INSTITUTO METODISTA GRANBERY

Relatório de atividade de qualidade de software

Jean Vidal Soares

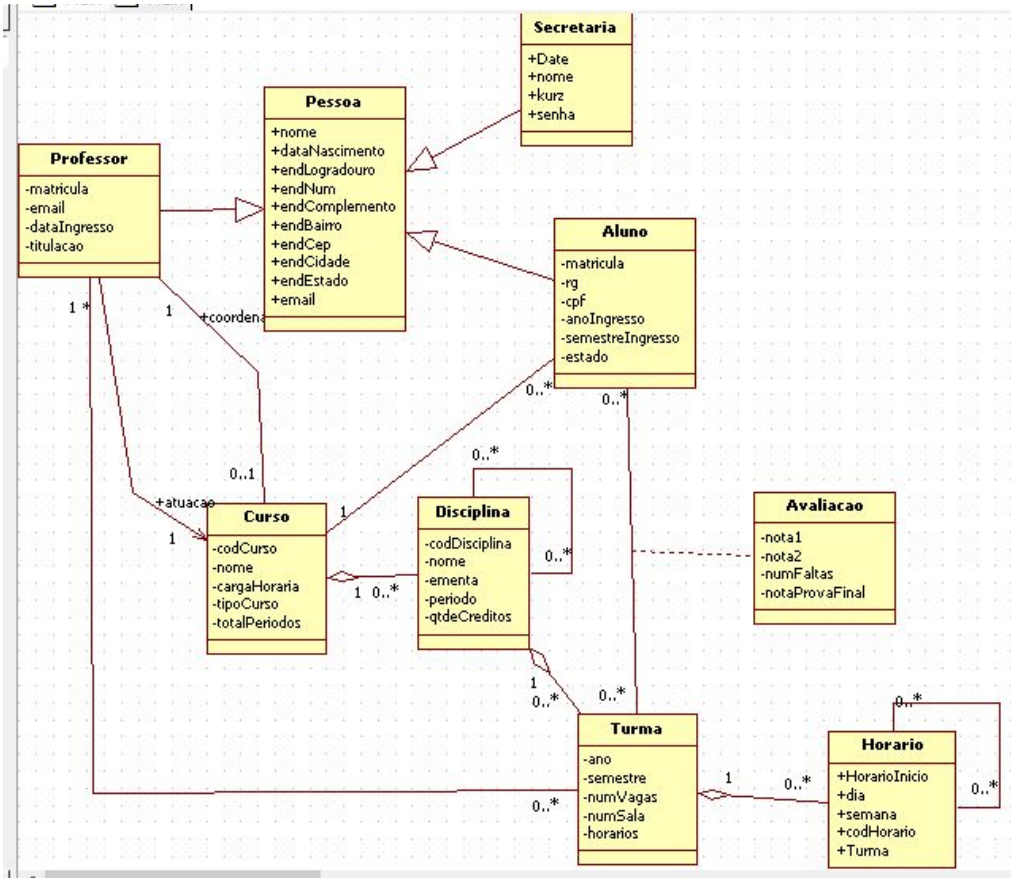
Gabriel Aparecido de Oliveira

Juiz de Fora

15/05/2018

1. DIAGRAMA.

Em relação ao diagrama apresentado e o software, foi apresentado incoerência por conta da tabela Horário, Secretaría, Pessoa que não fora apresentado inicialmente mas que faz parte do sistema.



2. REQUISITO FUNCIONAL

2.1. COPIA DO REQUISITO FUNCIONAL APRESENTADO

Requisito Funcional 1:

O sistema deverá permitir ao usuário efetuar login e liberar acesso ao sistema de acordo com o vínculo com a instituição, com login e senha individuais.

Requisito Funcional 2:

O sistema deve permitir à secretaria pesquisar cursos, oferecendo as opções de excluir, alterar e consultar. A interface deve mostrar o código do curso e o nome do curso e uma opção para incluir novo curso.

Requisito Funcional 3:

O sistema deve permitir à secretaria manter cursos contendo Código do Curso, Nome do curso, Professor Coordenador, Carga horária, Tipo de curso (técnico concomitante, técnico integrado, superior, especialização, mestrado, doutorado), Total de períodos.

Requisito Funcional 4:

O sistema deve permitir à secretaria pesquisar disciplinas através do nome e filtrando por curso, oferecendo as opções de excluir, alterar e consultar. A interface deve mostrar o código da disciplina e o nome da disciplina e uma opção para incluir nova disciplina.

