توجه به تعداد مولفه‌های موثر در انجام تراکنش نشان می‌دهد.

جدول۳ رابطه

جدول۳. قابلیت اعتماد سبک‌های معماری

نام سبک	نماد	قابلیت اعتماد
مخزن	RPS	$R_i . R_{rps}$
تخته سیاه	BKB	$R_c R_{bkb} R_{kr}$

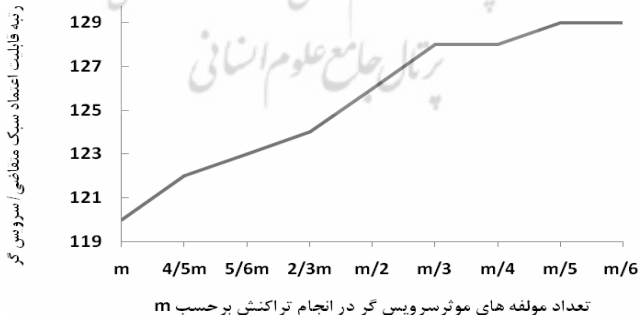
لوله و فیلتر	P/F	$\prod_{i=1}^{m} R_i$
لایه‌ای	LY D	$\prod_{i=1}^{m} R_i$
فراخوانی ضمنی	I/I	$R_d R_i$
متقاضی/سرویس گر	C/S	$R_{m/2}..R_2 R_1 R_{rps}$
کارگزار	BR K	$R_{csp} R_{brk} R_{ssp} R_{m/2}...$
شی گراء	OO	$\prod_{i=1}^{c} R_j$

در رابطه قابلیت اعتماد اغلب سبک‌های معماری، پارامتر m که نشانگر تعداد مؤلفه‌های مؤثر در انجام تراکنش است، وجود دارد. بنابراین برای بررسی تأثیر تعداد مؤلفه‌های مؤثر در انجام تراکنش در قابلیت اعتماد سبک‌های معماری، مقدار m، ۲، ۳، ۴، ۵، ۶ و ۷ در نظر گرفته می‌شود. مؤلفه‌های سبک شی‌گرا، یعنی کلاس‌ها نسبت به مؤلفه‌های سایر سبک‌های معماری ریزدانه‌تر می‌باشند، با این تقریب که به طور متوسط در هر مؤلفه ۳ کلاس در انجام تراکنش مؤثر باشند، متناظر با تعداد مؤلفه‌های مؤثر در انجام تراکنش، در سایر سبک‌های معماری، تعداد کلاس‌های مؤثر در انجام تراکنش(no) ۶، ۹، ۱۲، ۱۵، ۱۸ و ۲۱ در نظر گرفته می‌شود. قابلیت اعتماد هر مؤلفه/کلاس را ۰,۹۸ در نظر می‌گیریم. در رابطه قابلیت اعتماد سبک شی گرا، c تعداد کلاس‌های مشترک در موردهای کاربری است. شکل ۴، نمودار تغییر رتبه قابلیت اعتماد سبک شی‌گرا را برحسب درصد کلاس‌های مشترک در موردهای کاربری برای no=۱۵ را نشان می‌دهد. مطابق نمودار با افزایش این درصد، رتبه قابلیت اعتماد سبک کاهش می‌یابد. مؤلفان براساس تجارب خود در تولید سیستم‌های نرم‌افزاری، درصد کلاس‌های مشترک در موردهای کاربری را ۲۰ در نظر گرفتند.

شکل ۴. نمودار تغییر رتبه قابلیت اعتماد سبک شی‌گرا برحسب درصد کلاس های مشترک موردهای کاربری برای n_o=۱۵

در سبک‌های متقاضی/سرویس‌گر و کارگزار، در انجام تراکنش بخشی از m مؤلفه مؤثر می‌باشند. شکل ۵، نمودار تغییر رتبه قابلیت اعتماد سبک متقاضی/سرویس‌گر را برحسب تعداد مؤلفه‌هایی از سرویس‌گر که در انجام تراکنش مؤثر می باشند را برحسب m برای m=۵ نشان می‌دهد. مطابق نمودار با افزایش تعداد این مؤلفه‌ها، رتبه قابلیت اعتماد سبک کاهش می‌یابد. این مقدار نیز براساس تجارب مؤلفان در تولید سیستم‌های نرم‌افزاری، m/۲ در نظر گرفته شد. تولیدکنندگان نرم‌افزار براساس تجارب گذشته خود در تولید نرم‌افزار، پارامترهای درصد کلاس‌های مشترک در موردهای کاربری و تعداد مؤلفه‌های مؤثر سرویس‌گر در انجام تراکنش را تعیین می‌کنند.

