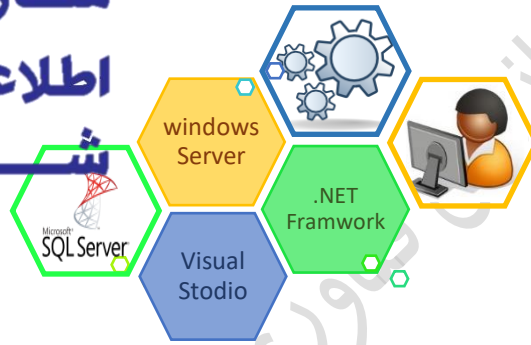




سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



دستورالعمل

الزامات فنی و امنیتی برنامه های نرم افزاری

شهرداری قم

Qom Municipality

Software Quality Attributes Instruction

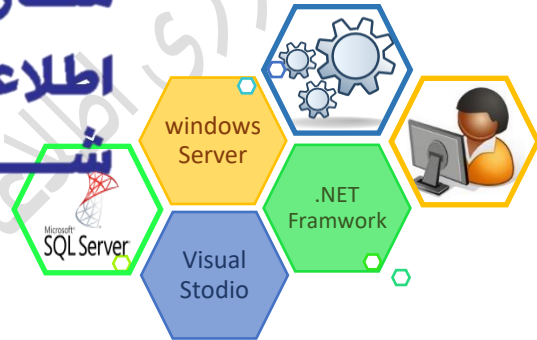
شناسنامه

کد شناسه :	APD.IT.STD.003
تاریخ انتشار :	۱۳۹۷/۰۵/۲۰
شماره ویرایش :	F04
تدوین :	کوروش محمدحسینی
ویرایش :	سید علی حسینیان وفا
بررسی :	حیدر مرتضوی
تایید نهایی :	کوروش محمدحسینی
تصویب :	همایون یزدانیپناه

توجه : این سند توسط سازمان فناوری اطلاعات و ارتباطات شهرداری قم تهیه شده است و هرگونه استناد، استفاده، کپی برداری، و یا بازنشر تمامی یا بخشهای آن بدون ذکر منبع ممنوع می باشد.



سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



فصل اول

اهداف و تعاریف

۱. هدف

این سند به منظور تعیین حداقل الزامات فنی مورد نیاز برای پذیرش یک برنامه نرم افزاری در شهرداری قم تدوین شده است. و براساس آن هر برنامه نرم افزاری (اپلیکیشن، سرویس، سیستم، سامانه، پرتال، وب سایت، و امثالهم) مرتبط با شهرداری قم و اجزاء تابعه آن می باید، واجد این الزامات باشند.

۲. دامنه کاربرد

الزامات ارائه شده در این سند، معیارهای پذیرش برنامه های نرم افزاری در تمامی پروژه هایی است که بصورت کامل یا جزئی، دارای بخش های نرم افزاری می باشند، و از این منظر، دامنه آن شامل هیچگونه محدودیتی از نظر حجم پروژه، نوع نرم افزارها، نوع کاربرد، تعداد کاربران، متدولوژی، چهارچوبه (Framework)، سکو (Platform)، بانک اطلاعاتی و غیره، نمی باشد.

در طول سند، هر زمان نیاز بوده است، استثنائات، و یا خصوصیات ویژه ای را، براساس شرایط خاص برنامه های نرم افزاری و یا ویژگی های آنها، به دقت در نظر گرفته شده است که منحصر به همان موارد بوده و قابل تفسیر و شمول به کلیت سند نبوده و جامعیت دامنه کاربرد آن را محدود نمی سازد.

۳. تعاریف

در نگارش این سند، از اصطلاحات زیر استفاده شده است :

۱,۳. **پروژه** : منظور از پروژه در این سند، هر پروژه فناوری اطلاعات و ارتباطات است که منجر به تولید یا استفاده از یک برنامه نرم افزاری در شهرداری قم و یا یکی از اجزاء تابعه آن بشود.

۲,۳. **برنامه نرم افزاری** : در این مستند هر جا از این واژه استفاده می شود منظور هر سیستم نرم افزاری، اپلیکیشن، سرویس، سیستم، سامانه، پرتال، وب سایت، و امثالهم است که قرار است در شهرداری یا سازمان فاوا و یا تحت نظر شهرداری و یا سازمان فاوا راه اندازی گردد.

۳,۳. **احراز هویت Authentication** : مجموعه عملیاتی است که طی آن بررسی و راستی آزمایی مربوط به نام کاربری و گذواژه وارد شده توسط کاربران، انجام می شود. این عملیات معمولاً در ابتدای ورود کاربر به برنامه انجام شده و نتیجه آن صدور مجوز ورود یا عدم ورود کاربر درخواست کننده به برنامه نرم افزاری می باشد.

۴,۳. **تعیین سطوح دسترسی Authorization** : مجموعه عملیاتی است که طی آن مشخص می شود که چه کاربرانی به چه بخش هایی از برنامه نرم افزاری، چه نوع دسترسی باید داشته باشند و همچنین شامل اعمال این دسترسی ها برای کاربران مورد نظر می باشد.

۵,۳. **Functionality** ^۱ : شامل روش های متداول و خدمات مشترک برای ارتباطات، ذخیره سازی، پردازش و دسترسی به داده ها در پلتفرم برنامه های نرم افزاری و حوزه زیرساخت های ارتباطات است. این موضوع شامل مصادیق زیر است :

۱,۳,۵. **Suitability** ^۲

عملکرد ^۱

مناسب بودن ^۲

۳,۵,۲ Accuracy ۲

۳,۵,۳ Interoperability ۴

۳,۵,۴ Security ۵

۳,۶ Usability^۶: یک خاصیت کیفی است، که تعیین کننده درجه آسانی استفاده از رابط کاربری می باشد. این

موضوع شامل مصادیق زیر است :

۳,۶,۱ Understandability ۷

۳,۶,۲ Learnability ۸

۳,۶,۳ Operability ۹

۳,۶,۴ Attractiveness ۱۰

۳,۷ Reliability^{۱۱}: (یا پایایی)، عبارت است از احتمال کارکرد صحیح و بدون عیب آن سیستم در طول مدت

مشخص و از پیش تعیین شده با شرایط و کیفیت معین. این موضوع شامل مصادیق زیر است :

۳,۷,۱ Maturity ۱۲

۳,۷,۲ Fault tolerance ۱۳

۳,۷,۳ Recoverability ۱۴

۳,۸ Portability^{۱۵}: امکان جابجایی برنامه نرم افزاری از یک محیط به محیط کاری دیگر را قابلیت حمل و نقل

گویند. این محیط می تواند کامپایلر، سیستم عامل و یا سخت افزار باشد. این موضوع شامل مصادیق زیر است :

۳,۸,۱ Adaptability ۱۶

۳,۸,۲ Install ability ۱۷

۳,۸,۳ Conformance ۱۸

۳,۸,۴ Replace ability ۱۹

۳,۹ Maintainability^{۲۰}: نگه داری یک برنامه نرم افزاری، پس از آنکه تحویل داده شد، برای تصحیح خطاها،

بهبود کارایی یا سایر صفات، و نیز تطبیق محصول برای یک محیط جدید. یا میزان تلاشی که لازم است تا اشکالی

که در برنامه نرم افزاری پیدا شده، تصحیح شود و یا خواسته های جدید در آن گنجانده شود. این موضوع شامل

مصادیق زیر است :

۳,۹,۱ Analyzability ۲۱

۳,۹,۲ Changeability ۲۲

۳ دقت

۴ قابلیت همکاری

۵ امنیت

۶ کاربرپذیری

۷ قابل درک بودن

۸ یادگیری

۹ عملیات

۱۰ جذابیت

۱۱ قابلیت اطمینان

۱۲ بلوغ

۱۳ تحمل خطا

۱۴ بازایی

۱۵ قابلیت انتقال

۱۶ انطباق پذیری

۱۷ قابلیت نصب

۱۸ همزیستی

۱۹ جایگزینی

۲۰ نگهداری

۲۱ تجزیه و تحلیل

۲۲ تغییر پذیری

۲۳ Stability .۳,۹,۳

۲۴ Testability.۳,۹,۴

۳,۱۰ **Efficiency**: کارآمدی یا اثربخشی نشان می دهد که یک برنامه نرم افزاری، چقدر در زمینه مورد استفاده، مناسب و اثربخش است. ویژگی کارآمدی، به توانایی آن در برآورده کردن نیازهای کاربر اشاره می کند، همچنین استفاده مطلوب از سخت افزار توسط این ویژگی بیان میشود. این موضوع شامل مصادیق زیر است:

۲۶ Time behavior .۳,۱۰,۱

۲۷ Resource Behavior .۳,۱۰,۲

۳,۱۱ **مستندات**: کلیه مدارک و اسناد کاغذی یا الکترونیکی است، که در جریان اجرای پروژه ایجاد می گردد. مانند: مکاتبات انجام شده بین طرفین و سازمان فاوا، ابلاغیه ها، اختاریه ها، صورتحسابها، صورتجلسات، مستندات فنی، مستندات آموزشی و ...

۳,۱۲ **شهرداری**: منظور از شهرداری در این مستند شهرداری قم یا یکی از اجزاء آن می باشد.

۳,۱۳ **مجری**: شخص حقوقی یا حقیقی که برنامه نرم افزاری یا خدمات نرم افزاری را به شهرداری ارائه می نماید.

۳,۱۴ **سازمان فاوا**: منظور از سازمان فاوا در این مستند، سازمان فناوری اطلاعات و ارتباطات شهرداری قم می باشد که وظیفه مدیریت و نظارت بر عملکرد مجری را برعهده دارد و کلیه مراسلات و مکاتبات و ابلاغیه ها توسط وی انجام شده و کلیه پرداخت ها می باید با تایید وی صورت پذیرد.

۳,۱۵ **کاربر**: هر فرد حقیقی است (اعم از کارکنان و مستخدمین شهرداری و یا سازمان فاوا، شهروندان، مشتریان و ذی نفعان برنامه های نرم افزاری) که بهره بردار و یا استفاده کننده از یک برنامه نرم افزاری می باشد.

۳,۱۶ **فیلدهای پایه (Entity)**: هر فیلد اطلاعاتی که دارای هویتی مستقل در برنامه نرم افزاری می باشد و باید به صورت مستقل در بانک اطلاعاتی نگه داری شود. این موضوع معمولاً به دلیل ایجاد گزارش گیری کارآمد، جلوگیری از ذخیره عناوین اطلاعاتی تکراری ناخواسته و امثالهم انجام می شود.

