

بسم الله الرحمن الرحيم

مہر و کریم پور

کارشناسی ارشد نرم افزار

تمرینات درس معماری

سوال اول:

مدل ۱+۴ مدل ۱+۴ توسط فیلیپ کروتچن برای "توصیف معماری سیستم‌های نرم‌افزاری" معرفی شد. این مدل مبتنی بر استفاده از چند view است. viewها برای توصیف سیستم از دید مصرف کنندگان مختلف و سرمایه‌گذاران نرم‌افزار است مانند کاربران نهایی، برنامه‌نویسان نرم‌افزار و مدیران پروژه. view ۴ در این مدل شامل مدل منطقی، توسعه، فرایند و فیزیکی می‌شود. از این‌ها گذشته از use caseها هم برای تشریح معماری استفاده می‌شود.

• **Logical view:** دید منطقی درگیر قابلیت‌هایی است که سیستم برای کاربران نهایی فراهم می‌آورد. برای نمایش دید منطقی از نمودارهای کلاس، Communication، و نمودار Sequence استفاده می‌شود.

• **Development View:** دید توسعه برای تشریح سیستم از دید یک برنامه‌نویس است و درگیر مدیریت نرم‌افزار است. به این View همچنین Implementation View هم می‌گویند. در UML از نمودار Component برای توصیف کمپوننت‌های سیستم استفاده می‌کند. برای نمایش این دید در UML از نمودار Package استفاده می‌کنیم.

• **Process View:** دید فرایند، درگیر وجهه پویای سیستم است. پروسه‌های سیستم را توصیف کرده و نحوه تعامل آن‌ها با هم. همچنین بر رفتار سیستم در زمان اجرا تمرکز دارد. دید فرایند همزمانی (concurrency)، توزیع، کارایی و مقیاس‌پذیری را پوشش می‌دهد. در UML از نمودارهای Activity برای نمایش آن استفاده می‌شود.

• **Physical View:** دید فیزیکی سیستم را از دید یک مهندس سیستم نمایش می‌دهد. این دید درگیر توپولوژی کمپوننت‌های نرم‌افزاری در لایه فیزیکی است، به علاوه ارتباطات فیزیکی بین این کمپوننت‌ها. در UML از نمودارهای Deployment برای نمایش لایه فیزیکی استفاده می‌شود.

• **Use Case:** یا سناریو. توصیف معماری با استفاده از مجموعه‌ای از use caseها، سناریوها انجام می‌شود. این سناریوها توصیفگر توالی ارتباطات بین اشیا و پروسه‌ها هستند.

سؤال دوم:

در مورد تحلیل نیازمندی‌ها را در مطالب قبلی صحبت کردیم. حال برای راحت تر شدن روند تحلیل نیازمندی‌ها به خصوص در مورد پروژه‌ها و سیستم‌های بزرگ، در مورد چک لیستی صحبت می‌کنیم که این روند را تسهیل کرده و به ما کمک می‌کند که چیزی را از قلم نیاندازیم این چک لیست را به اختصار FURPS و در روایت جدیدتر FURPS+ گویند. همان طور که گفتیم این چک لیست فقط یک یادآوری است که به ما می‌گوید آیا تا کنون در مورد این مطلب فکر کرده‌ای؟ آیا به این مطلب از این زاویه نگاه کرده‌ای؟ حال به شرح هر کدام از موارد FURPS می‌پردازیم:

• **Functional Requirements:** همان نیازمندی‌های کارکردی (یا تصریحی) است که در مطلب قبلی بدان اشاره شد.

• **Usability:** مانند مستندات راهنما، مستندات آموزشی.

• **Reliability:** برنامه ریزی برای آنچه که در زمان از کار افتادن سیستم باید انجام دهیم. مثلاً اگر بانک اطلاعاتی از کار بیافتد چه گونه سرویس دهی به کاربران قطع نشود.

• **Performance:** سیستم به چه حجمی از کاربران باید پاسخ دهد. زمان مناسب پاسخی سیستم چقدر باید باشد.