Requisito Funcional 5:

O sistema deve permitir à secretaria manter disciplinas, contendo código da disciplina, nome da disciplina, curso, ementa, período, quantidade de créditos, pré-requisitos.

Requisito Funcional 6:

O sistema deve permitir à secretaria pesquisar professores através do nome do professor, oferecendo as opções de excluir, alterar e consultar. A interface deve mostrar a matrícula do professor e o nome do professor e uma opção para incluir novo professor.

Requisito Funcional 7:

O sistema deve permitir à secretaria manter professores, contendo matricula, nome, data de nascimento, endereço (Logradouro, Número, Complemento, Bairro, CEP, Cidade, Estado), e-mail, data de ingresso, titulação, curso principal de atuação.

Requisito Funcional 8:

O sistema deve permitir à secretaria pesquisar turmas através do nome da disciplina e filtrando por ano, semestre e curso, oferecendo as opções de excluir, alterar e consultar. A interface deve mostrar o código da turma, nome da disciplina e nome do professor e uma opção para incluir nova turma.

Requisito Funcional 9:

O sistema deve permitir à secretaria manter turmas contendo código da disciplina, ano e semestre em que será lecionada, curso, nome do professor, número de vagas, sala, horários.

Requisito Funcional 10:

O sistema deve permitir à secretaria pesquisar alunos através do nome do aluno, oferecendo as opções de excluir, alterar e consultar. A interface deve mostrar a matrícula do aluno, nome do aluno e uma opção para incluir novo aluno.

Requisito Funcional 11:

O sistema deve permitir à secretaria manter alunos contendo Matricula (gerada pelo sistema), Nome, Data de Nascimento,

Endereço (Logradouro, Número, Complemento, Bairro, CEP, Cidade, Estado), RG, CPF, Email, Curso, ano/semestre de ingresso e estado (inicialmente matriculado) e posição no vestibular.

Requisito Funcional 12:

O sistema gera a matricula de cada aluno baseado no ano de ingresso do mesmo, código do curso e posição no ingresso, respectivamente. Exemplo: Aluno ingressou no ano 2012, pertence à turma de código 05 e sua posição no ingresso igual a 01, então sua matricula será 120501.

Requisito Funcional 13:

O sistema deve permitir à secretaria modificar estados de alunos para formado, trancado ou matriculado.

Requisito Funcional 14:

O sistema deve permitir ao aluno efetuar matricula em turmas referentes ao seu curso, caso ele tenha os pré-requisitos necessários, não haja coincidência de horários e esteja no estado de matriculado.

Requisito Funcional 15: (Verificar junto a tela)

O sistema deve permitir ao professor pesquisar turma através do ano opções de excluir, alterar e consultar.

A interface deve mostrar a matrícula do professor e o nome do professor e uma opção para incluir novo professor.

Requisito Funcional 16:

O sistema deve permitir ao professor manter notas e frequências para os alunos de suas turmas. São lançadas duas notas por ano/semestre e frequência total, além de nota da prova final para os alunos que tenham obtido média entre 40 e 60.

Requisito Funcional 17:

O sistema deve permitir ao professor calcular o resultado final do aluno considerando, para aprovação, frequência maior ou igual a 75%, média das duas notas maior ou igual a 60. Reprovação é atribuída ao aluno em caso de frequência menor que 75% ou média das duas notas menor que 40. No caso de média das duas notas entre 40 (inclusive) e 60 (exclusive), o aluno deve fazer prova final, sendo aprovado caso obtenha nota maior ou igual a 60.

Requisito Funcional 18:

O sistema deve permitir ao aluno pesquisar nota e frequência, exibindo todas as turmas ou filtrando por ano e semestre. A interface deve mostrar o código da disciplina o período, o nome da disciplina, a frequência do aluno as notas do aluno (nota 1, nota 2, prova final e media final) e o resultado.

Requisito Funcional 19:

O sistema deve permitir aos alunos consultarem os professores vinculados a cada curso e disciplina, filtrando por curso, ano e semestre. A consulta deve retornar, código da disciplina, período, disciplina e professor.

Requisito Funcional 20:

O sistema deve permitir ao aluno logado consultar a matriz curricular de seu curso por ano/semestre ou período assim como todas ao mesmo tempo, cada disciplina deverá ter um link para consultar a ementa.

.

Requisito Funcional 21:

O sistema deverá permitir aos alunos visualizarem as ementas de cada disciplina.

2.2. ANOMALIAS DO REQUISITO FUNCIONAIS.

Em analise aos requisitos funcionais , além das inconsistência com o software inicial devido não haver funcionalidade presente no documento, não foi identificado incoerência, duplo sentido, dados que possam contradizer e outros problemas.

