

آزمون پایان ترم معماری نرم افزار - استاد: دکتر رضوی

دانشجو: علیرضا شیاوسی - ۹۹۰۱۸۲۱۹۱

پاسخنامه در ۸ صفحه و به صورت PDF است.

پاسخنامه میان ترم

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

تعریف اصلی معماری نرم افزار این گونه است:

معماری نرم افزار یک سیستم، مجموعی از ساختار (سازه structure) ها است که برای استدلال سیستم مورد نیاز است. این ساختارها متشکل از عناصر نرم افزار، روابط بین آنها و خصوصیات (Properties) آنها است.

به صورت کلی معماری نرم افزار یعنی ساختار یا درواقع چیدمان و ترکیب ساختارها.

درست مثل ساختمان (من مهندس عمران هستم و با اجازه برای توضیح کامل از تمثیل و تشبیه به معماری ساختمان استفاده می کنم).

تصور کنیم که سازه در ساختمان معادل ساختار در نرم افزار باشد. انواع سازه داریم مثل فلزی (فولادی)، بتنی، مصالح خاص مثل بنایی و المان های مختلف که هرکدام رفتار خاص مثل رفتار الاستیک، الاستو پلاستیک و پلاستیک دارند.

المان های نرم افزاری را هم انواع مصالح، مبلمان و وسایل و ابزارها در نظر بگیریم.

یک معمار ساختمان درواقع از روش‌های موجود چیدمان و نوع و ترکیب ساختمان را طراحی می‌کند.

از سازه گرفته تا روش چیدمان و روابط فضاها. یعنی چیزی که قرار است برای کار خاصی طراحی شود و مدت مدیدی کاربرد داشته باشد. مثلاً برای یک ساختمان اداری نوع پلان، روابط فضاها، ابعاد و حتی مدل سازه و وسعت درب و ورودی و خروجی براساس میزان رفت و آمد طراحی می‌شود. بعد از آن اگر معماری به درجه بالاتری بطلبد معماری داخلی که از اجزاء ساختمان گرفته تا حتی شیرآلات را طراحی می‌کند و چیدمان مبلمان (دکوراسیون). در ادامه حتی معماری منظر یا لنداسکیپ نیز انجام می‌شود. یعنی هر بار مراحل بهتر و تکمیل‌تر می‌شوند.

در معماری نرم افزار هم همینطور است براساس بینش معمار، متدها یا همان روش‌ها و تکنولوژی‌ها موجود براساس نیاز سیستمی که قرار است طراحی شود، معماری نرم افزار آن را طراحی می‌کند. برای این طراحی از ساختارهای موجود، عناصر و غیر استفاده می‌کند و با ایجاد روابط بین آن‌ها و خواص و وضعیت‌ها یک سیستم که نیازها را برطرف و هدف را میسر می‌کند را طراحی می‌نماید.

پس نتیجه می‌گیریم معماری نرم افزار نیز یک علم و هنر است برای به کارگیری ساختارها، عناصر نرم افزاری، روابط بین آن‌ها و خواص آن‌ها برای رفع نیاز ساخت یک سیستم با هدف مشخص برای کاری یا کارهایی مشخص.

همانطور که معماری ساختمان از منظر زیبایی، کاربردی بودن، کیفیت ساخت و انواع معیارهای کیفی سنجیده می‌شود، معماری نرم افزار نیز از مناظر مختلف کیفی قابل سنجش است. مثل امنیت، عملکرد، دسترس پذیری و

برای ارزیابی کیفی می‌توان از روش سناریو استفاده کرد.

روش یا تکنیک سناریو همانطور که از نامش پیداست با استفاده از یک سناریو (نمایش نامه) کسب و کاری از منظر ذینفعان و مشتریان است.

برای اینکار نیازمندی‌ها و خواسته‌های ذینفعان را با استفاده از سناریوهای مختلف استخراج کرده و با معماری متناظر می‌کنند.