• **Supportability:** پشتیبانی سیستم پس از اجرا به چه نحوی خواهد بود. آیا به سیستم های جانبی دیگری برای پشتیبانی نیاز داریم، به طور مثال آیا برای کمک به کاربران سیستم، یک سیستم FAQ طراحی شده است؟

سؤال سوم:

exception (استثنا) یک رویداد ناخواسته است که کارکرد طبیعی برنامه را مختل می کند زمانی که Exception اتفاق می افتد اجرای برنامه متوقف می شود در چنین مواردی سیستم یک خطا به ما نمایش خواهد داد. یک چیزی خوبی که در مورد exceptions (استثناها) وجود دارد این است که در جاوا می توان آنها را handle (رسیدگی) کرد با handle کردن Exceptions (استثناها) امکان نمایش یک پیام معنی دار به کاربر انتهایی وجود دارد زیر پیام هایی که توسط سیستم در مورد Exceptions نمایش داده می شود برای کاربران انتهایی غنی قابل درک است.

یکی از مزایای exception handling, از بسته شدن برنامه هنگام بروز exception یا استثنا جلوگیری می کند به طور مثال فکر کنید یک محاسبه پیچیده توسط برنامه شما انجام شده است و کاربر در انتها به اشتباه عدد را به صفر تقسیم می کند و برنامه کرش کرده و به بیرون می آید اگر به درستی این exception رسیدگی شده باشد کاربر دیگر لازم نیست تمامی عملیات را اول انتها دوباره انجام دهد

سؤال چهارم:

در پروسه پردازش، پایگاه داده ها و شبکه های کامپیوتری، پروتکل دو مرحله ای (CP۲) یک نوع پروتکل تعهد اتمی است (PCA). این یک الگوریتم توزیع شده است که همه فرآیندهای را که در یک معامله اتمی توزیع شده شرکت می کنند، هماهنگ می کند که آیا مرتکب یا لغو (معکوس کردن) معامله (نوع خاصی از پروتکل اجماع) باشد.

مرحله فاعل درخواست (یا مرحله رأی گیری)، که در آن یک فرایند هماهنگ کننده تلاش می کند تا تمامی فرآیند مشارکت در معامله (به نام شرکت کنندگان، گروه های کاری یا کارگران) را برای انجام اقدامات لازم برای انجام یا لغو معامله و رأی دادن آماده کند "بله": مرتکب شده (اگر بخشی از بخش محلی شرکت کننده معامله به درستی به پایان برسد)، یا "نه": قطع (اگر مشکلی با بخش محلی شناسایی شده باشد)، و

فاکتور تعهد، که در آن، بر اساس رأی دادن از شرکت کنندگان، هماهنگ کننده تصمیم می گیرد که آیا مرتکب شود (تنها اگر همه رأی داده اند "بله") یا تراکنش را لغو کنند (در غیر این صورت) و نتیجه را به همه شرکت کنندگان اعلام می کند. سپس شرکت کنندگان با اقدامات مورد نیاز خود (مرتکب یا قطع) با منابع محلی معاملات خود (همچنین منابع قابل باز یافتی، مانند داده پایگاه داده) و بخش مربوطه در خروجی دیگر معامله (در صورت لزوم) دنبال می شود.

در پایگاه داده ها و پردازش تراکنش، قفل دو مرحله ای (LP۲) یک روش کنترل همزمان است که سریالی را تضمین می کند. همچنین نام مجموعه نتایج حاصل از برنامه های معامله پایگاه داده (تاریخچه) است. پروتکل از قفل استفاده می کند، توسط یک معامله به داده ها اعمال می شود، که ممکن است از سایر معامله ها از دسترسی به داده های مشابه در طول عمر معامله جلوگیری کند (تفسیر به عنوان سیگنال برای توقف).

با پروتکل LP۲، قفل ها در دو مرحله اعمال می شوند و حذف می شوند:

فاز گسترش: قفل ها به دست می آیند و بدون قفل آزاد می شوند.

مرحله فشرده سازی: قفل ها آزاد می شوند و هیچ قفل نمی شود.