۳,۱۷ **اطلاعات پایه**^{۲۸}: مجموعه ای از فیلدهای پایه که در بخش مشخصی از بانک اطلاعاتی نگه داری می شوند و برای مقدار دهی به آنها از درگاه واحد و مشخصی استفاده شده و در تمامی برنامه های نرم افزاری از مقادیر قبلا داده شده به آنها به صورت انتخابی استفاده می گردد. مانند: لیست نام دانشگاه ها، لیست اشخاص حقیقی، لیست مدارک تحصیلی و ...

۳,۱۸ **باگ Bug**: وجود یک اشکال در برنامه نرم افزاری (معمولاً در سطح کد نویسی) که باعث عملکردی نادرست در فرآیندها و روال های عملیاتی برنامه های نرم افزاری و یا توقف آنها می گردد.

۳,۱۹ **ارور Error**: وقوع یک مشکل در یک فرآیند و یا روال عملیاتی در برنامه نرم افزاری که می تواند به دلیل وجود یک باگ یا مشکلات ارتباطی فیما بین سرویس دهنده ها و سرویس گیرنده ها، کندی یا کاستی کیفیت در

^{۲۳} پایداری

^{۲۵} کارآمدی

^{۲۷} رفتار استفاده از منابع

^{۲۴} تست پذیری

^{۲۶} رفتار زمان

^{۲۸} Basic Information

بسترهای ارتباطی، انجام یک عمل اشتباه توسط کاربر، روال و یا عملی بر خلاف منطق پذیرش بانک اطلاعاتی، عدم وجود اطلاعات پایه و غیرپایه و ... بوجود بیاید. ارورها ممکن است باعث عملکرد نادرست برنامه شده و یا منجر به توقف تمامی یا بخشی از فعالیت های آنها شوند.

۳،۲۰. **استثنا Exception**: وقوع یک عمل یا اقدام در برنامه نرم افزاری که از قبل، برنامه نرم افزاری برای آن آماده سازی نشده و در مواجهه با آن پیش بینی لازم را ندارد. این نوع مشکل معمولاً با ظاهر شدن یک پیام به کاربر بروز می کند و از ادامه یک فرآیند و یا روال جلوگیری می نماید. شکل دیگری از این موضوع در زمانی رخ می دهد که برنامه قصد دارد داده ای با نوع یا طول یا مقدار نامناسب را در یک فیلد اطلاعاتی در بانک اطلاعاتی ذخیره نماید و بانک اطلاعاتی از وقوع این موضوع جلوگیری می نماید.

۳،۲۱. **ارگونومی نرم افزار Software Ergonomics**: مفهومی است که در رابطه با نحوه طراحی رابط های کاربری و نحوه پیاده سازی روال ها و فرآیندها، به منظور بهبود رابطه حسی و شناختی بهره برداران و کاربران از برنامه های نرم افزاری و همچنین سلامتی جسمی و روانی آنها صحبت می کند. کاربرپسندی^{۲۹}، یکی از اجزاء این مفهوم می باشد.

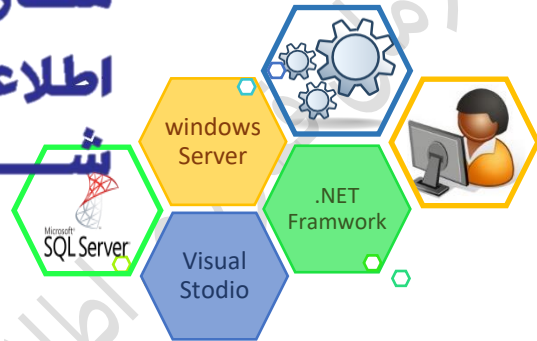
۳،۲۲. **UTF-8**: یا **Universal Character Set Transformation Format 8 bit**، فرمت هشت بیتی انتقال مجموعه کارکترهای جهانی است. این فرمت یک روش رمزگذاری با عرض متغیر (**variable-width**) چند بیتی (**multi-byte**) است که امکان ارسال و دریافت متون **Unicode** بین برنامه های نرم افزاری را فراهم می کند.

۴. منابع و مراجع

- ۴،۱ ISO-9126
- ۴،۲ ISO-12207
- ۴،۳ ISO-25010
- ۴،۴ IEEE-1208
- ۴،۵ Microsoft MSDN
- ۴،۶ OWASP Instruction



سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



فصل دوم

عملکرد

Functionality



۱. معماری

معماری برنامه نرم افزاری، ترجیحا بایستی حداقل دارای سه سطح باشد (Tree Tier) که عبارتند از :

۱.۱. Presentation Layer (UI,SI)

۱.۲. Business Logic Layer (BLL)

۱.۳. Data Access Layer (DAL)

۲. مدیریت ثبت وقایع Log Management

برنامه نرم افزاری، باید قابلیت های لازم برای ثبت و مدیریت وقایع مرتبط با کاربران را داشته باشد. این موضوع چه با استفاده روش های Log Management و چه با سایر روش ها انجام شود، برنامه نرم افزاری می بایست موارد مندرج در «دستور العمل احراز هویت و دسترسی کاربران سیستم های رایانه ای شهرداری قم APD.IT.I.STD.004»، مصوب سازمان فاوا، را دارا باشد.

۳. اعتبار سنجی داده های ورودی Data Validation

۳.۱. واسط کاربری باید با اعمال کنترل های لازم بر روی تمامی فیلدهای ورودی به شکلی اعمال شود که از دریافت داده های نامعتبر، خطرناک و یا اضافی به شرح زیر ممانعت به عمل آورد.

۳.۲. این کنترل ها از ابعاد زیر باید انجام شود :

۳.۲.۱. کنترل تطابق نوعی.

مثال : در فیلد نوع عدد نمی بایست امکان ورود متن وجود داشته باشد (جلوگیری از خطای سمت بانک اطلاعاتی).

۳.۲.۲. کنترل تطابق مقادیر.

مقدار وارد شده در یک فیلد باید با مقادیر مرتبط دیگر در همان فرم تطابق مقدار داشته باشد. مثال : در صورت وجود تاریخ شروع و پایان در یک فرم ، فیلد تاریخ شروع نباید مقادیر بزرگتر از مقدار تاریخ پایان را قبول کند.

۳.۲.۳. کنترل بازه های مجاز.

مقادیر عددی یا تاریخ می بایست در محدوده مجاز پذیرفته شوند، مثال : در سیستم حسابداری، تاریخ اسناد فقط در دوره مالی جاری قابل ورود باشند.

۳.۲.۴. کنترل مقدار طول مجاز.

در فیلدهای با مقدار طول مشخص، می بایست کنترل مقدار طول مجاز یا تعداد کارکتر مجاز انجام شود، مثال : فیلد سال دارای چهار کارکتر است، تعداد کارکتر کمتر یا بیشتر پذیرفته نشود.

۳.۲.۵. کنترل کارکترهای نادرست (تطابق Code Page).

با توجه به تعدد نسخه های سیستم عامل های کاربران و تفاوت های Code Page آنها ، برنامه نرم افزاری، می بایست توسط کنترل ها و توابع خود نسبت به یکسان سازی Code Page ارسالی به سمت بانک اطلاعاتی از نظر یکسان سازی Code Page های متفاوت ورودی با Code Page مورد پذیرش این سند اقدام نماید. این موضوع بخصوص در مورد حروف «ی» و «ک» بسیار مهم است.

۳.۲.۶. کنترل کدهای مخرب و نفوذگر.

کنترل های لازم بر روی تمامی سیستم برای ممانعت از وارد کردن کدهای برنامه نویسی و دستورات سیستمی و SQL ای اعمال گردد. در این موضوع باید به تهدیداتی چون Buffer Overflow و SQL Injection و Cross Site

Scripting و Code Inside Data و سایر اقدامات تخریب گرانه یا نفوذگرانه توجه شود. همچنین محتوای ورودی

ممکن است شامل دستورات Command Line و متغیرهای سیستمی و یا پارامترهای HTTP و کوکی ها باشد.

کنترل های لازم بر روی تمامی سیستم برای ممانعت از ورود دستورات Tag (از قبیل: html و xml) اعمال گردد (به منظور جلوگیری از اقدامات Code Injection و XSS و Script Injection).

در ورودی های با اندازه بالا (مانند: Textarea)، وجود توابعی مانند Eval بررسی شوند زیرا با استفاده از این توابع میتوان یک String را به یک کد اجرایی تبدیل کرد.

۳,۲,۷. مقادیر ساختارمند:

در مورد مقادیری که دارای ساختار و استراکچر مشخص هستند، می بایست از ورود مقادیر بدون رعایت ساختار مجاز جلوگیری گردد، مثال: فیدهای تاریخ که هر سیلاب آن دارای ساختار مشخصی است.

۳,۳. کنترل های موضوع این بخش باید به در خصوص داده های دارای اهمیت زیاد و یا ورودی های با قابلیت حمله های امنیتی به صورت لایه ای انجام شود. به این معنی که در هر یک از سه لایه برنامه نرم افزاری، در زمان تبادل اطلاعات کنترل های مجدد انجام گردد.

۳,۴. در ورودی هایی مانند Textbox و امثال آن مقدار کارکتر ورودی در اندازه مورد نیاز، محدود گردد.

۴. مدیریت سطوح دسترسی

برنامه نرم افزاری، باید قابلیت مدیریت سطوح دسترسی بر اساس «دستور العمل احراز هویت و دسترسی کاربران سیستم های رایانه ای شهرداری قم APD.IT.I.STD.004»، مصوب سازمان فاوا، را داشته باشد.

۵. کارتابل مشترک Single Cartable

در صورت راه اندازی کارتابل مشترک در شهرداری و یا سازمان فاوا، برنامه نرم افزاری، باید قابلیت کارکردن با آن را از طریق وب سرویس داشته باشد و تمامی فرآیندهای مرتبط را، مطابق با دستورالعمل و الزامات مربوطه پشتیبانی نماید.

۶. الزامات امنیتی

برنامه نرم افزاری باید حداقل موارد امنیتی زیر را دارا باشد:

۶,۱. مجری با پذیرش همکاری با شهرداری قم و اجزاء آن، تضمین می نماید که برنامه نرم افزاری ارائه شده فاقد هر گونه کد خرابکارانه و جاسوسی می باشد.

۶,۲. اطلاعاتی که، در کوکی ها و یا فایل های محلی نگه داری می شوند می بایست فقط به صورت رمزنگاری شده باشند. همچنین در هنگام بسته شدن برنامه نرم افزاری، کوکی های غیرضروری حذف شوند.

۶,۳. برنامه نرم افزاری ترجیحا می بایست در صورتیکه به میزان مشخصی بدون استفاده بماند به صورت خودکار از برنامه خارج شده و Session مربوطه بسته شود. در این صورت مقدار زمان و تنظیمات مربوط به این بخش می بایست در قسمت مدیریت، قابل تنظیم باشد.