سناریو به این شکل است که تعاملات بین ذینفعان و سیستم را شبیه سازی می‌کند و بررسی می‌کند که آیا آن‌ها به هدف خود می‌رسند یا نه؟ آیا نیاز آن‌ها برطرف می‌شود؟ و همچنین میزان برآورده شدن خواسته‌ها و به هدف رسیدن را بررسی می‌کنند. تکنیک‌های دیگری نیز قابل استفاده هستند مانند استفاده از پرسشنامه‌ها و مصاحبه‌ها اما روش سناریو از منظر کیفی می‌تواند به صورت تخصصی هر سیستم را مجزا بررسی و برای همان سیستم و ذینفعان همان پروژه اجرا گردد.

پاسخ نامه پایان ترم

۱. مدل یک صفت کیفی را به طور کامل شرح دهید.

صفت: Modifiability به معنای قابل تغییر بودن یا اصلاح پذیر بودن

صفت اصلاح پذیری میزان تغییر براساس زمان و هزینه را بررسی می کند. در واقع یک سیستم معماری شده از کیفیت اصلاح پذیر بودن برخوردار است که بتوان در آن تغییرات را با سرعت بالا (زمان کم) و با هزینه اندک انجام داد.

تغییرات ممکن است توسط توسعه دهنده ها، کاربرنهایی، نصب کننده های نرم افزاری مثل آپدیت های جدید یک نرم افزار اتفاق بیافتند. در اصلاح پذیری یکسری تاکتیک وجود دارند که سیستم را برای تغییرات آینده آماده می کند.

تاکتیک ها به شرح زیرند:

ساخت ماژول های کوچکتر: کوچکتر کردن هر ماژول باعث می شود که اصلاح پذیری بالا برود یعنی تغییرات راحتتر انجام شود. این راحتی یعنی اینکه زمان کمتر و هزینه کمتری ببرد. برای مثال اگر یک ماژول داشته باشیم که چندین کار می کند برای آپدیت یک بخش آن کل مجموعه آن ماژول درگیر می شود و می تواند حتی سبب ایجاد افزایش زمان اجرا در کل برنامه یا halt شدن آن تا زمان ایجاد تغییر شود. اما اگر همین ماژول را به ماژول های کوچک تقسیم کنیم که هرکدام فقط یک کار انجام می دهند، با اینکار هزینه ایجاد اصلاح و تغییر (مثل آپدیت کردن) کاهش می یابد و از بروز مشکلات دیگر جلوگیری می شود. زیرا فقط و فقط یک ماژول کوچک درگیر تغییر است و باقی ماژول ها پابرجا و درحال ارائه خدمت خواهند بود.

کاهش اتصال (Reducing coupling): یک مجموعه تاکتیک است که شامل تاکتیک‌های مثل کپسوله سازی encapsulation و محدود کردن وابستگی‌ها dependencies و معنا سازی یا انتزاع سازی سرویس‌های عمومی و کلی abstracting و ریفتور کردن refactoring و غیره می‌باشد.

افزایش انسجام یا پیوستگی (cohesion): در این روش پاسخ‌گویی‌ها یا به قولی مسئولیت‌ها (responsibility) را که هدفشان یک سو نیست یعنی برای یک هدف و منظور به خدمت گرفته نمی‌شوند را از یکدیگر جدا می‌کنیم. با اینکار از آشفتگی جلوگیری و دسته‌بندی‌های منظمی خواهیم داشت که باعث انسجام قسمت‌ها و انسجام کلی معماری می‌شود. در اینصورت تغییرات با هزینه کمتر و سرعت بالاتری اعمال می‌شود زیرا: ۱. نیاز به جداسازی حین تغییر نیست ۲. هزینه جستجو و دیباگینگ پایین می‌آید

به تعویق انداختن الزامات: این هم یک مجموعه تاکتیک است که بر روی زمان ساخت یا بیلد تایم، زمان بارگذاری load time و زمان اجرا اثر می‌گذارد.