سؤال پنجم:

Spring در برنامه نویسی تجاری جاوا معروف ترین فریمورک است و میلیون ها برنامه نویس با آن کار می کنند دلایل استفاده این همه برنامه نویس از این فریمورک performance بسیار بالا و تست راحت و معماری آن است. فریمورک spring یک پلتفرم اوپن سورس جاوا است که توسط Rod Johnson نوشته شده است. با استفاده از ویژگی های spring می توان هر برنامه جاوایی را نوشت. با این حال قابلیت هایی را برا نوشتن برنامه های تحت وب تحت پلت فرم Java EE دارد. هدف فریمورک spring این است که برنامه نویسی J۲EE را آسان تر کند و با استفاده از مدل برنامه نویسی POJO، برنامه نویسی را گسترش دهد. قبل از این که به ادامه بحث در مورد spring پردازیم دو کلمه J۲EE و POJO را معرفی می کنیم.

سؤال ششم:

در برنامه نویسی شی گرا، الگوی تزئین کننده یک الگوی طراحی است که اجازه می دهد رفتار به یک شی فرد به صورت پویا بدون تاثیر بر رفتار دیگر اشیاء از یک کلاس باشد. الگوی تزئین کننده اغلب برای پیروی از اصل مسئولیت واحد، مفید است، زیرا که اجازه می دهد که عملکرد بین کلاس ها با نگرانی های منحصر به فرد تقسیم شود. الگوی تزئین ساختار تقریباً یکسان است با زنجیره ای از الگوی مسئولیت، تفاوت این است که در یک زنجیره مسئولیت، دقیقاً یکی از کلاس ها درخواست را اداره می کند، در حالی که برای تزئین، تمام کلاس ها درخواست را اداره می کنند.

سؤال هفتم:

PAOS پروتکل دسترسی آسان به اشیاء و مخفف عبارت locotorP sseccA tcejbO elpmiS می باشد

این پروتکل برای رد و بدل کردن اطلاعات بین برنامه ها استفاده می شود. اطلاعات در SOAP به صورت پیام (Message) و از طریق پروتکل های موجود در اینترنت مانند HTTP منتقل می شود. به زبان ساده تر، SOAP یک پروتکل برای دستیابی به یک سرویس ارایه شده در وب (Web Service) می باشد. پروتکل SOAP یکی از عمومی ترین استاندارد هایی است که در وب سرویس ها استفاده می شود.

وقتی یک برنامه شروع به ارتباط با وب سرویس می کند، پیغام های SOAP وسیله ای برای ارتباط و انتقال دیتایین آن دو هستند. یک پیغام SOAP به وب سرویس فرستاده می شود و یک تابع را در آن به اجرا در می آورد، وب سرویس نیز از محتوای پیغام SOAP استفاده کرده و عملیات خود را آغاز می کند و در انتها نیز نتایج را با یک پیغام SOAP دیگر به برنامه اصلی می فرستد.

TSER که مخفف refsnarT etatS lanoitatneserpeR میباشد یک معماری وب سرویس است که از PTTH برای انتقال اطلاعات میان کلاینت و سرور استفاده میکند کار کردن با TSER بسیار ساده تر از وب سرویس های پیچیده ای مانند PAOS میباشد. و تمام افعال استاندارد این پروتکل را که توسط مرورگرهای وب قابل HTTP عموماً بر روی پروتکل RESTful یک سرویس به اصطلاح (GET, POST, PUT, DELETE) درک هستند کار میکند مانند

TSER یک مدل معماری برای طراحی برنامه های کاربردی شبکه است که ترکیبی از چند مدل معماری مبتنی بر شبکه می باشد و محدودیت هایی جهت تعریف یک واسط اتصال یکنواخت برای آن در نظر گرفته شده است. TSER مخفف عبارت refsnarT etatS lanoitatneserpeR است و متکی بر یک پروتکل ارتباطی بدون حالت، کلاینت سرور و با قابلیت ehcac کردن می باشد که در اکثر موارد پروتکل PTTH مورد استفاده قرار می گیرد. ایده اصلی معماری TSER این است که به جای استفاده از مکانیزم های پیچیده ای مانند ABROC، CPR یا PAOS برای اتصال ماشین ها از PTTH ساده برای برقراری ارتباط بین ماشین ها استفاده شود. مدل TSER شش قید برای معماری برنامه های شبکه تعریف می کند: