

Resumo: Perguntas a fazer ao elaborar um projeto arquitetural

Sobre entidades externas ao sistema

- ✎ Quais sistemas externos devem ser acessados?
- ✎ Como serão acessados?
- ✎ Há integração com o legado a ser feita? Como?

Determinação de oportunidades para o reuso de software

- ✎ Há interesse/conveniência/tempo em aproveitar tais oportunidades?
- ✎ Como isso pode ser feito?
 - ✎ Com componentes?
 - ✎ Construindo um framework?

Sobre a organização geral do sistema

- ✎ O sistema é centralizado ou distribuído?
- ✎ Como modularizar em subsistemas?
- ✎ Como minimizar acoplamento entre os pedaços?
- ✎ Lembrando que um sistema pode ser visto como sendo composto de três grandes camadas lógicas ...
 - ✎ Tais camadas serão lógicas ou terão existência física separada?
 - ✎ Onde colocar o business logic (como dividir entre as camadas)?
 - ✎ Qual é o nível de acoplamento, frequência de interações, volumes de dados trocados entre as camadas?
- ✎ Qual será a estrutura de comunicação e controle entre os subsistemas (como ligar as camadas e partições)?
 - ✎ Usando o Observer Pattern?
 - ✎ Usando o Model-View-Controller Pattern?
 - ✎ Que subsistema pode se comunicar com que outros?

- ✎ Quais são as interfaces importantes entre os pedaços?
 - ✎ Qual é o formato das mensagens (xml, ...)?
 - ✎ Como é feita a comunicação remota? CORBA? RPC? RMI? Message Queue?
- ✎ A programação será feita com qual paradigma? OO?
- ✎ Que linguagens e ferramentas serão usadas?
- ✎ Que frameworks serão usados?
 - ✎ struts? Tapestry? Java Server Faces?
- ✎ Como será feito a alocação de memória para os elementos do programa?
- ✎ Onde são armazenados os strings?
- ✎ Há considerações especiais de segurança que afetam o sistema?
- ✎ Como é a estratégia de tratamento de erros?

Sobre a camada de interface

- ✎ O sistema será acessado usando que tipos de clientes?
 - ✎ Browser?
 - ✎ Uso de applet?
 - ✎ Uso de script cliente?
- ✎ Como fazer a interface gráfica?
 - ✎ Com que ferramentas?
 - ✎ Com que componentes visuais?
 - ✎ Serão desenvolvidos? comprados?
 - ✎ Javascript ou outra linguagem de script do lado cliente será usada?
- ✎ Que modelo de componentes visuais será usado?
 - ✎ ActiveX?
 - ✎ Javabeans?
- ✎ Se a interface usar um browser, como será feita a geração de HTML dinâmico?
 - ✎ Servlets?
 - ✎ JSP?
 - ✎ ASP?
- ✎ Que ferramentas usar para a formatação de

relatórios?

- ✎ Há packages a desenvolver que poderiam ser comuns a várias partes da interface?
- ✎ Há considerações de localização (para outras línguas ou outros países)?

Sobre a camada de lógica da aplicação

- ✎ Escolha de middleware: qual modelo de componentes de servidor será usado?
 - ✎ COM+/.Net?
 - ✎ EJB?
- ✎ Quais são os componentes principais a fazer?
- ✎ Serão componentes persistentes?
- ✎ Serão componentes com ou sem estado?
- ✎ Quais são os algoritmos não triviais?
- ✎ Há uso de threads?
 - ✎ Como tratar concorrência? reentrância?
- ✎ De que forma atender aos requisitos para uso multiusuário?
 - ✎ Soluções para resolver a concorrência
- ✎ Que APIs serão usadas para acesso a:
 - ✎ Dados persistentes?
 - ✎ Serviço de mail?
 - ✎ Serviço de nomes?
- ✎ Há packages a desenvolver que poderiam ser comuns a várias partes da lógica da aplicação?
- ✎ Qual é a organização interna dos módulos executáveis?
 - ✎ Determinar sub-camadas e partições
- ✎ Quais são as interfaces importantes entre os pedaços?
- ✎ Onde verificar os business rules?
 - ✎ No SGBD?
 - ✎ No middleware?
- ✎ Como implementar aspectos de segurança?
- ✎ Como implementar os requisitos de auditoria?

Sobre a camada de dados persistentes

- ✎ Quais são as fontes de dados? externas? internas? existentes? novas?
- ✎ Como acessá-las?
- ✎ Que estratégia de persistência será usada:
 - ✎ Na memória (com recursos especiais como bateria, memória flash)?
 - ✎ Em arquivos?
 - ✎ Usando um SGBD?
- ✎ Qual paradigma de SGBD usar?
 - ✎ Relacional?
 - ✎ O-R?
 - ✎ O-O?
- ✎ Onde usar stored procedures para implementar os business rules ou garantir a integridade dos dados?
- ✎ Qual será a estratégia de interação entre a aplicação e os dados?
- ✎ De que forma atender aos requisitos para uso multiusuário (concorrência)?
- ✎ Como resolver o impedance mismatch entre a camada de lógica de aplicação e o esquema de persistência?
- ✎ Para grandes escalas, como oferecer connection pooling?
- ✎ Como implementar a segurança no SGBD?
- ✎ Como será feita a população dos bancos de dados?

Sobre requisitos de desempenho

- ✎ Há necessidade de bolar formas especiais de atender aos requisitos de desempenho?
 - ✎ Como prover escala?
 - ✎ Como prover failover?

O que deve ser produzido?

- ✎ Quais são os módulos executáveis a produzir?
 - ✎ Executáveis, DLLs, ...
- ✎ Quais são os arquivos importantes (de configuração,

etc.) e seu formato (xml, ...)?

- ✎ Como será resolvida a instalação do produto?
- ✎ O que será comprado, feito, modificado?
- ✎ O que será gerado automaticamente com uma ferramenta especial?
 - ✎ Exemplo: geração de esquema através de ferramentas CASE, ...

Sobre a integração futura

- ✎ Que interfaces devem ser expostas para facilitar a futura integração de aplicações (Enterprise Application Integration - EAI)?
 - ✎ Pode-se usar uma camada de API para expor a camada de business logic, colocar uma camada de script acima disso e ainda camada(s) de interface para ativar a business logic através de scripts.
 - ✎ Vantagens:
 - ✎ Fácil criação de macros e outras formas de automação
 - ✎ Fácil criação de testes de regressão
 - ✎ Clara separação de business logic da interface para o usuário
- ✎ Como será feita a exposição?
 - ✎ Com XML?
 - ✎ Através de uma API?

Perguntas adicionais

- ✎ Que ferramentas de desenvolvimento serão geradas?
- ✎ Como será o atendimento a outros requisitos (custo, mobilidade, uso de padrões, etc.)
- ✎ Há considerações especiais de administração a levar em conta?
 - ✎ Como será a troca de versões?
 - ✎ Como será a distribuição do software?
 - ✎ Via Internet/Intranet?
 - ✎ Via mídia?

- ✍ Há considerações de portabilidade?

Pergunta final

- ✍ Você já verificou que a arquitetura bolada pode atender a todos os requisitos levantados?
 - ✍ Isso é feito montando um "traceability matrix"

programa