پژوهشگاه علوم انسانی و مطالعات فرهنگی
پرتال جامع علوم انسانی



شکل ۵. نمودار تغییر رتبه قابلیت اعتماد سبک متقاضی/سرویس‌گر برحسب تعداد مؤلفه‌های مؤثر سرویس‌گر در انجام تراکنش برای m=۵

به ازاء تعداد مختلف مؤلفه‌های مؤثر در انجام تراکنش و براساس رابطه قابلیت اعتماد سبک‌های معماری در جدول ۳ و با توجه به مقدار قابلیت اعتماد هر مؤلفـه/کـلاس، مقادیر قابلیت اعتماد هریک از سبک‌های معماری محاسبه و در جدول ۴ درج شـده است. رتبه قابلیت اعتماد سبک‌های معماری، از رابطـه (۷) محاسـبه و در جـدول ۵ درج شده است.

جدول۴. ارزش قابلیت اعتماد سبک‌های معماری

نام سبک	نماد	m=۲	m=۳	m=۴	m=۵	m=۶	m=۷
		no =۶	no =۹	no =۱۲	no =۱۵	no =۱۸	no =۲۱
مخزن	RPS	۰,۹۶	۰,۹۶	۰,۹۶	۰,۹۶	۰,۹۶	۰,۹۶
تخته سیاه	BKB	۰,۹۴	۰,۹۴	۰,۹۴	۰,۹۴	۰,۹۴	۰,۹۴
لوله و فیلتر	P/F	۰,۹۶	۰,۹۴۱	۰,۹۲۲	۰,۹۰۴	۰,۸۸۶	۰,۸۶۸
لایه‌ای	LYD	۰,۹۶	۰,۹۴۱	۰,۹۲۲	۰,۹۰۴	۰,۸۸۶	۰,۸۶۸
فراخوانی ضمنی	I/I	۰,۹۶	۰,۹۶	۰,۹۶	۰,۹۶	۰,۹۶	۰,۹۶
متقاضی/سرویس‌گر	C/S	۰,۹۶	۰,۹۵۱	۰,۹۴۱	۰,۹۳۲	۰,۹۲۲	۰,۹۱۳
کارگزار	BRK	۰,۹۰۴	۰,۸۹۵	۰,۸۸۶	۰,۸۷۷	۰,۸۶۸	۰,۸۵۹
شی گراء	OO	۰,۹۷۶	۰,۹۶۴	۰,۹۵۳	۰,۹۴۱	۰,۹۳	۰,۹۱۹

در این رابطه، x_{ij} ارزش قابلیت اعتماد سبک در ردیف i–ام و ستون j–ام جـدول ۴، و مخرج رابطه مجموع ارزش قابلیت اعتماد ستون j–ام است. از ضریب ۱۰۰۰ در رابطه (۷) برای وضوح رتبه سبک‌ها استفاده شده است.

$$R_{ij} = \frac{x_{ij}}{\sum_{i=1}^{8} x_{ij}} * 1000 \qquad (۷)$$

جدول ۵. رتبه سبک‌های معماری از دید قابلیت اعتماد

نام سبک	نماد	m=۲	m=۳	m=۴	m=۵	m=۶	m=۷
		no =۶	no =۹	no =۱۲	no =۱۵	no =۱۸	no=۲۱
مخزن	RPS	۱۲۶	۱۲۷	۱۲۸	۱۲۹	۱۳۱	۱۳۲
تخته سیاه	BKB	۱۲۳	۱۲۴	۱۲۶	۱۲۷	۱۲۸	۱۲۹
لوله و فیلتر	P/F	۱۲۶	۱۲۵	۱۲۳	۱۲۲	۱۲۱	۱۱۹

لایه‌ای	LYD	۱۲۶	۱۲۵	۱۲۳	۱۲۲	۱۲۱	۱۱۹
فراخوانی ضمنی	I/I	۱۲۶	۱۲۷	۱۲۸	۱۲۹	۱۳۱	۱۳۲
متقاضی/سرویس‌گر	C/S	۱۲۶	۱۲۶	۱۲۶	۱۲۶	۱۲۵	۱۲۵
کارگزار	BRK	۱۱۹	۱۱۹	۱۱۸	۱۱۸	۱۱۸	۱۱۸
شی گراء	OO	۱۲۸	۱۲۸	۱۲۷	۱۲۷	۱۲۶	۱۲۶

ارزیابی سبک‌های معماری از دید کارآیی

سناریو عمومی ارائه شده توسط Bass و همکاران برای ارزیابی کارایی معماری نرم‌افزار، وقوع رویداد و پاسخ به آن است (بس، کلمنتس و کازمن، ۲۰۰۳: ۹۹). یک تعبیر دیگر، این سناریو، تعداد تراکنش های پردازش شده در واحد زمان و به بیان دیگر، تعداد مؤلفه‌های مؤثر در انجام تراکنش می‌باشد. قدر مسلم هر قدر تعداد مؤلفه‌های مؤثر در انجام تراکنش کمتر باشد، پاسخ معماری به رویداد یا به تعبیر دیگر تراکنش بهتر می‌شود. با توجه به تعریف تراکنش در سبک‌های معماری مختلف در بخش مبانی نظری، جدول ۶، وضعیت سبک‌های معماری را از نظر تعداد مؤلفه‌های مؤثر در انجام تراکنش نشان می‌دهد.