همه این تاکتیک‌ها سبب می‌شوند تا هزینه و زمان کاهش یافته و در بودجه تعیین شده قابل اجرا باشند.

۲. یک صفت کیفی تخیلی را طراحی و جداول سناریوهای عمومی آن و یک نمونه سناریو معین برای آن تعریف نمایید.

صفت: پرحرف

برای تعریف سناریوی یک صفت کیفی به ۶ شاخص یا بخش از سناریو (Portions of scenario) نیاز داریم که ۴ تای آن هسته اصلی هستند و ۲ تای آن مهم برای تعریف صفت. این شاخص ها عبارتند از:

(۱) Stimulus یا محرک

(۲) Source of stimulus یا منبع محرک

(۳) Response پاسخ

(۴) Artifact مصنوع

(۵) Environment محیط

(۶) Response measure اندازه گیری پاسخ

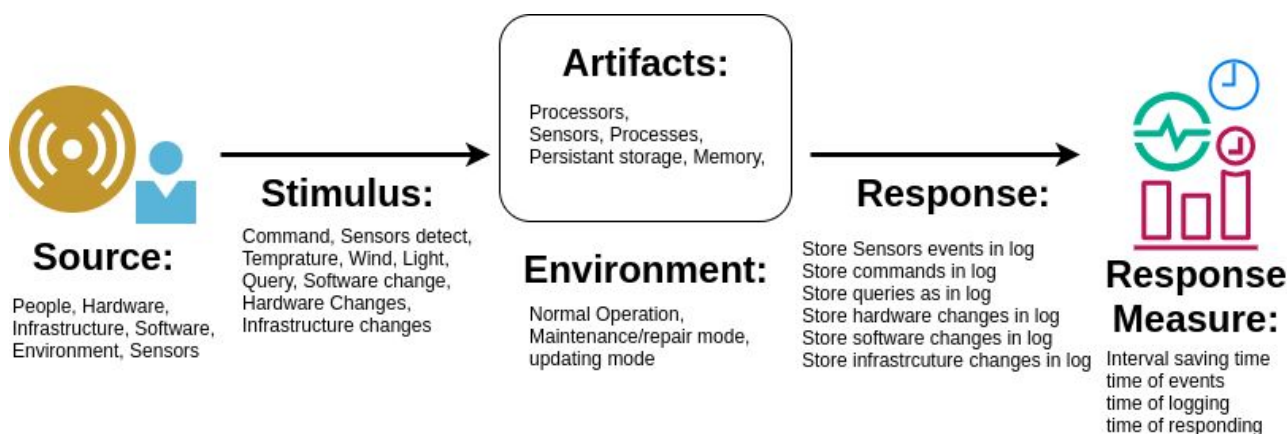
* شماره های ۱و۲و۳و۴ هسته اصلی ساخت سناریو و شماره های ۵و۶ دوتای مهم دیگر هستند.

تعریف سناریو: در این سناریوی فرضی، من فرض کرده ام که معماری برای یک سیستم گسسته و سنسور محور تعبیه می شود. برای مثال سنسورهای هواشناسی یا زمین شناسی یا راهنمایی رانندگی و غیره. برای تعریف سناریوی صفت پرحرف (Verbose) از جدولی که در ادامه ترسیم می کنم استفاده کرده ام.

جدول سناریوهای عمومی

بخش سناریو	ارزش‌های محتمل
منبع محرک	داخلی/خارجی: افراد، سخت افزار، نرم افزار، زیرساخت، محیط، سنسورها
محرک	دستور ورودی، تغییر دما، تغییر وزش باد، تغییر نور، کوثری، تغییر سخت افزار (خراب شدن/تعویض/آپدیت) تغییر نرم افزار (خراب شدن/تعویض/آپدیت)، تغییر زیرساخت
محیط	عملکرد عادی، حالت تعمیر، در حال به روزرسانی
مصنوع	پردازشگرها، فرایندها، ذخیره ساز دائمی، حافظه، سنسورها
پاسخ	ثبت وقایع محیطی در logهای مختص هر نوع واقعه، ثبت هرگونه تغییرات سیستمی اعم از زیرساخت، سخت افزار و نرم افزار در فایل log مربوط و همچنین ثبت تمام دستورات و کوثری‌ها در فایل‌های log مربوط
اندازه‌گیری پاسخ	زمان یا دوره زمانی ثبت وقایع زمانی که طول می‌کشد تا یک واقعه پیدا شده و ثبت شود