۶,۴. فایل هایی که، فراخوانی و باز می شوند در صورتیکه فقط نیاز به خواندن یا مشاهده آنها می باشد به صورت Read-only باز شوند.

۶,۵. در صورت استفاده از فایل های موقت، نام گذاری آنها یونیک و غیرقابل حدس زدن باشند.

۶,۶. تمامی اتصال هایی که با پایگاه داده برقرار می گردند بلافاصله پس از انجام تراکنش مورد نیاز باید بسته شوند.

۶,۷. نباید هیچ رکورد اطلاعاتی قابلیت حذف فیزیکی از بانک های اطلاعاتی را داشته باشند، تمامی حذف های رکوردها انحصارا می بایست به صورت منطقی بوده و رکورد مورد نظر به صورت غیرفعال درآید و در بانک های اطلاعاتی حفظ شود.

۶,۸. برای اتصال سرویس های نرم افزار به بانک های اطلاعاتی نبایست از دسترسی های ادمین و یا دسترسی های دارای سطح بالا (مانند SA) استفاده شود. برای این منظور یک Role با حداقل دسترسی مورد نیاز تعریف و استفاده گردد.

۶,۹. برنامه نرم افزاری برای عملکرد در نسخه سرور و همچنین کلاینت (نسخه های ویندوز دسکتاپ و وب) بایستی از حداقل دسترسی (ترجیحا Domain User) استفاده نمایند. در صورت نیاز به استفاده از هرگونه دسترسی بیشتر از Domain User، از دسترسی هایی که کاملا به صورت دقیق و فقط با افزودن سطوح دسترسی حداقلی مورد نیاز، با هماهنگی سازمان فاوا استفاده شود. به هر صورت استفاده از سطوح دسترسی بالا مانند Domain Admin و یا Local Admin به هیچ شکلی در سمت سرور و یا کلاینت مجاز نمی باشد.

۶,۱۰. ترجیحا می باید این امکان باید وجود داشته باشد که برای فرایندهای مهم (مانند حذف یک رکورد) بتوان روال تاییدیه از کاربر را فعال کرد.

۶,۱۱. ترجیحا جهت کاهش تاثیرات حملات DoS مصرف منابع توسط سرویس ها محدود شوند و همچنین این درخواست ها به بعد از ورود کاربر به سیستم (Login) منتقل شوند.

۶,۱۲. تلاش شود که از سرویس ASP.NET STATE استفاده نشود و این سرویس نیز غیرفعال شود. در غیر اینصورت و الزام برای استفاده از آن حتما با سازمان فاوا در خصوص اتخاذ روش های مناسب امن سازی هماهنگ شود.

۶,۱۳. در مورد Web Application ها هرچند که امکان File browsing روی سرور غیرفعال گردیده است، با این حال ترجیحا می باید، در تمام فولدرها حتی آنهایی که اسکریپت اجرایی برای IIS ندارند، یک فایل default.htm یا default.aspx با Redirect روی صفحه ۴۰۴ پرتال اصلی قرار داده شود.

۶,۱۴. در خصوص برنامه های نرم افزاری تحت وب که در بستر اینترنت مورد استفاده قرار میگیرند، ترجیحا می باید جهت جلوگیری از دسترسی ربات های نرم افزاری (Search Engines) به ساختار و فولدربندی، از فایل Robot.txt برای محدود کردن این دسترسی ها استفاده شود.

۶,۱۵. مجری موظف است برای برنامه های نرم افزاری که در شهرداری قم نصب و راه اندازی یا ارایه می نماید، گواهینامه رسمی امنیتی از آزمایشگاه های مورد تایید سازمان فاوا اخذ نماید. آزمون ها و گواهی های اخذ شده انحصارا می بایست متعلق به نسخه های در حال کار در مجموعه شهرداری قم و اجزاء آن و همچنین دارای تاریخ اعتبار جاری باشد. گواهینامه امنیتی برای آزمون های زیر می بایست ارایه شود:

Vulnerability Scanning	۶,۱۵,۱
Security Scanning	۶,۱۵,۲
Penetration Testing	۶,۱۵,۳
Risk Assessment	۶,۱۵,۴
Security Auditing (Code, Application and Operating System)	۶,۱۵,۵
Posture Assessment	۶,۱۵,۶
Confidentiality and Authentication and Authorization	۶,۱۵,۷

۶،۱۶. در صورتیکه سازمان فاوا راسا یا از طریق سایر اشخاص حقیقی و یا حقوقی اقدام به بررسی امنیتی برنامه نرم افزاری ننماید و در صورت مشخص شدن مشکلات و یا حفره های امنیتی، مسئولیت رفع عاجل و رایگان آنها با مجری می باشد. همچنین مجری موظف است همکاری کامل و کافی و رایگان با شهرداری و سازمان فاوا، در راستای انجام بررسی های امنیتی را داشته باشد.

۶،۱۷. نمونه هایی از حملات امنیتی^{۳۰} که سیستم های نرم افزاری می بایست در مقابل آنها نفوذ ناپذیر باشند به شرح زیر می باشد:

- Abuse of Functionality .۶،۱۷،۱
- Brute Force .۶،۱۷،۲
- Buffer Overflow .۶،۱۷،۳
- Content Spoofing .۶،۱۷،۴
- Credential/Session Prediction .۶،۱۷،۵
- Cross-Site Scripting (XSS) .۶،۱۷،۶
- Cross-Site Request Forgery (CSRF) .۶،۱۷،۷
- Denial of Service (DoS) .۶،۱۷،۸
- Fingerprinting .۶،۱۷،۹
- Format String .۶،۱۷،۱۰
- HTTP Request Splitting .۶،۱۷،۱۱
- HTTP Response Splitting .۶،۱۷،۱۲
- HTTP Request Smuggling .۶،۱۷،۱۳
- HTTP Response Smuggling .۶،۱۷،۱۴
- LDAP Injection .۶،۱۷،۱۵
- Null Byte Injection .۶،۱۷،۱۶
- Path Traversal .۶،۱۷،۱۷
- Predictable Resource Location .۶،۱۷،۱۸
- Remote File Include (RFI) .۶،۱۷،۱۹
- Session Fixation .۶،۱۷،۲۰
- SSI Injection .۶،۱۷،۲۱
- SQL Injection .۶،۱۷،۲۲
- URL redirectors Abuse .۶،۱۷،۲۳
- XML Attribute Blowup .۶،۱۷،۲۴
- XML External Entities .۶،۱۷،۲۵
- XML Entity Expansion .۶،۱۷،۲۶
- XML Injection .۶،۱۷،۲۷

۶،۱۸. در بخش هایی که امکان بارگزاری و آپلود فایل وجود دارد، می بایست فقط امکان بارگزاری فایل با نوع مورد استفاده در سیستم وجود داشته باشد. بطور مثال: در قسمت آرشیو تصاویر فقط امکان بارگزاری فایل های تصویری وجود داشته باشد.

۶،۱۹. از بارگزاری فایل هایی که قابلیت تخریب و یا نفوذ دارند، بخصوص فایل با انواع زیر جلوگیری شود: **exe, com, bat, asp, jsp, reg, php, scr, wsf, wsc, shs, vbe, htm, html, zip, rar**

۶،۲۰. امکان تعیین و تغییر نوع فایل های غیرقابل بارگزاری، در پنل های مدیریتی می بایست وجود داشته باشد.

³⁰ استخراج شده از WASC10

۶.۲۱. امکان تعیین و تغییر حداکثر حجم فایل قابل بارگزاری، می بایست در پنل های مدیریتی وجود داشت باشد (جهت جلوگیری از اقدامات مخرب بطور مثال حمله DoS و بهینه سازی ذخیره سازی).

۶.۲۲. اطلاعات مهم و یا محرمانه بر روی فایل های پیکربندی، ارتباطی و یا سیستمی (موجود در سرور و یا احتمالا کلاینت) مانند فایل web.config و فایل های حاوی connection string، همچنین اطلاعات محرمانه موجود در سیستم نرم افزاری (در صورت وجود و با تایید سازمان فاوا)، می بایست به صورت رمزنگاری شده ذخیره شوند. در چنین حالتی کلیدهای رمزگشایی می بایست در اختیار سازمان فاوا قراردادده شود.

۶.۲۳. برنامه نرم افزاری، نباید برای اجرا و یا عملکرد هیچ بخشی از خود نیاز به ارتباط با سرویس های بیرون از شبکه شهرداری قم (بخصوص سرویس های Cloud بیرون شبکه) منجمله سرویس های Google داشته باشد. چنین ارتباطاتی فقط در صورت موافقت سازمان فاوا و رعایت استانداردها و روش های ابلاغی قابل انجام خواهد بود

۶.۲۴. ترجیحا می باید برنامه نرم افزاری تحت وب به صورت HTTPS اجرا شوند.

۶.۲۵. بروی سرورهای مربوط به برنامه نرم افزاری (اعم از سرویس دهنده های بانک های اطلاعاتی و سرورهای اجرایی و عملیاتی) هیچگونه برنامه نرم افزاری اضافه (غیر از سرویس ها و اجزاء اصلی مرتبط با برنامه نرم افزاری) نباید نصب شود (منجمله برنامه های عمومی مانند Office، مدیریت فایل های فشرده، برنامه های گرافیکی و ...).

۶.۲۶. بروی سرورهای فوق الذکر حداقل درایورهای سخت افزاری می باید نصب شود.

۶.۲۷. ارتباط اینترنتی به هیچ عنوان نباید روی سرورهای فوق الذکر وجود داشته باشد (بجز در مورد سرویس هایی که به منظور ارائه سرویس و خدمات روی بستر اینترنت و با هماهنگی و نظارت و براساس دستورالعمل های سازمان فاوا راه اندازی می شوند).

۶.۲۸. بروی سرورهای فوق الذکر فقط افراد واجد صلاحیت می باید اجازه Login داشته باشند و تمامی دسترسی های غیرضروری شبکه ای، مستقیم و غیرمستقیم به این سرورها و سرویس های روی آنها می باید حذف شوند. لیست افراد واجد صلاحیت برای هر سرور توسط سازمان فاوا مشخص می شود.

۶.۲۹. امکان دسترسی به سرویس های بانک های اطلاعاتی (منجمله بانک های اطلاعاتی SQL Server) فقط می باید در اختیار افراد واجد صلاحیت قراردادده شود و تمامی دسترسی های غیرضروری شبکه ای، مستقیم و غیرمستقیم به بانک های اطلاعاتی و سرویس های آنها می باید حذف شوند. لیست افراد واجد صلاحیت برای هر بانک اطلاعاتی و سرویس های آنها توسط سازمان فاوا مشخص خواهد شد.

۶.۳۰. هیچ یک از بانک های اطلاعاتی و فایل های حاوی اطلاعات، در هیچ شرایطی نمی بایست بروی درایو سیستم عامل سرورها (درایو C یا مثالهم) قرار داده شوند. تمامی اطلاعاتی اعم از ثابت یا متغیر منحصرآ می باید بر روی درایوهای غیر سیستم عاملی و با هماهنگی سازمان فاوا قرار داده شوند.

۷. بانک های اطلاعاتی

۷.۱. بانک اطلاعاتی قابل قبول برای ذخیره سازی اطلاعات توصیفی سیستم های نرم افزار SQL Server 2016 R2 به بعد و Oracle 11g Release2 به بعد می باشد.

۷.۲. بانک اطلاعاتی قابل قبول برای ذخیره سازی اطلاعات مکانی سیستم های نرم افزار SQL Server 2016 R2 به بعد و یا ترجیحا PostgreSQL می باشد.

۷,۳. در خصوص برنامه های نرم افزاری که اطلاعات را به صورت سیستم های فایل ذخیره می نمایند میبایست از روش های استاندارد File Streaming و براساس استانداردها و روش های امن و بهینه شده آن، استفاده شود.

۸. سکو (Platform)

۸,۱. پلتفورم قابل قبول سمت سرویس دهنده ها :

۸,۱,۱. Windows Server 2012 R2 و به بعد.

۸,۱,۲. Linux نسخه های : Suse Enterprise . Redhat Enterprise . CentOS . Ubuntu .

۸,۲. پلتفورم قابل قبول سمت سرویس گیرنده ها، در برنامه های نرم افزاری ویندوز دسکتاپ، MS Windows 7,8,10 تمامی Edition ها می باشد.

۸,۳. پلتفورم قابل قبول سمت سرویس گیرنده ها، در برنامه های نرم افزاری تحت وب، محدودیت خاصی ندارد و می بایست در تمامی Browser های استاندارد قابلیت اجرای کامل داشته باشند.

۸,۴. پلتفورم قابل قبول سمت سرویس گیرنده ها، در موبایل اپلیکیشن ها، سیستم عامل Android نسخه چهار به بالا و سیستم عامل iOS دو سال گذشته می باشد.

۹. زبان برنامه نویسی

۹,۱. تمامی زبان های برنامه نویسی که سیستم های نرم افزاری تولید شده با آنها قابل راه اندازی بر روی پلتفورم های پذیرفته شده در این مستند هستند، مورد پذیرش می باشند.

۹,۲. زبان های برنامه نویسی، که به صورت ذاتی متعلق به سایر پلتفورم ها بوده و توسط نرم افزارهای واسط و ایمولیتورها بر روی پلتفورم های پذیرفته شده، راه اندازی می گردند قابل پذیرش نمی باشند.

۱۰. سرورهای مجازی

برنامه نرم افزاری، باید قابلیت نصب و عملکرد کامل، بر روی سرورهای مجازی (Virtualization) مطابق با استانداردها و رویه های مرسوم در شهرداری قم را داشته باشد.

۱۱. استوریج

بانک های اطلاعاتی برنامه نرم افزاری، باید قابلیت نصب و بهره برداری کامل در استوریج های موجود در دیتاسنتر سازمان فاوا را، با حداکثر کیفیت و کارآیی داشته باشد.

۱۲. دیتاسنتر

تمامی برنامه های نرم افزاری می بایست به صورت کامل، در دیتاسنتر شهرداری قم نصب و راه اندازی گردند.

۱۳. Cloud

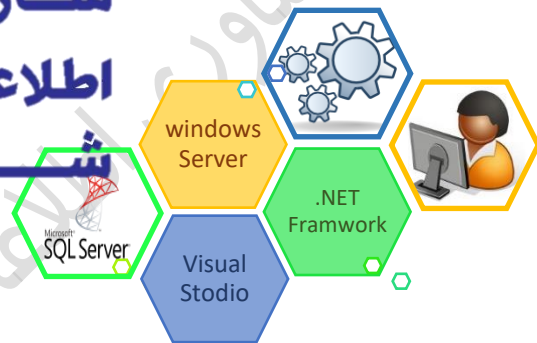
ذخیره سازی هیچگونه داده و اطلاعات برنامه های نرم افزاری، بر روی فضای اینترنت و Cloud قابل قبول نمی باشد و تمامی اطلاعات و داده ها می بایست در فضای استوریج های دیتاسنتر شهرداری قم، ذخیره سازی شوند.

۱۴. سیستم مختصات و تصاویر جغرافیایی

سیستم مختصات مورد پذیرش، برای اطلاعات جغرافیایی، ذخیره سازی و پردازش آنها و مازول های مرتبط با آن صرفاً سیستم مختصات جهانی WGS-84 می باشد سیستم تصاویر مورد پذیرش، UTM می باشد.



سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



فصل سوم

کاربردپذیری

Usability

۱. مدیریت اطلاعات پایه

برنامه نرم افزاری در موضوع مدیریت اطلاعات پایه باید دارای توانایی های زیر باشد :

- ۱.۱. باید (در صورت نیاز) قابلیت ایجاد، مقدار دهی و نگهداشت کلیه اطلاعات و داده های پایه را دارا باشد.
- ۱.۲. باید دارای روالهای ارزیابی برای مقدار دهی و ویرایش اطلاعات پایه به شکلی باشد که از ایجاد مقادیر تکراری، ناخواسته، و یا مقادیر اشتباه جلوگیری نماید.
- ۱.۳. باید به شکلی طراحی و اجرا گردد که پس از ایجاد و مقداردهی اطلاعات پایه، در تمامی بخش ها، از طریق ابزارهای مناسب (مانند لیست های بازشدنی) بتوان مقدار مورد نظر را انتخاب کرد.
- ۱.۴. در پنل های مربوط به ایجاد، مقدار دهی و ویرایش یا فراخوانی و انتخاب اطلاعات پایه ، باید روشهای مناسب برای جستجو و یافتن مقادیر مورد نظر به سهولت فراهم شده باشد.
- ۱.۵. در صورت ابلاغ سازمان فاوا مبنی بر وجود مقادیر پایه ای سازمانی (به صورت وب سرویس، دیتابیس و یا سایر روش های استاندارد)، برنامه نرم افزاری باید قابلیت فراخوانی و استفاده از آنها را به صورت On-line بدون نیاز به ذخیره سازی مجدد در بانک اطلاعاتی را داشته باشد.
- ۱.۶. در صورت ابلاغ سازمان فاوا مبنی بر وجود مقادیر پایه ای سازمانی (به صورت وب سرویس، دیتابیس و یا سایر روش های استاندارد)، برنامه نرم افزاری باید قابلیت فراخوانی و استفاده از آنها را به صورت Off-line ، به شکل ذخیره سازی در بانک اطلاعاتی خود و بروزرسانی (Sync) آنها در بازده های زمانی قابل انتخاب داشته باشد (تشخیص نوع ارتباط On-line یا Off-line در مورد هر یک از انواع اطلاعات پایه با سازمان فاوا خواهد بود. در صورت تامین زیرساخت لازم توسط شهرداری مانند ESB و امثالهم ، این بند قابل حذف خواهد بود).

۲. ارگونومی Software Ergonomics

- ۲.۱. بخش های مختلف رابط کاربری، باید از یک رویه و الگوی یکسان برخوردار باشند.
- ۲.۲. به منظور ایجاد حداکثر سهولت در طراحی و پیاده سازی رابط کاربری از رویه های پذیرفته شده و مرسوم در سایر سیستم های نرم افزاری بهره گیری شود.
- ۲.۳. در برنامه های تحت وب صفحات نمایش داده شده (Form ها) به کاربر باید از اسکرول افقی پرهیز شود.
- ۲.۴. در برنامه های ویندوز دسکتاپ و موبایل اپلیکیشن از اسکرول افقی و عمودی پرهیز شود.
- ۲.۵. اسکرول افقی در مورد جداول داخل فرم ها بلامانع است.
- ۲.۶. هنگامی که عملیات درخواستی کاربر مدتی به طول می انجامد با روشهای مناسب (مانند : نمایش پنجره انتظار، جملات کوتاه، نشانک های گرافیکی، ساعت شنی ، Progress Bar) لزوم انتظار کاربر اطلاع رسانی شود.
- ۲.۷. کنترل های روی فرم ها باید دارای Tab Order مناسب باشد.
- ۲.۸. جهت افزایش سرعت کاربری، روال های اصلی برنامه می بایست هم دارای دکمه (Button) های مشخص باشند و هم توسط ترکیبات کلیدی میانبر (Shortcut) قابل انتخاب باشند.
- ۲.۹. کلیدها و گزینه ها می بایست با توجه به وظیفه ای که به عهده دارند دارای Caption (عنوان) معنادار باشند.
- ۲.۱۰. برای اجرای فرآیندها یا ارایه گزینه ها به کاربر، از کلیک راست ماوس استفاده نشود.

۳. ماژول گزارش و آمار

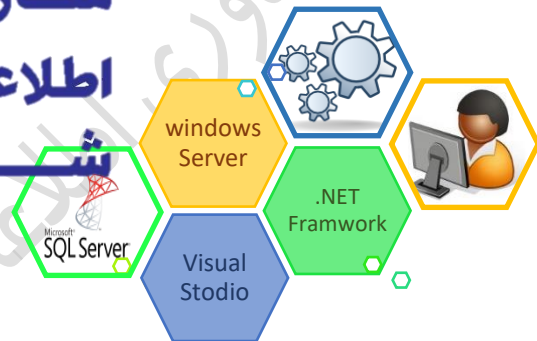
برنامه نرم افزاری باید دارای ماژول گزارشات و آمار مطابق با مشخصات زیر باشد :

- ۳.۱. امکان فراخوانی و مشاهده گزارشات مورد نیاز در اشکال و قالبهای گرافیکی و نموداری وجود داشته باشد.
- ۳.۲. اگر داده های اطلاعاتی برنامه نرم افزاری، واجد مختصات جغرافیایی باشند، لازم است گزارشات سیستم بر روی نقشه به صورت گزارشات مکانی هم قابل مشاهده باشند.
- ۳.۳. نتیجه هر گزارش را باید بتوان با فرمت های استاندارد (Excel ، PDF ، تصاویر و ...) خروجی گرفت.