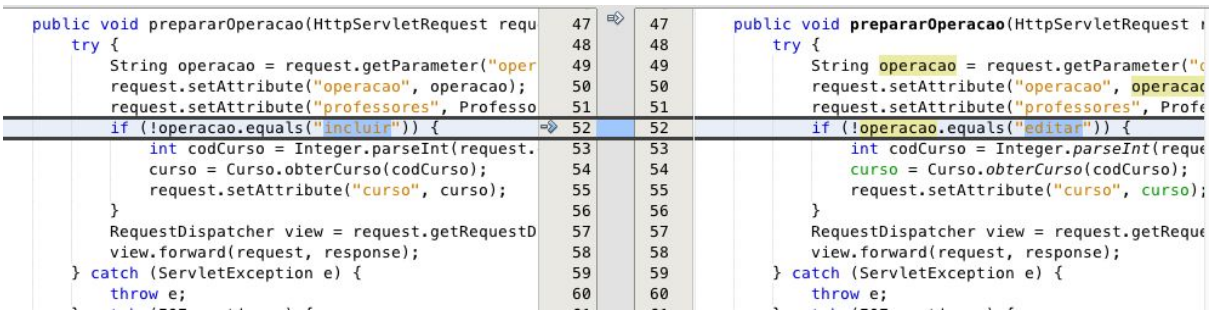
3. Protótipo

- Na imagem pesquisaCurso.ep está vazia
- Na imagem manterTurma.ep possui campos sem descrição
- Na imagem pesquisaAluno.ep possui campos sem descrição
- Na imagemManter.ep aluno falta o campo Status

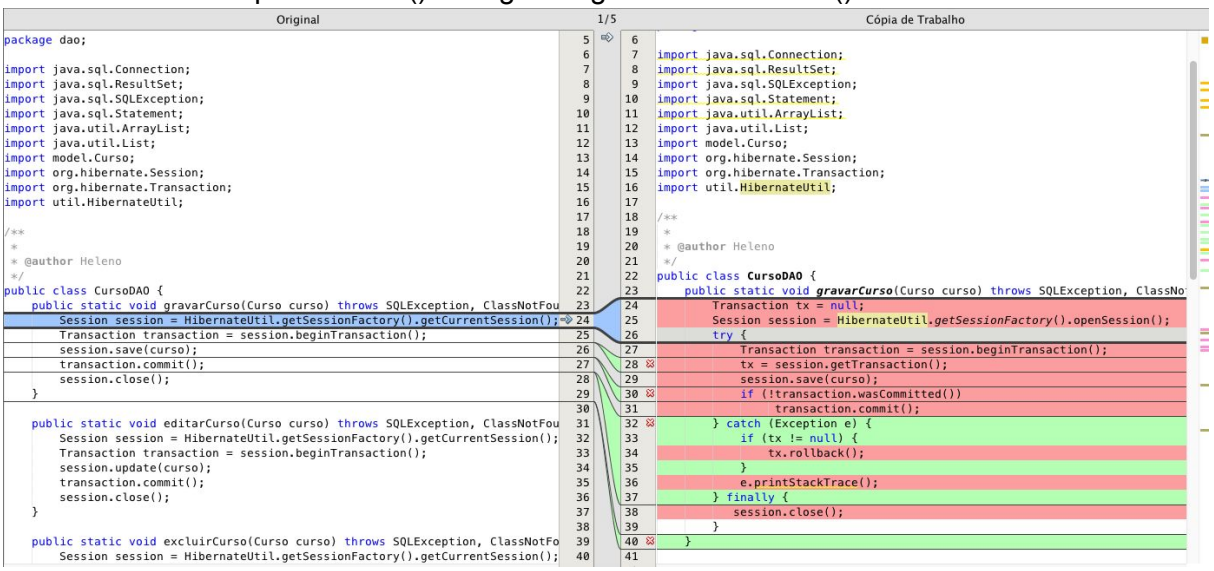
4. Relatório de manutenção

No momento que foi dado a atividade, fui procurando cada parte do programa que precisava de manutenções rápidas para o primeiro funcionamento do mesmo. Com isto, comecei importando as bibliotecas que era necessárias para cada processo. Após importar cada biblioteca, comecei a configurar todo o banco para que possa ter um processamento melhor para a verificação do processo. Nisto percebi que, no arquivo que foi disponibilizado para nós tinha 2 arquivos com scripts para o banco de dados, podendo dar para nós uma desconfiança em qual utilizar, mas aprofundando melhor o projeto, pude verificar que no arquivo hibernate.reveng tem os nomes de cada coluna, dando me um direcionamento melhor para continuar no processo.