مقادیر m و n_o، همانند مقادیر m و n_o درمحاسبه رتبه سبک‌های معماری از دید قابلیت اعتماد در نظر گرفته شد. در صورتی که زمان صرف شده در هر مؤلفه از هر سبک معماری را t در نظر بگیریم، مؤلفه‌های سبک شی‌گرا، یعنی کلاس‌ها، نسبت به مؤلفه‌های سایر سبک‌های معماری ریزدانه‌تر می‌باشند؛ بنابراین زمان صرف شده در هر کلاس در مقایسه با زمان صرف شده در هر مؤلفه در سایر سبک‌های معماری به طور معمول کمتر و نسبتی از t است. شکل ۶، نمودار تغییر رتبه قابلیت اعتماد نسبت به تغییر مقدار t در سبک شی‌گرا را برای n_o=۱۵ نشان می‌دهد. مطابق نمودار هر قدر مقدار زمان صرف شده در هر کلاس برحسب t کمتر باشد، رتبه کارایی این سبک افزایش می‌یابد. این مقدار با توجه به تجربه مؤلفان در توسعه سیستم‌های شی‌گرا t/4 در نظر گرفته شد.

شکل ۶. نمودار تغییر رتبه قابلیت اعتماد سبک شی‌گرا برحسب t برای n_o=۵

تولید کنندگان نرم‌افزار براساس تجارب گذشته خود در تولید نرم‌افزار، می‌توانند مقدار این پارامتر را تعیین کنند. با توجه به موارد فوق و با حذف t، تعداد مؤلفه‌های مؤثر در تراکنش در سبک‌های معماری مطابق جدول ۶ می‌باشد. با جایگزین کردن مقادیر m و n_o، در جدول ۶، جدول ۷ ارزش کارایی سبک‌های معماری را برحسب مقادیر مختلف m و n_o نشان می‌دهد.

جدول۶. کارایی سبک‌های معماری برحسب مؤلفه‌های مؤثر در انجام تراکنش

نام سبک	نماد	تعداد مؤلفه‌های مؤثر در انجام تراکنش
مخزن	RPS	2
تخته سیاه	BKB	3
لوله و فیلتر	P/F	m
لایه‌ای	LYD	m
فراخوانی ضمنی	I/I	2
متقاضی/سرویس‌گر	C/S	m/2
کارگزار	BRK	m/2+3
شی گراء	OO	no/4

جدول ۷. ارزش کارایی سبک‌های معماری

نام سبک	نماد	m=۲	m=۳	m=۴	m=۵	m=۶	m=۷
		no =۶	no =۹	no =۱۲	no =۱۵	no =۱۸	no =۲۱
مخزن	RPS	۲	۲	۲	۲	۲	۲
تخته سیاه	BKB	۳	۳	۳	۳	۳	۳
لوله و فیلتر	P/F	۲	۳	۴	۵	۶	۷
لایه‌ای	LYD	۲	۳	۴	۵	۶	۷
فراخوانی ضمنی	I/I	۲	۲	۲	۲	۲	۲
متقاضی/سرویس‌گر	C/S	۱	۱,۵	۲	۲,۵	۳	۳,۵
کارگزار	BRK	۴	۴,۵	۵	۵,۵	۶	۶,۵
شی گرا	OO	۱,۵	۲,۲۵	۳	۳,۷۵	۴,۵	۵,۲۵

رتبه کارایی سبک‌های معماری، از رابطه (۸) محاسبه شده و در جدول ۸ درج شده است.

جدول ۸. رتبه سبک‌های معماری از دید کارایی

نام سبک	نماد	m=۲	m=۳	m=۴	m=۵	m=۶	m=۷
		no =۶	no =۹	no =۱۲	no =۱۵	no =۱۸	no =۲۱
مخزن	RPS	۱۱۸	۱۵۰	۱۷۴	۱۹۴	۲۰۹	۲۲۲
تخته سیاه	BKB	۷۸	۱۰۰	۱۱۶	۱۲۹	۱۴۰	۱۴۸
لوله و فیلتر	P/F	۱۱۸	۱۰۰	۸۷	۷۷	۷۰	۶۴
لایه‌ای	LYD	۱۱۸	۱۰۰	۸۷	۷۷	۷۰	۶۴
فراخوانی ضمنی	I/I	۱۱۸	۱۵۰	۱۷۴	۱۹۴	۲۰۹	۲۲۲
متقاضی/سرویس‌گر	C/S	۲۳۵	۲۰۰	۱۷۴	۱۵۵	۱۴۰	۱۲۷
کارگزار	BRK	۵۹	۶۷	۷۰	۷۰	۷۰	۶۸
شی گراء	OO	۱۵۷	۱۳۳	۱۱۶	۱۰۳	۹۳	۸۵