سازمان فناوری اطلاعات و ارتباطات شهرداری قم



سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



فصل چهارم

قابلیت اطمینان

Reliability



۱. مدیریت خطاها

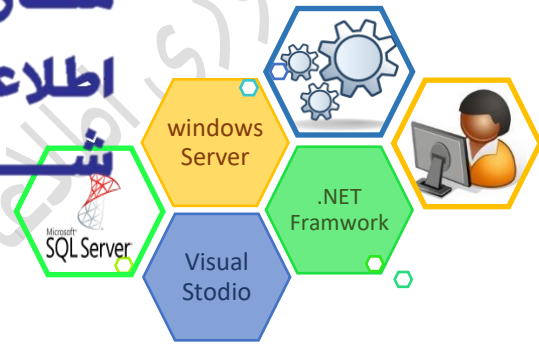
- ۱.۱. برنامه نرم افزاری باید قابلیت مدیریت خطاها را به شرح زیر داشته باشد :
- ۱.۲. پیام های خطا باید مدیریت شده و دقیق باشد و کاربران را از اشکالات مطلع نماید.
- ۱.۳. در زمان نمایش پیام های خطا و یا هر پیام دیگری و یا در زمان واکنش سیستم به رخداد های خطا، نباید هیچگونه اطلاعات سیستمی از برنامه نرم افزاری یا مشخصات و اطلاعات سرویس ها و سرویس دهنده ها نمایش داده شود. همچنین بطور کلی هیچ پیام سیستمی اکیدا نباید نمایش داده شود و تمامی پیام ها باید مدیریت شده و با محتوای تغییر داده شده و مورد نظر نمایش داده شوند.
- ۱.۴. تمامی خطاها باید مدیریت شده باشند و هیچ پیام خطای پیش فرضی به کاربر نمایش داده نشود.
- ۱.۵. برنامه نرم افزاری، باید حداقل خطا را داشته باشد و تلاش شود که کاربران در هنگام استفاده از برنامه با هیچ پیام خطایی روبرو نشوند. خطاها بلافاصله پس از تشخیص باید به صورت ریشه ای رفع شوند.
- ۱.۶. برنامه نرم افزاری باید توانایی مقابله با خطاها را داشته باشد، به این معنا که در صورت وقوع یک خطا (حتی خطاهای از قبل پیش بینی نشده) باید حداقل موارد زیر را بتواند به انجام رساند :
 - ۱.۶.۱. به شکل قابل قبولی با نمایش پیام مناسب از برنامه خارج شود (Error Handling).
 - ۱.۶.۲. از تخریب اطلاعاتی که در بانک اطلاعاتی ذخیره شده اند جلوگیری نماید (Data Corruption Protection)
 - ۱.۶.۳. از گم شدن یا از دسترس خارج شدن فرآیند (های)ی که در حال انجام بوده (اند) جلوگیری نماید (Data Lost Protection).
 - ۱.۶.۴. فرآیند یا اطلاعات جاری در آن لحظه را به آخرین مرحله پایدار خود برگرداند (Rollback).
 - ۱.۶.۵. در شرایط وقوع خطا کنترل های خود بر روی امنیت سیستم و جلوگیری از نفوذ، و هک و طبقه بندی اطلاعات و امثالهم را از دست ندهد. بطور کلی پس از وقوع یک خطا، یا برنامه بدون وقفه و نقصی در سایر قسمت ها به کار خود ادامه دهد و یا برنامه نرم افزاری بسته شده و کاربر به بیرون از برنامه هدایت شود.
- ۱.۷. در صورتیکه برنامه نرم افزاری در سمت بهره بردار دچار مشکلی شد که منجر به خروج از برنامه گردد ، لازم است طی روشی مناسب به مسئول معرفی شده توسط سازمان فاوا به صورت خودکار اطلاع رسانی شده و طی پیام مناسبی استفاده کننده از برنامه خارج شده و تمامی حافظه و کوکی های مرتبط پاک شوند.
- ۱.۸. برنامه نرم افزاری باید ظرفیت تحمل خطای (Fault tolerance) قابل قبولی داشته باشد، به این معنا که در اثر وقوع مشکلات روتین و یا قابل پیش بینی، دچار مشکل نشود و از قبل برای وقوع آنها آماده سازی شده باشد.

۲. ثبت مشکلات

- ۲.۱. ترجیحاً می بایست برنامه نرم افزاری باید دارای بخش Event Log داخلی باشد که وقایعی مانند موارد زیر را با حداکثر جزئیات قابل دستیابی، در آن ثبت نماید :
- ۲.۲. قطعی ارتباط بین سرویس گیرنده، سرویس دهنده ها ، و بانک های اطلاعاتی.
- ۲.۳. توقف و یا ریست شدن سرویس برنامه.
- ۲.۴. قطعی ارتباط با هر یک از وب سرویس های ورودی (و خروجی).
- ۲.۵. وقوع هر گونه خطا، در هر یک از اجزاء سیستم.



سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



فصل پنجم

قابلیت انتقال

Portability

برنامه های نرم افزاری به تفکیک پلات فورم باید واجد خصوصیات زیر باشند :

۱. تحت وب

۱.۱. طراحی برنامه نرم افزاری، بایستی به گونه ای انجام شود که نیاز به نصب هیچگونه نرم افزار اضافی در سمت سرویس گیرنده ها وجود نداشته باشد (بطور مثال : Macromedia flash، Java applet و Silverlight و یا سایر Runtime Application ها مانند Active X).

۱.۲. در موارد بسیار ویژه و خاص (به طور مثال نیاز به استفاده از سخت افزارهای خاصی مثل اسکنرها و نیاز به نصب OCX های ویژه ای برای آنها) و امکان نصب موارد مورد نیاز از طریق نصب درون برنامه ای مقدور نمی شود، تمامی مواردی که لازم است نصب شوند می بایست در یک پکیج متمرکز شده (فایل Setup) و در اختیار سازمان فاوا برای نصب قرار گیرد. ترجیحا نصب موارد مورد در چنین شرایطی به صورت خودکار صورت گرفته و یا لینک نصب در اختیار استفاده کننده قرار داده شود.

۱.۳. طراحی برنامه نرم افزاری باید به صورت Responsive باشد به شکلی که با رزولیشن های مختلف صفحه نمایش به سادگی سازگار شده و بدون مشکل عمل نماید.

۱.۴. امکانات و قابلیت های برنامه های نرم افزاری بایستی به طور کامل از طریق مرورگرهای کامپیوترهای شخصی قابل دسترسی و استفاده بوده و در مرورگرهای موبایل و تبلت، حداکثر قابلیت ممکن جهت استفاده آسان در نمای اختصاصی این دستگاه ها را دارا باشد.

۱.۵. برنامه های نرم افزاری تحت وب باید حداقل بر روی ویرایش های حال مربوط به دو سال گذشته مرورگرهای زیر بدون اشکال و با سهولت و بدون تغییر در تنظیمات و پیکربندی مرورگر (صرفا با تنظیمات پیشفرض مرورگرها) و ساختار و عملکرد برنامه اجرا و عمل نماید (منظور از دو سال گذشته، یعنی اینکه در هر زمان که سیستم مورد استفاده می باشد تا دو سال گذشته آن و از این منظر تغییرات مورد نیاز برای بروزرسانی های جدید مرورگرها در هر زمان که نیاز باشد بر عهده مجری است) :

MS IE .۱،۵،۱

Google Chrome .۱،۵،۲

Mozilla Firefox .۱،۵،۳

۲. ویندوز دسکتاپ

۲.۱. برنامه نرم افزاری باید دارای روال بروزرسانی خودکار نسخه های کلاینتی خود باشد. به این معنا که سرویس متمرکزی وجود داشته باشد، که براساس یک روال مشخص و قابل قبول نسخه های کلاینتی را به صورت خودکار بروزرسانی نماید. این قابلیت به شکلی باید عمل نماید که هیچ نسخه ای از برنامه کلاینتی با ویرایش قدیمی تر از آخرین ویرایش نباید بتواند وارد برنامه شود. هر نسخه از برنامه کلاینتی در آغاز اجرا می بایست شماره ویرایش خود را با سرویس مربوطه کنترل کرده و در صورت مغایرت از ادامه ورود به برنامه جلوگیری و با نمایش پیام مناسبی به کاربر؛ عملیات به بروزرسانی خودکار انجام شود.

۲.۲. برنامه نرم افزاری باید قابلیت اجرای بدون نقص در نسخه های مختلف Windows (7,8,10) تمامی Edition ها را داشته باشد.

۲.۳. ترجیحا برای اجرای برنامه نرم افزاری، نیاز به پیش نصب نرم افزارهای جانبی وجود نداشته باشد.

۲.۴. نسخه کلاینت برنامه نرم افزاری باید دارای فایل نصبی (Setup.exe) بوده و تمامی نرم افزارهای پیش نیاز و مورد نیاز (فقط در صورت ضرورت) و تنظیمات سیستمی می بایست توسط آن صورت گیرد.

۲.۵. برنامه نرم افزاری باید دارای یک روال آسان برای برقراری ارتباط با سرور(ها) باشد، بطور مثال : فایل همراه (به صورت رمزنگاری شده). که در چنین حالتی پیمانکار می بایست ابزارهای لازم برای ایجاد و تغییر فایل های همراه و یا تنظیم روش ارتباطی، به منظور تنظیم مجدد ارتباط با سرورها را در اختیار سازمان فاوا قرار دهد.

۲.۶. برنامه های نرم افزاری تحت ویندوز نباید برای اجرا و عملکرد خود، هیچگونه وابستگی به تنظیمات محیطی سیستم عامل، سخت افزار، کاربر نصب کننده، کاربری که به سیستم عامل لاگین نموده است و سطح دسترسی وی، و سایر مشخصات سیستم داشته باشد.

۲.۷. برنامه های نرم افزاری ویندوز دسکتاپ نباید برای اجرای و عملکرد خود، هیچگونه نیازی به تغییر در تنظیمات سیستم عامل، مانند : **Date & Time, Currency, Region** و غیره داشته باشند و باید بتوانند روی تنظیمات پیش فرض سیستم عامل نصب و عمل نمایند.

۲.۸. تمامی فایل های برنامه نرم افزاری می باید انحصاراً در یک شاخه قرار گیرند و قراردادن فایل های مختلف مربوط به برنامه نرم افزاری در شاخه های مختلف مجاز نمی باشد. این شاخه ترجیحاً در درایو سیستمی (پیش فرض درایو C) قرار داده نشود.

۲.۹. هیچ بخشی از برنامه نرم افزاری نباید به تنظیمات و فولدرهای مربوط به **Users** وابسته باشد.

۲.۱۰. **Shortcut** اجرایی برنامه نرم افزاری حتماً باید بر روی دسکتاپ قرار داده شود. این **Shortcut** نباید به کاربر خاصی وابسته باشد (بطور مثال : در **Desktop** مربوط به **All User** قرار داده شود).