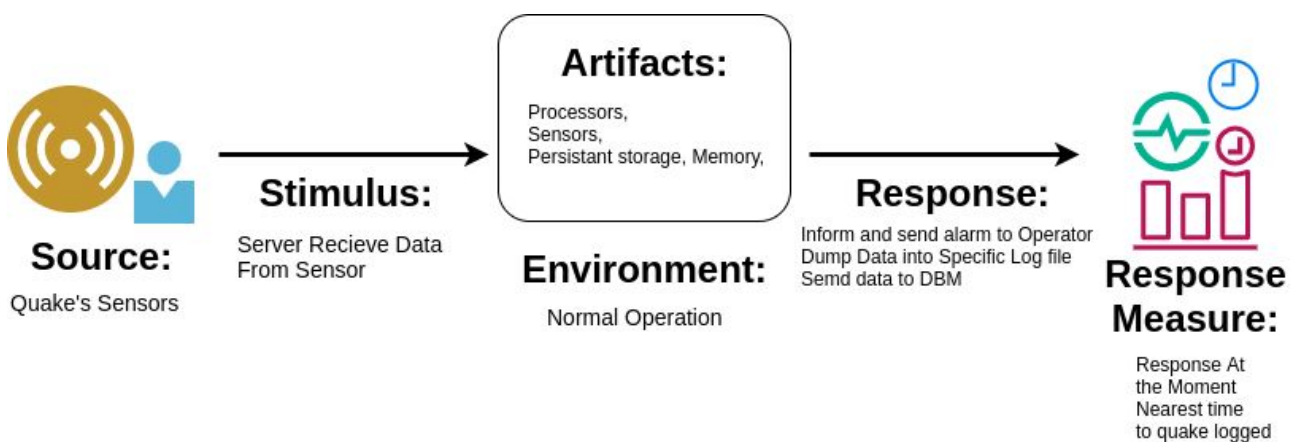
نمودار یا دیاگرام سناریوی عمومی (General Scenario) به شکل زیر می‌شود.



سناریوی معین (Concrete Scenario) مطابق شکل نموداری که در ذیل پس از تعریف یک سناریوی معین می‌آید، خواهد بود.

سناریوی معین برای یک سنسور لرزه نگار کنار یک آتشفشان

یک لرزه توسط سنسور ثبت می‌شود و این ثبت لرزه به سرور فرستاده می‌شود برای بررسی بیشتر. در سناریو ما برای صفت کیفی پرحرف یا verbose هدف بررسی کیفی ثبت و نگهداری است این داده است. سناریو از یک لرزش شروع می‌شود که توسط یک سنسور لرزه‌ای ثبت شده و مهر زمانی می‌خورد و سپس به سرور ارسال می‌شود. به محض رسیدن داده به سرور یک مهر زمانی به متادیتا اضافه شده و اطلاعات و آلارم به اوپراتور فرستاده می‌شود به صورت کامل (پرحرف - Verbose). در همین حین اطلاعات زمانی لاگ در یک فایل ثبت و در نهایت برای پایگاه داده ارسال می‌شود.



پایان

این فایل‌ها به صورت PDF و ODT (فایل نوشتاری قابل ویرایش)، فایل تصاویر و فایل اصلی تصاویر (قابل ویرایش) ارسال شده‌اند. فایل پاسخ‌نامه در ۸ صفحه PDF است.