در این رابطه، x_{ij} تعداد مؤلفه‌های مؤثر در انجام تراکنش سبک در ردیف i-ام و ستون j-ام جدول ۷، و مخرج رابطه مجموع معکوس ارزش کارایی ستون j-ام است. از آنجا که ارتباط سبک‌های معماری با تعداد مؤلفه‌های مؤثر در انجام تراکنش

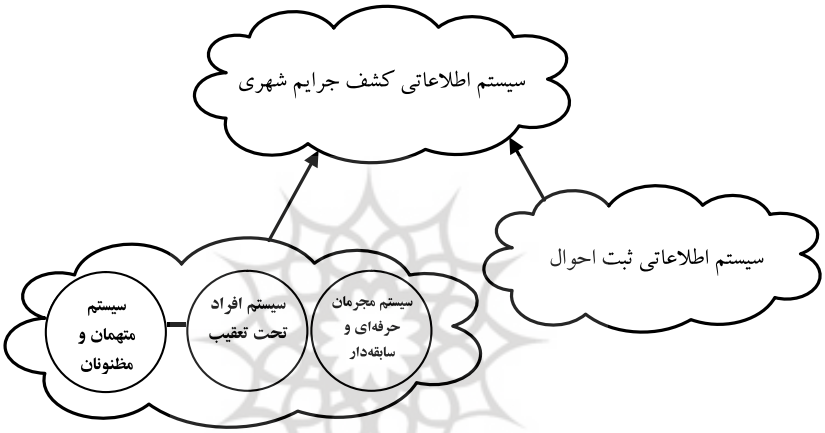
معکوس است (با افزایش تعداد مؤلفه‌های مؤثر در انجام تراکنش، کارایی سبک معماری کاهش می یابد)، از این رابطه استفاده می‌کنیم.

$$R_{ij} = \frac{1/x_{ij}}{\sum_{i=1}^{8} 1/x_{ij}} * 1000 \qquad (۸)$$

۲ - بررسی موردی

قرار است برای پلیس کشف جرائم شهری، یک سیستم اطلاعاتی طراحی و پیاده سازی گردد که امکان ذخیره و بازیابی مشخصات و سوابق جرائم مجرمین را داشته باشد. این سیستم دریچه ای به سیستم های دیگر پلیس است که اطلاعاتی را درباره مجرمان حرفه‌ای و سابقه‌دار، افراد تحت تعقیب و متهمین و مظنونین و ... نگهداری می‌کنند و از این طریق سوابق تخلفات متهم از سایر سیستم‌های پلیس اخذ می‌گردد، و در سیستم به عنوان بخشی از پرونده متهم ذخیره می‌شود. ضمن اینکه سوابق مجرمیت قبلی متهم که از سوی مراجع قضایی صادر شده است نیز از این سیستم استعلام می‌شود. از طرفی بایستی از طریق سیستم امکان استعلام مشخصات هویتی متهم از سیستم ثبت احوال نیز ممکن باشد. در هنگام بازجویی از متهم، استناد بازپرس/بازجو به آگاهی از سوابق مجرمیت و تخلفات او معمولاً در اعتراف به ارتکاب جرم مؤثر است. از طرف دیگر، یکی از مواردی که در شدت مجازات تعیین شده برای متهم از سوی قاضی مؤثر است، وجود سابقه مجرمیت است. بنابراین، پس از تکمیل پرونده متهم و ارجاع او به دادگاه، سوابق مجرمیت او از سیستم استعلام و ضمیمه پرونده می‌گردد. از آنجا که مرجع صدور حکم قاضی است؛ بنابراین، اطلاعات تصمیم نهایی قاضی(حکم) را دادگاه باید در سیستم وارد کند و با توجه به پراکندگی دادگاهها در سراسر کشور، تمام دادگاهها باید امکان ورود اطلاعات حکم را داشته باشند. با توجه به حضور پلیس کشف جرائم در سراسر کشور، امکان اخذ استعلام های ذکر شده در پایگاههای کشف جرائم پلیس در سراسر کشور نیز لازم است. در نتیجه توزیع شدگی، حجم بالای اطلاعات سیستم و افزایش مداوم اطلاعات آن و چند منظوره بودن سیستم، از ویژگی‌های این سیستم است. با توجه به ضرورت رسیدگی سریع به پرونده متهم در دفاتر پلیس و دادگاه و ضرورت امکان

ورود اطلاعات و اخذ استعلام از سیستم و سیستم‌های دیگر حاوی سوابق متخلفین در سراسر کشور، این سیستم باید از قابلیت اعتماد و کارایی یا سرعت پاسخ بالا به استعلام ها برخوردار باشد. از آنجا که همه سیستم ها در طی زمان گسترش می‌یابند، قابلیت نگهداری نیز از ویژگی‌های مهم سیستم می باشد. بنابراین با توجه به مشخصات سیستم، صفات کیفی آن به ترتیب تقدم ۱) قابلیت اعتماد؛ ۲) کارایی و ۳) قابلیت نگهداری می‌باشد. شکل ۷، تعامل این سیستم با سایر سیستم‌های اطلاعاتی پلیس و سیستم ثبت احوال را نشان می‌دهد.



شکل۷. سیستم اطلاعاتی پلیس کشف جرایم شهری و تعامل آن با سایر سیستم های اطلاعاتی

تعیین مشخصه‌های مهم سیستم و پالایش سبک‌های معماری نرم‌افزار

از آنجا که افراد با سیستم کشف جرائم تعامل می‌کنند، سیستم از نوع محاوره ای و غیر واکنشی است؛ زیرا در سیستم های واکنشی، سیستم به طور مداوم در تعامل با محیط است و به واکنش های محیطی پاسخ می‌دهد. از سبک‌های معماری توصیف شده در این تحقیق سبک لوله و فیلتر محاوره ای نیست. ضمن اینکه سبک فراخوانی ضمنی خاص سیستم های واکنشی است. بنابراین، این دو سبک مشخصه‌های لازم برای سیستم کشف جرائم را ندارند. در نتیجه در مرحله دوم سایر سبک‌های معماری مورد رتبه بندی واقع می‌شوند.

اخذ صفات کیفی سیستم

در تعریف سیستم، به صفات کیفی مورد نظر آن اشاره شد کـه قابلیـت اعتمـاد بـالا، کارایی و قابلیت نگهداری بودند.

تعیین اولویت نسبی صفات کیفی

با توجه به اهمیت صفات کیفی سیستم و جدول مقایسه زوجی روش AHP، ماتریس مقایسه زوجی صفات کیفی را مقـداردهی و بـه روش AHP، اولویـت نـسبی صـفات کیفی را محاسبه می کنیم. جدول ۹، ماتریس مقایسه زوجی و اولویت نسبی صـفات کیفی را نشان می دهد.

جدول ۹. ماتریس مقایسه زوجی صفات کیفی سیستم

	قابلیت اعتماد	کارایی	قابلیت نگهداری	اولویت نسبی
قابلیت اعتماد	۱	۱،۵	۲	۰،۴۵۹
کارایی	۱/۱،۵	۱	۱،۶	۰،۳۲۵
قابلیت نگهداری	۱/ ۲	۱/۱،۶	۱	۰،۲۱۶

تعیین اندازه تقریبی سیستم و تعداد مؤلفه‌های مؤثر در انجام تراکنش

با توجه به وظایف سیستم، که نگهداری مشخصات و سوابق جـرایم مجرمـان اسـت، تعداد مؤلفه‌های مؤثر در انجام تراکنش ۳ در نظر گرفته می‌شود. با توجه به: ۱) تعداد موردهای کاربری سیستم (حدود ۱۵مورد کاربری)؛ ۲) تعداد کلاس های موجودیتی سیستم؛ ۳) داشتن وظیفه ذخیره و بازیابی مشخصات و سوابق جرایم مجرمان و عدم انجام پردازش‌های پیچیـده توسـط سیـستم و ۴) حجـم زیـاد اطلاعـات موجـود در سیستم، اندازه سیستم متوسط به پایین در نظـر گرفتـه مـی‌شـود. بنـابراین، تعـداد تقریبی مؤلفه‌های سبک معماری مناسب سیستم، ۴ در نظر گرفته می‌شود.

ارزیابی و رتبه‌بندی نسبی سبک‌های معماری از دید هر یک از صفات کیفی

از آنجا که ارزیابی سبک‌های معماری از دید صفات کیفی در بخش یافته های تحقیق انجام شد، در این مرحله با توجه به تعداد صفات کیفی مورد نظر سیـستم، و تعیـین

تقریبی تعداد مؤلفه‌های سبک معماری(۴) و تعداد مؤلفه‌های مؤثر در انجام تراکنش(۳)، با استفاده از جدول‌های مربوط به رتبه نسبی هر یک از سبک‌های معماری از دید صفات کیفی مختلف، رتبه‌های نسبی هر یک از سبک‌های معماری تعیین که در ادامه تشریح می‌شود.