۲.۱۱. شاخه ای که حاوی فایل های برنامه است ترجیحاً نباید در درایو سیستمی (پیش فرض درایو C) قرار داده شود

۲.۱۲. برنامه نرم افزاری برای نصب و عملکرد ترجیحاً نباید نیازی به تغییر در **Registry**. تغییر در **Group policy**. تغییر در سطوح دسترسی فولدرهای سیستمی و درایو سیستمی (پیش فرض درایو C) ویندوز داشته باشد.

۲.۱۳. برنامه نرم افزاری برای نصب و عملکرد نمی باید به هیچ وجه تغییری در فایل های سیستمی مانند **DLL** ها انجام دهد، تمامی فایل های مورد نیاز می باید در فولدر اختصاصی برنامه قرار گیرند و بازنویسی و تغییر فایل های سیستمی سیستم عامل به هیچ عنوان مجاز نمی باشد (منجمله **DLL** های مربوط به صفحه کلید).

۳. موبایل اپلیکشین

۳.۱. روال نصب موبایل اپلیکشن باید به صورتی باشد که در هنگام نصب، هیچ دسترسی را مطالبه ننماید. درخواست دسترسی های سیستمی پس از نصب و در هنگام استفاده از هر بخش از سامانه می بایست انجام شده و فقط دسترسی های ضروری درخواست شود. اپلیکیشن نباید به جز دسترسی های کاملاً ضروری دسترسی اضافه ای را در اختیار بگیرد.

۳.۲. در هنگام ورود به اپلیکشن، و در صورت وجود نسخه جدید برنامه در سرور مرکزی، پیام مناسب به کاربر داده شده و از وی برای بروزرسانی برنامه دعوت شود.

۳.۳. در صورتیکه برای عملکرد صحیح و کامل اپلیکیشن، نیاز به بروزرسانی می باشد، از ادامه کار اپلیکیشن، جلوگیری شده و کاربر را اجبار به نصب نسخه جدید نماید.

۳,۴. در هنگام نصب نسخه جدید می بایست به صورت خودکار جایگزین نسخه قبلی شود (نیاز به Uninstall کردن نسخه قبلی نباشد) و همچنین تمامی تنظیمات و دیتاهای محلی حفظ شوند و نسخه جدید (بروزرسانی شده) با تنظیمات قبلی شروع بکار نماید.

۳,۵. برای بروزرسانی برنامه امکان همزمان استفاده از مارکت های رسمی و یا صفحه وب مخصوص برنامه وجود داشته باشد، ولی در هر صورت به هیچ وجه برای بروزرسانی برنامه نباید وابستگی به مارکت ها وجود داشته باشد و برنامه به عنوان پیشنهاد اصلی، کاربر را به صفحه وب Official اپلیکیشن هدایت کرده و امکان دانلود مستقیم نسخه جدید به کاربر ارائه شود.

۴. فونت و Code Page

۴,۱. برنامه نرم افزاری برای اجرا و عملکرد نباید نیاز به تغییر Code Page پیش فرض و یا نصب Code Page خاصی داشته باشد. Code Page مورد پذیرش انحصارا Unicode براساس UTF-8 می باشد.

۴,۲. برنامه نرم افزاری برای اجرای نباید نیاز به نصب فونت خاصی داشته باشد و می بایست از فونت های پیش فرض سیستم عامل (ترجیحا Tahoma) استفاده گردد.

۵. لایسنس و مالکیت

۵,۱. در صورتی که در جریان فرایند تولید و بهره برداری از برنامه نرم افزاری، اجزاء سیستم، و بانک اطلاعات آن، نیاز به استفاده از ابزارهای تجاری و یا نرم افزارهای تولید شرکتهای دیگر باشد (مانند کامپوننت های برنامه نویسی، برنامه های تهیه ی نسخه ی پشتیبان و...) مجری موظف به تامین لایسنس نرم افزارهای مذکور خواهد بود. به شکلی که هیچگونه مشکلی برای بهره برداری و استفاده کامل از برنامه نرم افزاری، وجود نداشته و نیاز به تهیه هیچ لایسنسی نباشد.

۵,۲. مالکیت مادی و معنوی بانک اطلاعاتی و اطلاعات ذخیره شده در آن (بجز روال ها، فانکشن ها، و کدهای برنامه نرم افزاری که سمت بانک اطلاعاتی ایجاد شده اند) به صورت کامل و بدون قید و شرط متعلق به سازمان فاوا و شهرداری می باشد. شیوه رفتار و تحویل گیری بانک های اطلاعاتی برنامه نرم افزاری مطابق با «دستورالعمل نصب، راه اندازی و استقرار برنامه های نرم افزاری شهرداری قم APD.IT.I.STD.007» میباشد.

۵,۳. در صورتیکه مالکیت مادی و معنوی برنامه نرم افزاری به شهرداری و یا سازمان فاوا منتقل شده است، مجری مجاز به نصب هیچگونه قفل نرم افزاری و یا سخت افزاری بر روی برنامه نرم افزاری و اجزاء آن نمی باشد.

۵,۴. در صورتیکه مالکیت مادی و معنوی برنامه نرم افزاری به صورت کامل به مجری تعلق دارد، مجری مجاز (و مکلف) است برای جلوگیری از بهره برداری غیرمجاز برنامه نرم افزاری، اقدام به نصب قفل کنترل کننده به صورت سخت افزاری و یا نرم افزاری بر روی آن نماید.

۵,۴,۱. توجه شود که مسئولیت قفل گذاری برنامه نرم افزاری با مجری بوده و در صورت عدم قفل گذاری و یا قفل گذاری ناکارآمد، که منجر به سرقت نسخه های برنامه نرم افزاری و یا تکنولوژی آن گردد، مسئولیتی متوجه شهرداری و یا سازمان فاوا نخواهد بود.

۵,۴,۲. قفل گذاری به شکلی باید انجام شود که سازگاری و تناسب کامل با استانداردها، دستورالعمل ها و رویه های امنیتی و اجرایی سازمان فاوا و شهرداری را داشته باشد.

- ۵,۴,۳. در صورت استفاده از قفل سخت افزاری (ویا نرم افزاری) مجری موظف است، انتخاب تکنولوژی قفل را به شکلی انجام و به سازمان فاوا تحویل نماید که به لحاظ فنی سازگاری کامل با تجهیزات سخت افزاری و نرم افزاری سازمان فاوا داشته باشد.
- ۵,۴,۴. هرگونه قفل گذاری که اجرا و یا بهره برداری از برنامه نرم افزاری را به مشخصات سخت افزاری و یا نرم افزاری محیط اجرا، سرویس دهی و یا ذخیره سازی داده ها محدود میکند ممنوع است.
- ۵,۴,۵. ترجیحا از روش قفل گذاری به روش استفاده از دانگل های سخت افزاری استفاده نشود (به دلیل مشکلاتی که در جابجایی و نگه داری سخت افزاری قفل ها و امکان از کارافتادن آنها و مشکلات و زمانبر بودن جایگزینی آنها در پی خواهد داشت).
- ۵,۴,۶. روش مطلوب قفل گذاری، استفاده از کدفعال سازی، شماره سریال و روش های مشابه سیستم های نرم افزاری حرفه ای و بین المللی می باشد. در چنین حالتی کدهای فعال سازی و شماره سریال ها و غیره طی صورتجلسه ای رسمی به نماینده رسمی معرفی شده توسط سازمان فاوا تحویل داده می شود.
- ۵,۴,۷. در صورتیکه قفل گذاری به روش دانگل سخت افزاری باشد، لازم است جهت امکان جایگزینی سریع آن در زمان خرابی، یک فقره قفل جایگزین دقیقا با امتیازات و مشخصات و امکاناتی که قفل اصلی ایجاد می کند، به صورت پلمب شده در اختیار حراست سازمان فاوا قرارداد شده شود، از این قفل جایگزین قفل در شرایط خرابی قفل اصلی و با تنظیم صورتجلسه مربوطه با حضور نمایندگان سازمان فاوا و شهرداری، فک پلمب خواهد شد.
- ۵,۴,۸. مجری موظف است نهایت همکاری را در جهت تعویض دانگل های معیوب در اسرع وقت بنماید. دانگل ها در طول مدت همکاری طرفین در گارانتی مجری هستند و تعویض آنها برای شهرداری و سازمان فاوا فاقد هزینه میباشد. پس از تعویض دانگل معیوب و راه اندازی دانگل جدید و تایید صحت عملکرد آن توسط سازمان فاوا، دانگل جایگزین مجددا پلمب شده و به حراست سازمان فاوا تحویل داده می شود.
- ۵,۴,۹. در صورت تمایل سازمان فاوا و برای مقاصد آموزشی و تستی و انجام آزمون های لازم (امنیتی، کارآمدی و ...) مجری موظف است یک نسخه مشابه از قفل سخت افزاری اصلی تحویل شده را، به صورت امانی، بدون هزینه و برای دوره های زمانی مشخص و قابل تمدید در اختیار سازمان فاوا قرارداد دهد.
- ۵,۴,۱۰. دوره های زمانی فوق می توانند شش ماهه و یا یکساله باشند.
- ۵,۴,۱۱. این موضوع صرفا با درخواست رسمی شهرداری و فقط به نماینده معرفی شده تحویل خواهد شد.
- ۵,۴,۱۲. در پایان دوره زمانی مشخص شده، مجری موظف به پیگیری برای استرداد و یا تمدید دوره زمانی می باشد.
- ۵,۴,۱۳. تمدید دوره زمانی با درخواست رسمی سازمان فاوا انجام خواهد شد.
- ۵,۴,۱۴. مجری مجاز است به نحو مقتضی بر استفاده سازمان فاوا از این قفل امانی نظارت نماید (نظارت مذکور نباید باعث بروز اختلال در عملکردها و فعالیت های مورد نظر سازمان فاوا گردد).
- ۵,۵. در صورتیکه مالکیت مادی و معنوی برنامه نرم افزاری به صورت مشارکتی فیما بین مجری و سازمان فاوا تقسیم شده است، علاوه بر استفاده از روش های قفل گذاری به روش های فوق الذکر، مجری موظف است یک نسخه از آخرین سورس و مستندات فنی کامل سیستم را مطابق با «دستورالعمل نصب، راه اندازی و استقرار برنامه های نرم افزاری شهرداری قم APD.IT.I.STD.007» به سازمان فاوا تحویل نماید.
- ۵,۵,۱. سورس تحویل شده باید کامل بوده و همراه کامپوننت های لازم و مربوطه تحویل داده شود به شکلی که در محیط Compiler های سازمان فاوا قابلیت ویرایش و Compile و Deploy داشته باشد.

۵,۵,۲. مستندات فنی کامل شامل تمامی مستندات کاربری، راهبری، تولید، تحلیل و طراحی، دیاگرام ها، دیتاستراکچر بانک های اطلاعاتی و به طور کلی تمامی مستنداتی است که برای شناسایی کامل سورس برنامه ها و امکان رفع اشکالات، توسعه و نگه داری آنها مورد نیاز می باشد.

۵,۶. در صورتیکه مالکیت مادی و معنوی برنامه نرم افزاری متعلق به مجری باشد، و لایسنس به صورت Per User خریداری شده باشد، مجری مجاز است تعداد کاربران استفاده کننده از برنامه را توسط یکی از روش های قفل گذاری محدود نماید.

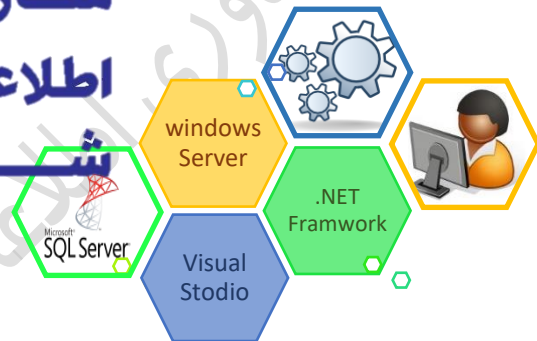
۵,۶,۱. در چنین صورتی برنامه می بایست، پس از تکمیل شدن تعداد کاربر خریداری شده، از ایجاد کاربر جدید جلوگیری نموده و این موضوع را به شکل پیام سیستمی مناسب به راهبر برنامه اعلام نماید.

۵,۶,۲. برنامه نرم افزاری نباید اجازه دهد کاربران مازاد بر تعداد لایسنس خریداری ایجاد شوند، ولی در هر صورت اگر به دلایلی نیز این اتفاق بیافتد (منجمله خطای بخش کنترلی برنامه)، برنامه نرم افزاری به شکلی باید عمل نماید که فقط آخرین کاربران مازاد ایجاد شده غیرفعال شوند و برای عملکرد سایر کاربران که در سقف مجاز تعریف شده باشند نباید هیچ گونه مشکلی ایجاد گردد.

اطلاعات و ارتباطات شهرداری قم



سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



فصل ششم

کارآمدی

Efficiency



۱. تعداد کاربران

برنامه نرم افزاری، باید قابلیت پاسخگویی به تعداد کاربر همزمان مورد نیاز بهره برداران، را داشته باشد. این تعداد در رابطه با هر برنامه نرم افزاری متفاوت بوده و متناسب با جامعه بهره برداران توسط سازمان فاوا تعیین میشود.

۲. قابلیت توزیع بار

برای افزایش کارآمدی، سرعت، و پایداری، می بایست برنامه نرم افزاری دارای قابلیت توزیع بار (منجمله : Load Balancing و Clustering) باشد.

۳. گواهینامه های کارآمدی

مجری موظف است برای همان نسخه از برنامه نرم افزاری که در شهرداری قم نصب می کند و یا در اختیار شهرداری قم قرار میدهد، گواهینامه رسمی از مراکز مورد تایید سازمان فاوا برای آزمون های کارایی زیر دریافت و ارایه نماید :

۳,۱ Load Test

۳,۲ Stress Test for n user

۳,۳ Configuration Test

۴. تکنیک های بارگزاری صفحات

به منظور افزایش سرعت سامانه های وب و پایین آمدن زمان بارگزاری صفحات، لازم است از تکنولوژی های بارگزاری جزئی اشیاء و اطلاعات مانند : Ajax استفاده شود.

۵. متوسط زمان بارگزاری صفحات

۵,۱ برای برنامه های نرم افزاری تحت وب، حداکثر زمان مجاز برای بارگزاری کامل هر صفحه، در شبکه درون سازمانی، کمتر از ۳ ثانیه می باشد.

۵,۲ برای برنامه های نرم افزاری ویندوز اپلیکیشن، حداکثر زمان مجاز برای بارگزاری کامل هر صفحه، در شبکه درون سازمانی، کمتر از یک ثانیه باشد.

۶. بهینه سازی معماری

۶,۱ در صورتیکه افزایش حجم تراکنشهای روی برنامه نرم افزاری، باعث کندی پاسخگویی سیستمی گردد. لازم است مجری از روش های بهینه سازی سرعت استفاده نماید. این روش ها شامل قابلیت های توزیع بار، بازطراحی و بهینه سازی معماری بانک اطلاعاتی، Database Cache, Server Bus و ... می تواند باشد.

۶,۲ برنامه نرم افزاری باید بصورت درونی برای بهبود سرعت، در انجام فرآیندها بهینه سازی شود و جبران کمبود سرعت سیستم بوسیله افزایش منابع سخت افزاری و شبکه ای قابل پذیرش نمی باشد.

۶,۳ در صورتیکه در برنامه نرم افزاری، نیاز به انجام فرآیندهای تکرار شونده و پردازش های محاسباتی باشد (مانند صدور فیش انبوه نوسازی و یا گزارش گیری های پیچیده و محاسباتی)، مسئولیت بهینه سازی معماری سیستم برای کارکرد صحیح و مطلوب برنامه نرم افزاری، در تمامی شرایط با مجری می باشد.

تعداد با سازمان فاوا³¹

۶,۴. در تولید برنامه های نرم افزاری از کمپوننت های سنگین که باعث کندی بارگزاری صفحات می شوند پرهیز شود.
۶,۵. برای افزایش سرعت و کارآیی سیستم، فایل های MDF و LDF بانک های اطلاعاتی SQL، در دو هارد یا پارتیشن جداگانه نصب شوند.

۶,۶. گزینه های **Auto Close**، **Auto Shrink**، **Compressing** موجود در **SQL Server** باعث کاهش کارآیی میشوند، لازم است این موارد غیرفعال شده و عملیات لازم به صورت دستی و یا از طرق مناسب دیگر انجام شوند.

۶,۷. ترجیحا از **SQL Server Priority** در قسمت **Processor** در صفحه مشخصات سرور استفاده شود.

۶,۸. ترجیحا در مورد بانک های اطلاعاتی با جداول حجیم، از قابلیت های **DB Partitioning** استفاده شود.

۶,۹. برای بهبود عملکرد برنامه نرم افزاری، و جلوگیری از کاهش سرعت عملکرد آن بخصوص پس از راه اندازی و شروع بهره برداری، مجری می باید، پیشنهادات لازم در خصوص **Tuning** پلات فرم به منظور افزایش **Performance** آن را به سازمان فاوا اعلام نماید.

۷. ارتباط با بانک های اطلاعاتی

۷,۱. تمامی **Session** های ارتباطی با بانک های اطلاعاتی باید فقط در زمان ارسال و یا دریافت اطلاعات برقرار و سپس بسته شوند، به این معنا که از برقراری ارتباطات با بانک های اطلاعاتی به صورت **Session** های باز و ممتد خودداری شود.

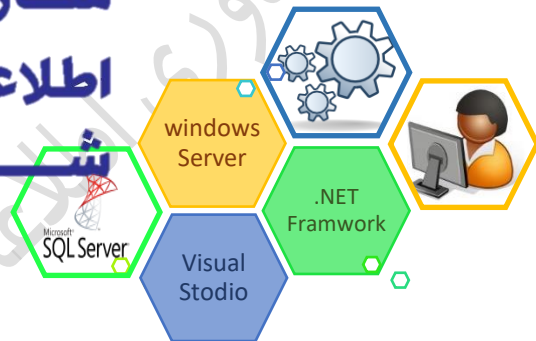
۷,۲. پس از بسته شدن برنامه، کلید کانکشن ها به بانک های اطلاعاتی و سرور بسته شده و هیچ کانکشنی باز نماند.

۷,۳. تمامی ارتباطات با بانک های اطلاعاتی باید فقط از طریق لایه **Data Access Layer** برقرار شود.

۷,۴. تمامی ارتباطات با بانک های اطلاعاتی باید در سطح قابل قبولی امن سازی شوند به شکلی امکان حملات هکری به حداقل کاهش پیدا کند.



سازمان فناوری
اطلاعات و ارتباطات
شهرداری قم



فصل هفتم

نگه داری

Maintainability

۱. موتور تولید و مدیریت گردش کار Workflow Management Engine

برنامه های نرم افزاری که موتور تولید و مدیریت گردش کار هستند می بایست واجد الزامات زیر باشند:

- ۱.۱. در صورتیکه نرم افزار مورد نظر، واجد موتور مدیریت گردش کار باشد، لازم است پنل Visual با قابلیت کاربری ساده و آسان و بدون نیاز به کدنویسی، برای مدیریت آن (ایجاد، ویرایش، حذف) در سیستم وجود داشته باشد
- ۱.۲. برنامه نرم افزاری باید دارای قابلیت برقراری ارتباط بین گردش کارهای تولید شده و فرآیندهای داخلی برنامه باشد به صورتیکه بتوان برای فرآیندهایی که نیاز به بررسی شروط و تصمیم گیری بین چند گزینه مختلف دارند، گردش کار مورد نیاز را تعریف و به فرآیند مورد نظر مرتبط نمود.
- ۱.۳. موتور مدیریت گردش کار، همچنین باید قابلیت خودکار سازی فرآیندهای مورد نیاز را به سیستم اضافه نماید به صورتیکه پس از دریافت یک ورودی و یا انتخاب یک گزینه و یا صدور یک فرمان توسط کاربر، برنامه نرم افزاری، براساس گردش کار تعریف شده، بخش های بعدی فرآیند را به صورت خودکار به انجام رساند.
- ۱.۴. موتور مدیریت گردش کار، باید دارای کنترل های داخلی برای جلوگیری از ایجاد و یا تعریف گردش کارهای ناقص، مشکل دار، دارای لوپ و حلقه های نامتناهی، دارای تناقضات منطقی و امثالهم باشد. و ضمن جلوگیری، با نمایش پیام های مناسب، اطلاعات کافی در خصوص مشکلات مربوطه را به کاربر ارایه نماید.
- ۱.۵. کنترل های لازم در هنگام ویرایش یک گردش کار اعمال شوند به صورتیکه برای روال ها، اسناد، نامه ها، پرونده ها و سایر مواردی که براساس آن گردش کار در حال گردش می باشند، مشکلی بوجود نیاید. در این راستا موضوعات زیر رعایت شوند:

- ۱.۵.۱. هیچ یک از موارد فوق الذکر، مفقود نشود.
- ۱.۵.۲. هیچ یک از موارد فوق الذکر، بلا تکلیف رها نشود.
- ۱.۵.۳. هیچ تاییدیه، هامش، دستور و امثالهمی مفقود نشود.
- ۱.۵.۴. گردش کار فعلی حذف نشود و تغییر نکند و به صورت ورژن قبلی ذخیره شود و گردش کار جدیدی با تعریف جدید ایجاد و جایگزین آن شود.
- ۱.۵.۵. بتوان مشخص کرد که مواردی که گردش آنها با گردش کار قبلی شروع است براساس همان گردش کار به اتمام رسند و یا با گردش کار جدید به گردش ادامه دهند.

- ۱.۶. موتور مدیریت گردش کار و تمامی بخش های مرتبط با آن می بایست استانداردهای حاکم بر این موضوع (WFMC Standards , WFMC Coalition) را رعایت نموده باشد.
- ۱.۷. مثال هایی از کاربردهای این بخش: قابلیت ارسال خودکار رونوشت های یک نامه به گیرندگان پس از ثبت در دبیرخانه، قابلیت ارسال یک سند مالی پس از تایید توسط یک کاربر به کاربر بعدی برای بررسی و تایید، و ...

۲. موتور مدیریت قوانین سامانه

- در صورتیکه برنامه نرم افزاری، دارای فرآیندهایی است که در آنها پردازش قوانین و یا ضوابط قانونی مورد استفاده قرار می گیرند، لازم است موتور مدیریت قوانین، با قابلیت های زیر، در آن وجود داشته باشد.
- ۲.۱. برنامه نرم افزاری، باید دارای پنل Visual با قابلیت کاربری ساده و آسان و بدون نیاز به کدنویسی، برای مدیریت قوانین (ایجاد، ویرایش، حذف) و به جریان انداختن آنها باشد.

۲.۲. برای هر قانونی که در برنامه نرم افزاری درج می شود، امکان فعال و غیرفعال سازی، تعیین تاریخ شروع اعمال و تاریخ اتمام اعتبار آن (و یا بدون تاریخ اتمام) وجود داشته باشد.

۲.۳. قوانین نباید در سورس برنامه و یا فایل های همراه (مانند فایل ها DLL)، هاردکد شوند، قوانین باید به صورت اطلاعات، در بانک اطلاعاتی ذخیره شده و در زمان نیاز از آن فراخوانی و استفاده شوند.

۲.۴. همچنین روشی برای برقراری هر قانون درج شده در برنامه نرم افزاری، با یک یا چند فرآیند برنامه، وجود داشته باشد که بدون نیاز به کدنویسی یا تغییر در سورس برنامه بتوان یک قانون جدید را تعریف و به یک یا چند فرآیند متصل نمود.

۲.۵. بررسی عدم مغایرت های قوانین : برنامه نرم افزاری، باید دارای روال های خودکنترلی به شکلی باشد که از وارد شدن قوانینی که تناقضات منطقی، دارای همپوشانی یا نقض همدیگر هستند جلوگیری نماید.

۲.۶. مثالهایی از کاربردهای این بخش : درج ضوابط طرح تفصیلی در سیستم شهرسازی و بررسی و درخواست های صدور پروانه با آنها، و یا درج قوانین کار در سیستم منابع انسانی و اعمال آنها در قراردادهای المانهای مربوط به قراردادهای کارگری می باشد.

۳. موتور مدیریت فرمول های محاسباتی سیستم

در صورتیکه در برنامه نرم افزاری، نیاز به محاسبات عددی یا ریالی وجود داشته باشد که بر مبنای فرمول های محاسباتی عمل می کنند، برنامه نرم افزاری، باید دارای موتور مدیریت خودکار فرمول های محاسباتی با مشخصات زیر باشد :

۳.۱. برنامه نرم افزاری، باید دارای پنل Visual با قابلیت کاربری ساده و آسان و بدون نیاز به کدنویسی، برای مدیریت فرمول ها (ایجاد، ویرایش، حذف) و به جریان انداختن آنها، و اتصال هر فرمول به یک یا چند فرآیند باشد.

۳.۲. برای هر فرمول که درج می شود، امکان فعال و غیرفعال سازی، تعیین تاریخ شروع اعمال و تاریخ اتمام اعتبار آن (و یا بدون تاریخ اتمام) وجود داشته باشد.

۳.۳. فرمول ها نباید در سورس برنامه و یا فایل های همراه (مانند فایل ها DLL)، هاردکد شوند بلکه باید به صورت مقادیری در بانک اطلاعاتی ذخیره شده و در زمان نیاز از آن فراخوانی و استفاده شوند.

۳.۴. موتور مدیریت فرمول ها باید بتواند سوابق فرمول های قبلی مربوط به یک فرآیند را حفظ کرده و در هنگام محاسبات معوقه هر دوره زمانی را با توجه به زمان آن از فرمول متعلق به همان دوره زمانی استفاده نماید.

۳.۵. مثال های از کارکردهای این بخش : فرمول های محاسباتی حقوق و دستمزد، فرمول های محاسباتی عوارض نوسازی و ...

۴. موتور تولید و مدیریت فرم ها (ماژول فرم ساز)

در صورتیکه برنامه نرم افزاری دارای ماژول فرم ساز باشد لازم است قابلیت های زیر در آن وجود داشته باشند :

۴.۱. باید توانایی ایجاد خودکار بانکهای اطلاعاتی مرتبط با فرم های ایجاد شده، شامل : جداول، فیلدها، ارتباطات و سایر المانهای مربوطه را داشته باشد.

۴.۲. بانک های اطلاعاتی که به این شکل به صورت خودکار تولید می شوند می بایست دارای استانداردهای لازم براساس اصول طراحی و مهندسی بانکهای اطلاعاتی باشند (شامل : Unique Key , Indexing , Relationship , Cascade Update , Cascade Delete , نرمال سازی حداقل تا سطح سه (3NF : Third Normal Form) و...)

۴,۳. فرم های ایجاد شده باید با استفاده از روش های آسان و بدون نیاز به کد نویسی، قابلیت فراخوانی در سیستم را داشته باشند.

۴,۴. ماژول فرم ساز باید قابلیت استفاده از انواع فیلدهای موجود در بانک های اطلاعاتی مرتبط با خود را داشته باشد.

۴,۵. در ماژول فرم ساز باید قابلیت ارتباط فرم های ساخته شده و فیلدهای آنها با **WEB Service** های استاندارد به منظور دریافت، نمایش و پردازش اطلاعات وجود داشته باشد.

۵. موتور تولید و مدیریت گزارشات (ماژول گزارش ساز)

در صورتیکه برنامه نرم افزاری، دارای ماژول گزارش ساز باشد لازم است قابلیت های زیر در آن وجود داشته باشند:

۵,۱. ماژول گزارش ساز باید قابلیت برقراری ارتباط کامل با تمامی اجزاء بانک های اطلاعاتی خود را برای ایجاد گزارشات مورد نیاز، با روش های سهل و آسان و بدون نیاز به کد نویسی را داشته باشد.

۵,۲. ماژول گزارش ساز باید قابلیت استفاده از تمامی فیلدهای موجود در بانک اطلاعاتی خود را فراهم آورد. همچنین امکان تعریف فیلترهای مختلف برای محدود سازی دامنه گزارش و تعیین دامنه های مختلف را فراهم نماید.

۵,۳. در ماژول گزارش ساز باید قابلیت ارتباط با **WEB Service** های استاندارد برای دریافت، نمایش و یا پردازش اطلاعات مورد نیاز وجود داشته باشد.

۵,۴. ماژول گزارش ساز باید قابلیت ایجاد خروجی های استاندارد از گزارشات را فراهم آورد (مانند: Excel, PDF, ...)

۵,۵. ماژول گزارش ساز باید قابلیت ارسال خروجی چاپی تمامی گزارشات ساخته شده را روی چاپگرهای استاندارد به صورت انعطاف پذیر داشته باشد.

۵,۶. دارای قابلیت ایجاد داشبوردهای مدیریتی با امکان سفارشی سازی آنها در سطوح مختلف سازمانی، باشد.

۶. قابلیت پشتیبانی و توسعه

۶,۱. بعد از استقرار برنامه نرم افزاری، باید قابلیت پشتیبانی و توسعه آن از جهات مختلف موجود باشد. معماری برنامه نباید توسعه آنرا محدود سازد. همچنین ابزارهای کمکی برای سهولت فرآیند پشتیبانی باید وجود داشته باشند.

۶,۲. فرآیندها و انتظارات ذی نفعان، با گذشت زمان تغییر می کند و برنامه نرم افزاری باید به گونه ای تولید و مستند سازی شده باشد که توسعه های آتی آن بدون تغییرات اساسی در ساختار، امکانپذیر باشد.

۶,۳. در صورت تغییر در نسخه برنامه نرم افزاری، باید نسخه جدید با نسخه های قبلی آن سازگاری منطقی داشته باشد. بطور مثال: فرآیندهای انجام کار، تنظیمات محیطی و کاربری، طراحی های گزارشات، نماهای فرمها و امثالهم نبایست تغییر نمایند.

۶,۴. برنامه نرم افزاری باید دارای شماره گذاری نسخه (ورژن) براساس روش های مرسوم و قابل قبول باشد و در صورت توسعه و یا بروزرسانی تغییرات، شماره نسخه ها نیز به روش معناداری تغییر نماید.

۷. وب سرویس ها

۷,۱. برنامه نرم افزاری، باید قابلیت برقراری ارتباط در سطح تبادل دیتاها (و در صورت نیاز فرآیندها) با سایر برنامه های نرم افزاری درون یا برون سازمان را به شرح ذیل داشته باشد:

۷,۲. سعی شود حداقل مقدر برای روال های اصلی و پرکاربرد درون برنامه از وب سرویس استفاده گردد. به این معنا که ارتباط بین اجزاء برنامه و بانک های اطلاعاتی ترجیحا از طریق وب سرویس برقرار شود.

۷,۳. در صورت ارایه زیرساخت ارتباطی (ESB (Enterprise Service BUS توسط سازمان فاوا، لازم است برنامه نرم افزاری، قابلیت ارتباط و تبادل اطلاعات و فرآیندها را با آن، مطابق با مستند الزامات فنی ارتباط با ESB مصوبه سازمان فاوا را داشته باشند.

۷,۴. در صورت اعلام سازمان فاوا، برنامه نرم افزاری باید قابلیت ارتباط با سایر برنامه های نرم افزاری، بدون ارتباط با ESB را با استفاده از وب سرویس های استاندارد، دارا باشد.

۷,۵. مثال هایی از ارتباط وب سرویسی : استفاده از نقشه ها و اطلاعات جغرافیایی، سرویس های احراز هویت و ...