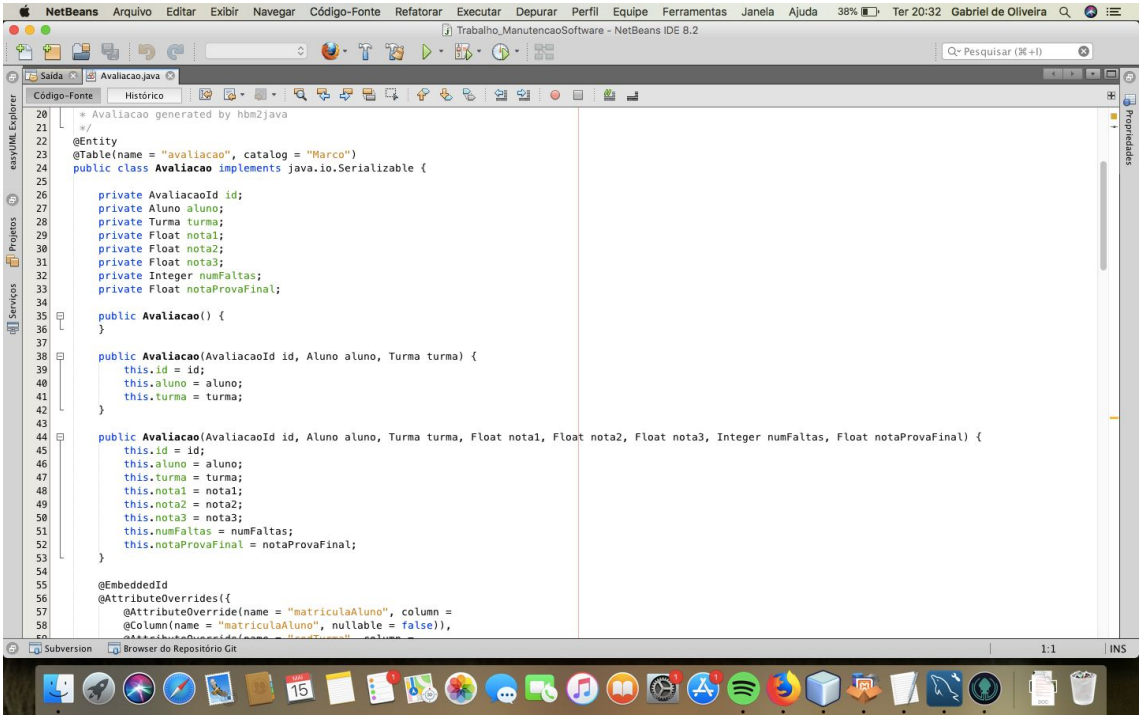
Percebi que no método prepararOperacao tinha uma condição de incluir, sendo que no modo teria que ser editar:



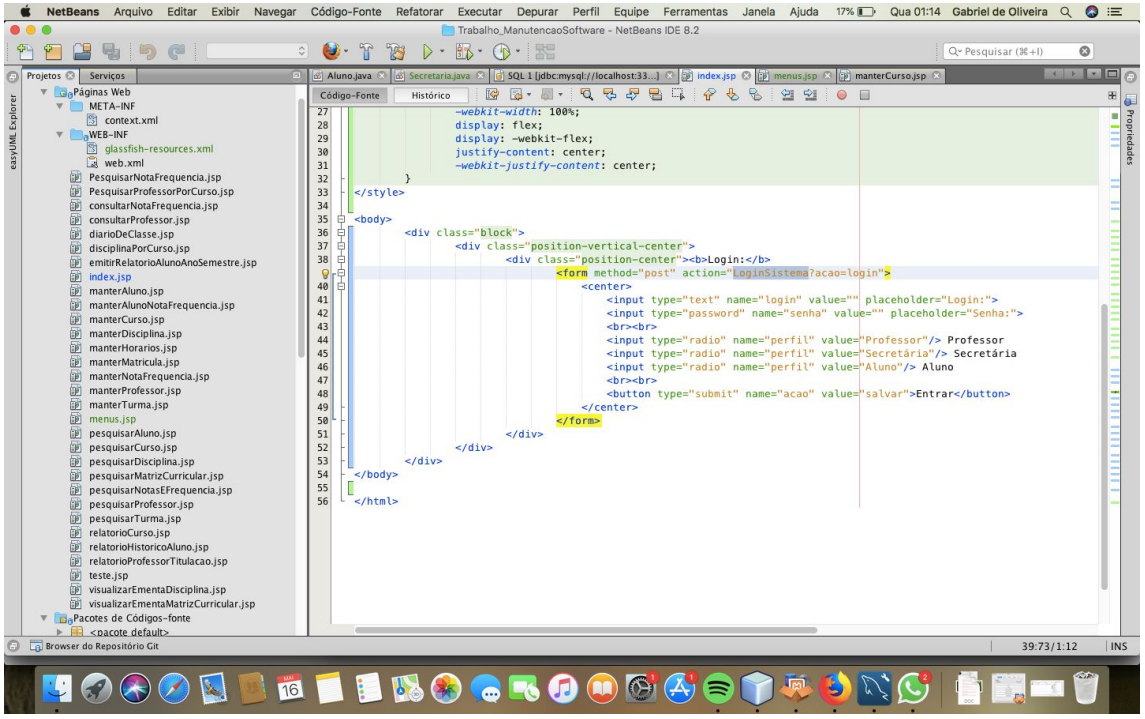
Após as modificações feitas, percebemos que quando inserimos um dado, estava dando erro 500. Analisando e pesquisando sobre o erro, decidimos em trocar as formas dos métodos de inserir, editar e excluir os dados na forma de que, colocando openSession() no lugar de getCurrentSession():



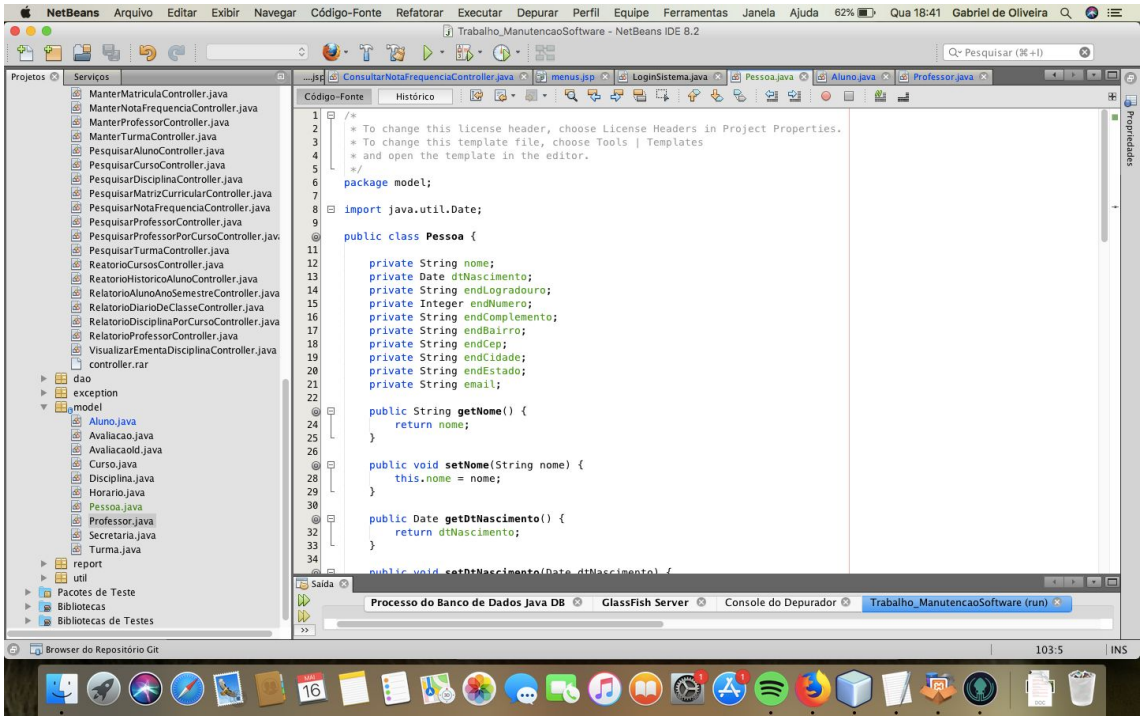
Ocorrendo a transição conforme o esperado, foi aplicado em todos os arquivos pertencentes a pasta DAO.



Foi inserido mais um atributo tanto na classe Avaliacao, tanto na tabela Avaliacao. O atributo foi nota3.



Foi criado uma tela de login como pedido no arquivo de requisitos disponibilizado para nós.



Criado classe Pessoa pois percebemos que, precisamos as classes Professor e Secretaria poderia ter herança de atributos onde seria usado para os funcionamento.