

تاریخ: ۱۴۰۰/۱۰/۲۱

پیوست : ندارد

بسمه تعالی

پاسخنامه آزمون تشریحی معماری نرم افزار نیمسال اول ۱۴۰۰-۱۴۰۱ نام استاد جناب دکتر سید علی رضوی ابراهیمی

نام دانشجو : سید جواد بسیطی - شماره دانشجویی : ۹۹۰۱۸۹۹۸۹ - تلفن : ۰۹۱۳۱۵۳۵۷۷۴

سوال یک میانترم:

معماری نرم افزار چیست و نحوه ارزیابی یک معماری طرح شده را شرح دهید؟

معماری یک سیستم نرم افزاری ، مجموعه ای از ساختارهای مورد نیاز و مناسب یک سیستم نرم افزاری است که شامل عناصر ، چیدمان و روابط بین آنها و ویژگی ها و خصوصیات این دو است. عملا ایده اصلی در معماری تقسیم نرم افزار به اجزاء آگاهانه است و از آن سه رسته مهم ماژولها ، کامپوننت و اتصال دهنده ها و ساختارهای تخصیص را می توان نام برد. به عبارتی معماری ابزار پیاده سازی کیفیت است.

روشهای ارزیابی معماری در دسته بندی های مختلفی قرار می گیرد، یکی از دسته بندی های مهم روشهای مبتنی بر سناریو است ، نحوه ارزیابی توسط چک لیستهای طراحی (صفات کیفی) و مبتنی بر سناریو صورت می پذیرد.

فرایند ارزیابی معماری، مستندات معماری نرم افزار، توضیحات مشکل و خصوصیات نیازمندیها را بعنوان ورودی دریافت می کند. سپس، مستندات معماری تصحیح شده، بهبود داده شده و سازماندهی شده را بعنوان خروجی فرایند ارزیابی تولید می کند. مستندسازی معماری نرم افزار شامل موارد مهمی است مانند: دیدگاه های معماری، خصوصیات المانهای معماری، اتصال دهنده ها و دیگر اطلاعات مهم درباره چگونگی ارتباط المانهای معماری، و خصوصیات قیدها که در تصمیمات معماری تاثیر دارند. تیمی که ارزیابی یک معماری را هدایت می کنند، از انواع مختلف ذینفعان تشکیل می شود، نظیر مدیران پروژه، سرپرستان سیستم، معماران، توسعه دهندگان و کاربران. نقش هر ذینفع مطابق با حوزه دانشش تعیین می شود.

ارزیابی مبتنی بر سناریو (SAAM):

هدف روش تحلیل معماری نرم افزار مبتنی بر سناریو بررسی صحت مفروضات و اصول معماری پایه در برابر مستنداتی است که ویژگیهای مطلوب یک برنامه کاربردی را توصیف کند. این تحلیل، به ارزیابی ذاتی ریسک ها در یک معماری کمک می کند. بازرسی معماری، تمرکز بر نقاط بلقوه اشکال نظیر تداخل نیازمندیها یا مشخصات طراحی ناقص از دید یک ذینفع منحصر بفرد را هدایت می کند. در مجموع این روش، به مقایسه معماری های نرم افزار کاندید شده کمک می کند، شش فعالیت مهم وجود دارد که عبارتند از:

۱) مشخص کردن نیازمندیها و محدودیتهای طراحی،

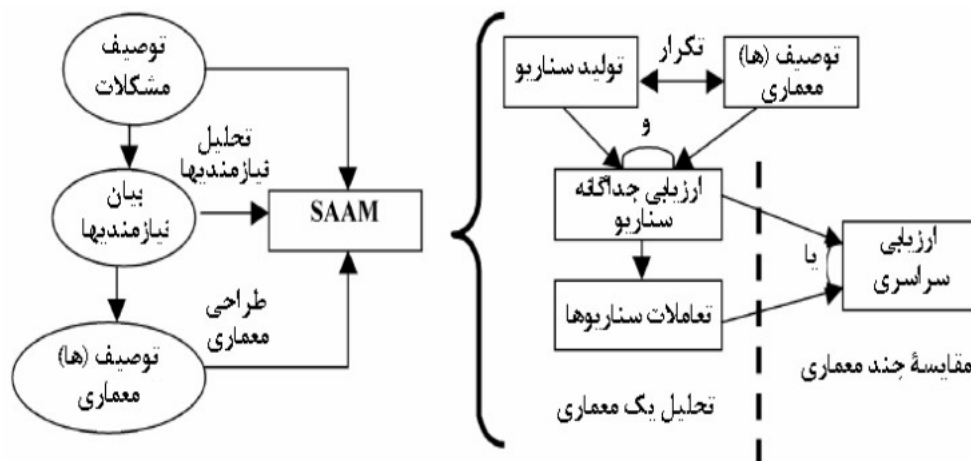
۲) توصیف معماری نرم افزار،

۳) چک کننده محدودیت،

۴) استخراج سناریوها،

۵) اولویت بندی حالات

۶) ارزیابی معماری ها با توجه به سناریوها



سوال یک پایانترم:

۱- مدل یک صفت کیفی را بطور کامل شرح دهید.

تا اینجا گفته شد که هر سیستم یک معماری دارد ولی جهت رسیدن به کیفیت و یا معماری مطلوب ، معماری سیستم باید ارزیابی گردد و پایه این ارزیابی از طریق تعریف و مستند سازی صفات کیفی است ابتدا تعریفی از صفات کیفی : نیازمندیهایی که مشخص کننده کیفیت عملکرد است و نه خود عملکرد، به اصطلاح غیر عملکردی است و عملا کیفیت عملکردها را توصیف می کند به عبارتی اگر قرار هست فلان عملکردی در سیستم اجرا شود آن عملکرد با چه سرعتی مد نظر است که حداقل کیفیت مطلوب را ارایه دهد. یا مثلا توصیف میزان انعطاف پذیری نرم افزار در زمان ورودی عملکرد یا مثالی دیگر میزان سهولت یادگیری عملکرد و غیره ... به طور کلی خصوصیات کیفی عملکرد ها را توصیف می کند. حال بیان این مدل صفات کیفی به صورت - (قالب) - یکسری از سناریو ها شرح میگردد ، که این سناریو ها در حالت عمومی و کانکریت (معین) تعریف میگردد

سناریوها در حالت عمومی با شرح ۶ بخش توصیف می گردد و در جدول قرار می گیرند.

- ۱- محرک (Simulus) (مثلا شرطی که نیاز دارد برای باز خورد روی بدهد)
- ۲- منبع محرک (Source of stimulus) موجودیتی که محرک را ایجاد می کند.
- ۳- پاسخ (Response) پاسخ سیستم
- ۴- اندازه گیری پاسخ (Response measure) - تعریف شکل پاسخ در اینجا پارامترها به کمیت تبدیل می شوند و نحوه اندازه گیری تشریح می گردد
- ۵- محیط (Environment) (پارامترها و شرایط محیطی محرک در آن شرایط وارد میشود)
- ۶- محصول (Artifact) - (کدام بخش از سیستم)

در بالا (بخش قبل) شرح سناریوهای عمومی بود که به طور بالقوه می تواند مربوط به هر سیستمی و هر صفت کیفی باشد ولی خارج از موارد سناریوهای است که تنیده با همان صفت کیفی و خاص آن سیستم است و باید تشریح گردد که سناریوهای معین ما هستند

تاکتیکها - طرحهای معماری بدوی

خوب بخش شیرین قضیه تاکتیکهایی است که برای دستیابی و تشریح موارد بالا باید به کاربرد این تاکتیکها می توانند کمک گیری از یکسری الگوها (patterns) هایی باشند که معماران قبلا به کار برده اند و در عمل برای نیل به هدفی تکامل یافته اند لذا مهم است که برای هر هدف یا صفات کیفی خاص مجموعه سیستم خود از آنها استفاده کنیم و از بکار بردن تاکتیکهای اختراعی خود که در محیط واقعی تست فراوان نشده اند پرهیز کنیم.

