**Correção trabalho 3 de Análise e Projeto de Sistemas**

**Thiago Souza Nobrega e Grupo (8,5).**

**Configuração WebDep (2 de 5)**:

Problemas nas chamadas dos métodos nos diagramas de sequência. Classes chamando elas mesmas onde isto não ocorre. (OK)

No diagrama de estados os estados estão descritos como se fossem atividades: Testar deveria ser testando... O diagrama não ficou bom, não representa realmente o que está sendo realizado na tela. (OK)

Diagramas de classes: Algumas associações devem ser somente dependências. Instanciar uma classe ou chamar um método não cria uma associação, mas sim uma dependência. Associação é somente se houver atributos entre as classes. A navegabilidade também está com problema. Não é só fazer engenharia reversa e para fazer o diagrama de classes. Onde estão as multiplicidades? (OK)

**Relatório de Acesso as Falhas (3 de 5):**

O diagrama de estados ficou bom.

Os diagramas de classes contêm os mesmos problemas citados na seção anterior. (OK)

Problemas com as linhas de vida do diagrama de sequência. (Não ficou bom o diagrama, muitos problemas) (OK com uma força)

O Protótipo ficou bom, mas faltou entregar o código-fonte.

**Renato Rodrigues e Grupo (9,0)**

**Importar Log (4 de 5)**:

Nos diagramas de sequência faltaram os atores, e JSP caso existam. Quem chama o command? (OK)

No diagrama de classes a classe ImportarLog é um JSP? Se for ela não tem doGet e doPost

No diagrama de estados ficou bom.

**Relatório de Sumário de Códigos HTTP (4 de 5):**

No diagrama de sequência as classes do tipo Command e FrontControllerServlet estão classificadas com o estereótipo control, mas na verdade elas representam boundary (view).

No diagrama de sequência e de classes não está sendo utilizado o DAO para acesso aos dados.

O diagrama de estados ficou bom.

O protótipo ficou bom.

**Gabriel Eduardo e Grupo (10)**

**Importar / Excluir Logs Periodicamente (4, 5)**

Diagrama de classes. Problema com as relações entre as classes. São todas associações? Se são, onde estão as multiplicidades e direcionalidade. Algumas associações provavelmente são de pendências. (Ok)

Diagrama de Sequência. No insert logs tem alguns problemas nas linhas de vida. (OK)

O diagrama de estados ficou bom.

**Relatório de Defeitos (4,5 de 5)**

O diagrama de sequência ficou bom.

O diagrama de estados não contém estados para as validações dos parâmetros do relatório. Precisa de maior detalhamento. (OK)

O diagrama de classes ficou bom.

**Caio Potter e Grupo (8,5)**

**Relatório de Acesso HTTP (2 de 5)**

O diagrama de classes e o de sequência não coincidem. O de classes está fantasioso, não corresponde a arquitetura do sistema nem de longe (ver documento de arquitetura). (OK)

No diagrama de classes as classes de modelo do sistema não aparecem, somente uma chamada RegistroAcessoRelatórioHTTP, que na minha visão não deveria ser um modelo.

O desafio nesta funcionalidade era incluir os gráficos, que apesar de termos no documento de arquitetura uma forma de fazer isto, ela não foi utilizada.

O diagrama de estados mistura ações nos estados e nas transições, o que não é recomendado. Senti falta de estados e transições que precisariam ser modeladas para mostrar a mudança das abas, etc. (ok)

**Relatório de Scatter Plot de falhas no tempo (3 de 5)**

Diagrama de sequência, somente não entendi o exibirgráfico.jsp, o gráfico não vai ser exibido no mesmo jsp dos filtros? (OK)

No diagrama de classes a arquitetura de tags para exibição dos gráficos aparecem, mas não consta isto no diagrama de sequência. Os diagramas devem ser consistentes ente si.

O diagrama de classes só mostra a parte dos gráficos. E todo o resto da funcionalidade, onde está? (OK)

O diagrama de estados tem um problema. Quando fazemos um fork de uma transição de estados representamos que o sistema está nos dois estados que saem do Fork. O que no seu diagrama não é consistente, pois não se pode ter a tabela e o gráfico selecionados ao mesmo tempo. (OK– adicionar um estado de validação e possibilidade de voltar para filtrar novamente)

**Luis Enrique e Grupo (8,0)**

**Relatório Perfis de Acesso (2 de 5):**

Diagrama de estados. Os estados estão representando ações e não estados efetivamente. (OK)

Diagrama de Sequência. Problemas nas linhas de vida. Não está utilizando o DAO. (OK parcial – ainda faltou o entity na parte de TAG)

Diagrama de Classes. Problemas nas associações que deveriam ser dependências, não há multiplicidades. (OK)

Não há nenhuma referência aos gráficos que foram solicitados. (OK)

**Manter Lista de E-mail de Notificação de Falhas (CRUD) (2 de 5):**

Diagrama de sequência. Está muito longe do que deveria ser. Não contempla as diversas classes na arquitetura. (OK)

Os protótipos ficaram bons.

O diagrama de estados com estados que representam atividades no lugar de estados. (Ok parcial – mostra pelo menos a mudança da tela)

Diagrama de classes: Falta o DAO, as multiplicidades e problemas nas relações de associação que deveriam ser dependências. (OK)

**Thiago Martins e Grupo (8,0)**

**Relatório de Scatter Plot da relação entre quantidade de acesso e de falhas (1,5 de 5):**

Diagrama de sequência. Faltam algumas coisas, não tem DAO por exemplo. Problemas nas linhas de vida. (Ok)

Diagrama de classes. Também faltam algumas coisas. Não tem command, dao ou outros elementos da arquitetura como o frontController. (Ok)

Diagrama de estados modelado como se fosse de atividades. (Ok com uma ajudinha)

**Configurar periodicidade de leitura (1,5 de 5):**

Mesmos problemas da seção anterior.

**Artur Avner (8,0)**

Não era para modelar somente o envio do e-mail, mas também detectar quando enviar o e-mail. Ver na especificação. (Não ok)

A única classe de modelo é Email? Não precisaria ler os registros de loc? (OK)

Diagrama de e4stados: Importar Log é ação, não estado. (OK)