با توجه به تعداد مؤلفه‌های نرم‌افزار(۴)، رتبه سبک‌های معماری از دید قابلیت نگهداری مطابق مقادیر ستون سوم جدول ۲ است. با توجه به تعداد مؤلفه‌های مؤثر در انجام تراکنش در سیستم (۳)، رتبه سبک‌های معماری از دید قابلیت اعتماد مطابق ستون سوم جدول ۵ و از دید کارایی مطابق ستون سوم جدول ۸ است.

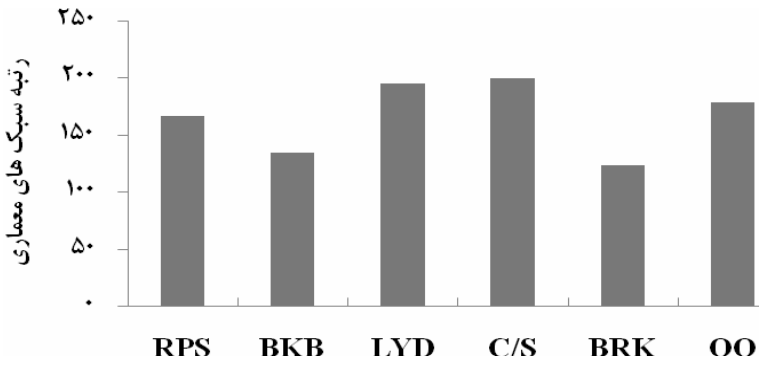
تعیین رتبه نهایی سبک‌های معماری

با توجه به اولویت نسبی صفات کیفی سیستم و رتبه نسبی هریک از سبک‌های معماری از نظر صفات کیفی مختلف، و پس از انجام محاسبات روش AHP با ابزار اکسپرت چویس[1] (www.expertchoice.com، ۲۰۰۹)، رتبه نهایی سبک‌های معماری محاسبه و در جدول ۱۰ درج شده است. برطبق نتایج، سبک متقاضی/سرویس‌گر، بهترین و سبک کارگزار بدترین رتبه را برای سیستم اطلاعاتی فوق کسب کردند. شکل ۸، نمودار رتبه نهایی سبک‌های معماری برای سیستم اطلاعاتی پلیس کشف جرائم شهری را نشان می دهد.

جدول ۱۰. رتبه سبک های معماری برای سیستم اطلاعاتی پلیس کشف جرائم شهری

نماد	RPS	BKB	LYD	C/S	BRK	OO
رتبه	۱۶۷	۱۳۵	۱۹۶	۲۰۰	۱۲۴	۱۷۹

1- Expert Choice



شکل ۸. نمودار رتبه نهایی سبک‌های معماری
برای سیستم اطلاعاتی پلیس کشف جرائم شهری

اعتبارسنجی مدل پیشنهادی

در ارزیابی صفات کیفی قابلیت نگهداری و کارایی از روش ارزیابی مبتنی بر سناریو استفاده شده است که یکی از متداول‌ترین روش‌های ارزیابی معماری نرم‌افزار است. در ارزیابی صفت کیفی قابلیت اعتماد از رویکرد دیاگرام بلوکی قابلیت اعتماد استفاده شده است که یک روش معتبر و شناخته شده در ارزیابی قابلیت اعتماد نرم‌افزارها با پیکربندی های مختلف می باشد. برای ترکیب نتایج سنجش نیز از روش تصمیم گیری چند معیاره AHP استفاده شده است که از روش‌های تصمیم گیری چند معیاره مؤثر است و در اغلب تصمیم‌گیری چند معیاره مورد استفاده قرار می‌گیرد.

نتیجه گیری

در این مقاله، به نقش سیستم های اطلاعاتی در انجام مؤثر مأموریت‌ها و برقراری نظم و امنیت تاکید شد و به مصداق‌هایی از این سیستم‌ها در ناجا به طور اجمالی اشاره شد. از طرف دیگر، ویژگی‌های خاص سیستم‌های اطلاعاتی ناجا مانند: ۱) برخورداری از مقیاس بزرگ؛ ۲) قابلیت نگهداری یا گسترش پذیری به لحاظ تأمین نیازهای جدید و احیانا رفع نواقص موجود؛ ۳) نیاز به برخورداری از قابلیت اعتماد بالا با توجه به ضرورت عملیاتی بودن مداوم؛ ۴) نیاز به سرعت پاسخگویی یا کارایی بالا

به لحاظ تعامل گسترده با مردم و ۵) ضرورت تعامل میان سیستم‌ها با یکدیگر و با سایر سیستم‌های اطلاعاتی کلان جامعه و ۶) بحث یکپارچگی این سیستم‌ها، مطرح شد و لزوم توجه به معماری نرم‌افزار آنها به لحاظ تأثیر اساسی آن در تأمین ویژگی‌های مورد اشاره، تأکید گردید. از آنجا که انتخاب سبک معماری از تصمیم‌های مهم مرحله طراحی محسوب شده و در تأمین صفات کیفی سیستم اطلاعاتی نقش اساسی ایفاء می‌کند، و در حال حاضر انتخاب سبک‌های معماری براساس تجربه معمار انجام می‌شود، روشی برای رتبه‌بندی و توصیه سبک‌های معماری براساس ارزیابی سبک‌های معماری از دید صفات کیفی ارائه گردید.