سوال دو پایانتروم:

صفت کیفی نمونه ای که اینجا مناسب میبینم مثال بزنم :

ابتدا مقدمه ای گفته شود در بحث نرم افزارهایی که با هدف عملیات ورود داده حجیم تولید می گردند ، مثلا در پروژه های آرشیو اطلاعات قدیمی ارگانها و داده آمایی آنها و به طور خاص برای نرم افزارهایی که برای اینگونه عملیات ورود حجیم داده تهیه و تولید می گردند مشاهده میشود که بیشترین سرعت ورود اطلاعات برای انجام کار اپراتور لازم است و اینکار عموما با تسلط ده انگشتی روی کیبرد انجام پذیر می باشد بدون توقف برداشت دست از روی کیبرد، بنابر این هر گونه لزوم استفاده از Mouse بسیار سرعت اپراتور را کند می کند، و این بیشترین سرعت صرفا با تسلط ده انگشتی روی تمام دکمه های کیبرد حاصل شده بدون برداشت دست از روی کیبرد.

بنابر این اینجا می خواهیم در اینجا یک صفت کیفی به نام `MouseLessAbility` یا `FullShortkeyAbility` یا `speed Entry` تعریف کنم که تا سنجشی باشد برای کمترین الزام استفاده از `mouse` و یا بیشترین سرعت بهره وری از کیبرد و کلیدهای سرعتی داخلی آن البته گرچه این صفت بیشتر خصوصیت UI است و عموما در فرانت اند مطرح است و کمتر در بخشهای دیگر ولی گاهی در `Back-end` (در الگوی معماری سرویس گرا) تاثیر می گذارد و بی تاثیر نیست و مثلا متد و امکاناتی در آنجا نیز گاهی مورد لزوم می شود. و مهمتر از همه انتخاب معماری مناسب است مثلا اجزاء و ابزار بکار گیری در تولید در پیاده سازی این صفت کیفی بسار خود را نمایان می کند ، شاید در طراحی برای این صفت اصلا ابزارهای `Desktop` ارجح از وب قرار گیرد و یا اگر الزام به وب است کامپوننت و اجزایی با بیشترین کارایی `AJAX` طراحی گردد و ارتباطات سیستم با این اجزاء که معمار برگزیده طراحی گردد.

سناریوی عمومی آن:

- محرک : منو یا قسمتی از نرم افزار که الزاما در زمانی باید با موس انتخاب شود . و یا شرایطی که در آن زمان **Shortcut** تعیین شده برای عملکردی از کار بیافتد
- ۲ منبع تحریک : اپراتور - کاربر ورود داده
- ۳ پاسخ : اجرا شدن عملکرد با موس
- ۴ اندازه گیری پاسخ : برای اجرای هر **use case** (یک بخش کامل ورود داده ای- مثلا یک پرونده) از ابتدا تا انتها صفر بار مجاز است نیاز به بکار گیری موس شود.

۵- محیط : باز شدن برخی دیاگنها روی یکدیگر که الزام به حرکت و استفاده ز موس را مجاب می کند.

۶- محصول : ثبت یک پرونده اطلاعاتی بدون بکار گیری موس

سناریوی خاص:

تاکتیک:

الگوها و پترن های تست شده موجود عموما بدین گونه اند و با مختصر خصوصی سازی این طرح ارائه می شود:

- استفاده از دکمه اینتر برای عبور و جابجایی از هر المان روی صفحه به المان دیگر
- چنانچه نشانگر (کرسر) روی دکمه های اجرایی (Button) قرار گرفت کلید **Space** نسبت به اجرای عملکرد اقدام می کند
- توجه: دکمه **Enter** در اینجا نیز عمل جابجایی را انجام می دهد و نه اجرای عملکرد باتون
- جابجایی بر روی صفحات و دیاالوگهای ورودی با شورتکاتهایی مثلا کلید کنترل و اعداد صورت می گیرد

با تشکر از حوصله جنابعالی – سید جواد بسیطی