ابتدا صفات کیفی قابل ارزیابی در سطح سبک‌های معماری، تعیین و براساس ارزیابی کمی سبک‌های معماری از دید این صفات، روشی برای رتبه بندی سبک‌های معماری و توصیه سبک معماری با رتبه بالاتر برای سیستم با توجه به اولویت نسبی صفات کیفی آن سیستم با استفاده روش AHP ارائه گردید. به منظور بررسی مدل پیشنهادی، برای سیستم اطلاعاتی پلیس کشف جرایم سبک‌های معماری رتبه‌بندی شدند.

نوآوری‌های این مقاله، روش‌مند کردن و ارائه مدلی برای انتخاب سبک معماری، براساس تحلیل تأثیر سبک‌های معماری بر روی صفات کیفی نرم‌افزار با استفاده ارزیابی سبک‌های معماری نرم‌افزار و با بهره گیری از روش AHP برای ترکیب نتایج ارزیابی سبک‌های معماری می باشد. این در حالی است که انتخاب سبک معماری در حال حاضر بطور شهودی انجام می‌شود. جزئیات نوآوری‌های مقاله به شرح زیر است:

۱. پالایش صفات کیفی سبک‌های معماری و تعیین صفات کیفی قابل ارزیابی در سطح سبک‌های معماری

۲. ارزیابی مبتنی برسناریو سبک‌های معماری نرم‌افزار از دید قابلیت نگهداری نرم‌افزار و تعمیم آن به قابلیت حمل نرم‌افزار

۳. ارزیابی مبتنی برسناریو سبک‌های معماری نرم‌افزار از دید کارایی

۴. تحلیل اثر اندازه نرم‌افزار(تعداد مؤلفه‌ها) بر رتبه بندی سبک‌های معماری در پشتیبانی از قابلیت نگهداری و قابلیت حمل

۵. تحلیل تعداد مؤلفه‌های مؤثر در انجام تراکنش بر رتبه بندی سبک‌های معماری در پشتیبانی از قابلیت اعتماد و کارایی

۶. ارائه مدلی برای انتخاب سبک معماری نرم‌افزار با استفاده از روش تصمیم گیری چند معیاره AHP

پیشنهادها

با توجه به نقش سیستم‌های اطلاعاتی ناجا در افزایش ارتقاء امنیت عمومی پیشنهاد می‌شود:

- در فرایند طراحی سیستم‌های اطلاعاتی ناجا، به معماری نرم‌افزار و شیوه طراحی معماری و انتخاب سبک معماری توجه اساسی شود.
- با توجه به حجم بالای فعالیت‌های تولید و توسعه نرم‌افزار و لزوم کار گروهی در فناوری اطلاعات و ارتباطات ناجا، با استانداردسازی روش‌های تولید و توسعه نرم‌افزار، امکان هماهنگی و ارتباط میان عوامل تولید و توسعه نرم‌افزار فراهم گردد تا ضمن افزایش کیفیت و بهره وری محصولات نرم‌افزاری، امکان نظارت مناسب بر سیستم‌های تولید شده توسط شرکت‌ها وجود داشته باشد.
- مستندسازی کلیه فرایندهای تولید و توسعه نرم‌افزار متناسب با آخرین استانداردهای مورد استفاده نیز نقش مهمی در افزایش کیفیت و بهره‌وری ایفا می‌کند، ضمن اینکه با ورود افراد جدید به مجموعه امکان انطباق سریع آنان با روش جاری کار از طریق مطالعه مستندات موجود وجود خواهد داشت.
- در این تحقیق میزان پشتیبانی سبک‌های معماری از امنیت نرم‌افزار مورد بررسی قرار نگرفت که در ادامه این تحقیق قابل بررسی می باشد.
- استفاده از ابزار کنترل و مدیریت پروژه و آشناسازی کارکنان مجموعه با این ابزار نیز از موارد ضروری تلقی می‌شود.
- از آنجا که تولید و توسعه بخشی از سیستم های اطلاعاتی ناجا توسط بخش خصوصی انجام می‌شود، مطالعه، انتخاب و بومی‌سازی روش مناسب تخمین هزینه و تلاش تولید سیستم، در برآورد دقیق قیمت و مدت زمان لازم برای تولید سیستم

اهمیت قابل توجهی دارد.

- شناخت سطح سازمان از نظر مدل تکامل یا بلوغ قابلیت‌ها[1] (CMM) و تدوین یک طرح راهبردی برای ارتقاء آن در افزایش کیفیت و بهره وری محصولات نرم‌افزاری فناوری اطلاعات و ارتباطات ناجا نقش اساسی دارد.

منابع:

جلیلی، سعید و شاه محمدی،غلامرضا، " انتخاب سبک معماری نرم‌افزار با استفاده از فرایند تحلیل سلسله مراتبی"، دوازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، دانشگاه شهید بهشتی، اسفند ۸۵.

شاه محمدی، غلامرضا، جلیلی، سعید، " ارزیابی کمی سبک‌های معماری نرم‌افزار از دید قابلیت اعتماد با رویکرد مدل‌سازی دیاگرام بلوکی"، سیزدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، جزیره کیش، ۱۳۸۶.

شاه محمدی، غلامرضا، جلیلی، سعید، " ارزیابی کمی سبک‌های معماری نرم‌افزار مبتنی بر سناریو از دید قابلیت نگهداری "، چهاردهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، دانشگاه صنعتی امیرکبیر، ۱۳۸۷.

بی‌نام (۱۳۸۶). گزارش ویژه به کارگیری فناوری اطلاعات در ناجا، دو ماهنامه تخصصی فناوری اطلاعات و ارتباطات ناجا، شماره اول.

L. Bass, P. Clements, and R. Kazman: "Software Architecture in Practice." Addison-Wesley، 1998.

L. Bass, P. Clements, and R. Kazman: "Software Architecture in Practice (2rd Edition)." Addison-Wesley، 2003.

M. Shaw، and D. Garlan، "software architecture: Perspectives Discipline on an Emerging"، Prentice Hall، 1996.

1- Capability Maturity Model

C. Seo, G. Edwards, S. Malek, N. Medvidovic, " A Framework for Estimating the Impact of a Distributed Software System's Architectural Style on its Energy Consumption", 7st Working IEEE/IFIP Conf. on Software Architecture, pp. 277-280, 2009.

F. Buschmann، et al.، "Pattern-Oriented Software Architecture- A system of Patterns". John Wiley & Sons، 1996.

N.B. Harrison and P. Avgeriou، "Leveraging Architecture Patterns to Satisfy Quality Attributes"، First European Conf. on Software Architecture، Springer، pp. 263-270، 2007.

P. Avgeriou، and U. Zdun، "Architectural patterns revisited:a pattern language"، Proc. of 10st European Conf. on Pattern Languages of Programs، pp.1-39، 2005.

M. Svahnberg, C. Wohlin, "A Comparative Study of Quantitative and Qualitative Views of Software Architectures", Empirical Assessment and Evaluation in Software Engineering, Keele, UK, 2003.

T. L. Saaty and L .G. Vargas، "Models، Methods، Concepts & Applications of the Analytic Hierarchy Process،" Kluwer Academic Publisher، Dordrecht، the Netherlands، 2001.

A. Silberschatz، H.F. Korth، and S. Sudarshan،"Database System Concepts"، 3rd Edition، McGraw-Hill Series in Computer Science ،1999.

ISO، International Organization for Standardization، "ISO 9126-1:2001، Software engineering – Product quality، Part 1: Quality model", 2001.

R. Tripathi، and R. Mall، "Early Stage Software Reliability and Design Assessment"، Proc. of the 12rd Asia-Pacific Software Engineering Conference، pp.619 - 628، 2005.

www.expertchoice.com.

H. Zhu, "Software Design Methodology", Elsevier Butterworth-Heinemann, 2005.

J. Shao ،and BH. Far،"Development of an Intelligent System for Architecture Design and Analysis،" Proc. Of the Canadian Electrical and Computer Engineering Conference، pp.539- 542، 2004.

H. Reza and E. Grant،"Quality-Oriented Software Architecture" ،the IEEE Int. Conf on Information Technology: Coding and Computing، pp. 140 – 145، 2005.

W. Liu and S. Easterbrook, "Eliciting Architectural Decisions from Requirements using a Rule-based Framework," The 2nd Int. Workshop on Software Requirements and Architectures (STRAW '03) at ICSE'03, 2003.
Z. Manna, A. Pnueli, " The Temporal Logic of Reactive and Concurrent Systems", Springer-Verlag, Berlin,Germany, 1992.
S. Moaven, J. Habibi, H. Ahmadi and A. Kamandi, " A Decision Support System for Software Architecture-Style Selection ", 6th International Conference on Software Engineering Research, Management and Applications, 2008.
F. Losavio، et al،" ISO Quality standards for measuring architectures "،
The Journal of System and Software 72، Elsevier، pp. 209-223، 2004.
IEEE-1471. 2000. IEEE recommended practice for architectural descriptions of software-intensive systems. New York: IEEE, 2000.

پژوهشگاه علوم انسانی و مطالعات فرهنگی
پرتال جامع علوم انسانی