Atividade de Aula – Trabalho Prático

Disciplina	ASW – Arquitetura de Sistemas de Informação Web

Nome do aluno	Gleydson Rocha
Nome do aluno	Leonardo Müller
Nome do aluno	Arthur Luís Viñas dos Santos
Nome do aluno	Valéria Queiroz Alvarenga
Nome do aluno	Maykon Luís Capellari
Nome do aluno	Sinaide José dos Santos Júnior

Objetivos

Exercitar os seguintes conceitos vistos em sala de aula:

- ✓ Aplicar os conceitos sobre dado, informação e conhecimento.
- ✓ Analisar os conceitos sobre os padrões de interoperabilidade.
- ✓ Aplicar conceitos na prática sobre arquitetura: MVC, REST e SOA.
- ✓ Aplicar os conceitos na prática de TDD e BDD.
- ✓ Praticar o trabalho em equipe, utilizando as tecnologias, ferramentas e metodologias ágeis.

Enunciado

O seu time foi contratado para desenvolver um módulo web que calcula o aumento salarial de uma empresa. Este sistema precisa atender aos seguintes requisitos:

1 – Recuperar todos os funcionários do banco de dados (Tabela Pessoa) e calcular o aumento salarial.
2 – Somente funcionários Ativos (Coluna Pessoa.Ativo = “S”) poderão ter seu salário aumentado.
3 – Somente funcionários com salários entre R$1000,00 a R$5000,00 deverá receber aumento salarial.
4 – Para todos os funcionários que atendam as regras 2 e 3 deverá ser aplicado um fator base de aumento de 10%.
5 – Para todos os funcionários que atendam a regra 4 e para cada ano trabalhado na empresa, o funcionário deverá receber um adicional de + 1% limitado ao máximo de 10%.
O cálculo de anos trabalhados pode ser feito usando a coluna IniciodeContrato da tabela Pessoa. Exemplo: ((DateTime.Now - row.Field<DateTime>("INICIOCONTRATO")).Days / 365).
6 – Depois que o sistema calcular o aumento dos funcionários, deverá exibir uma tela com os novos salários calculados.
7 – Criar uma funcionalidade para deletar do banco de dados os funcionários inativos e retornar a quantidade deletada para a página web.

- O trabalho está sendo feito em grupo com objetivo de praticarem as metodologias ágeis e verificarem como BDD e TDD podem ajudar na agilidade. Antes de começarem a executar as atividades se reúnam em equipes, leiam as atividades, identifiquem dentro da equipe as pessoas que são mais indicadas para executar cada atividade, dividam as atividades e executem sempre comunicando entre vocês. Ao final repassem todas as atividades com a equipe, vejam se está tudo correto, façam os últimos ajustes e envie o trabalho final.
- Este sistema web é o mesmo feito na atividade prévia, portanto, os itens 1, 2, 3, 4 e 6 estão implementados.
- Estará implementado também os testes unitários que cobrem o cálculo do aumento do funcionário usando a tecnologia do MSTest.
- Abra no Visual Studio o projeto AulaPratica.sln e certifique que esteja sendo executado com sucesso e o sistema funcionando corretamente para começar as atividades.

Atividades

O grupo deverá desempenhar as seguintes atividades:

1. De acordo com a aplicação, identifique um tipo de usuário para o sistema e liste o que pode ser considerado dado e informação para este usuário. (Vale 10% do total de pontos).

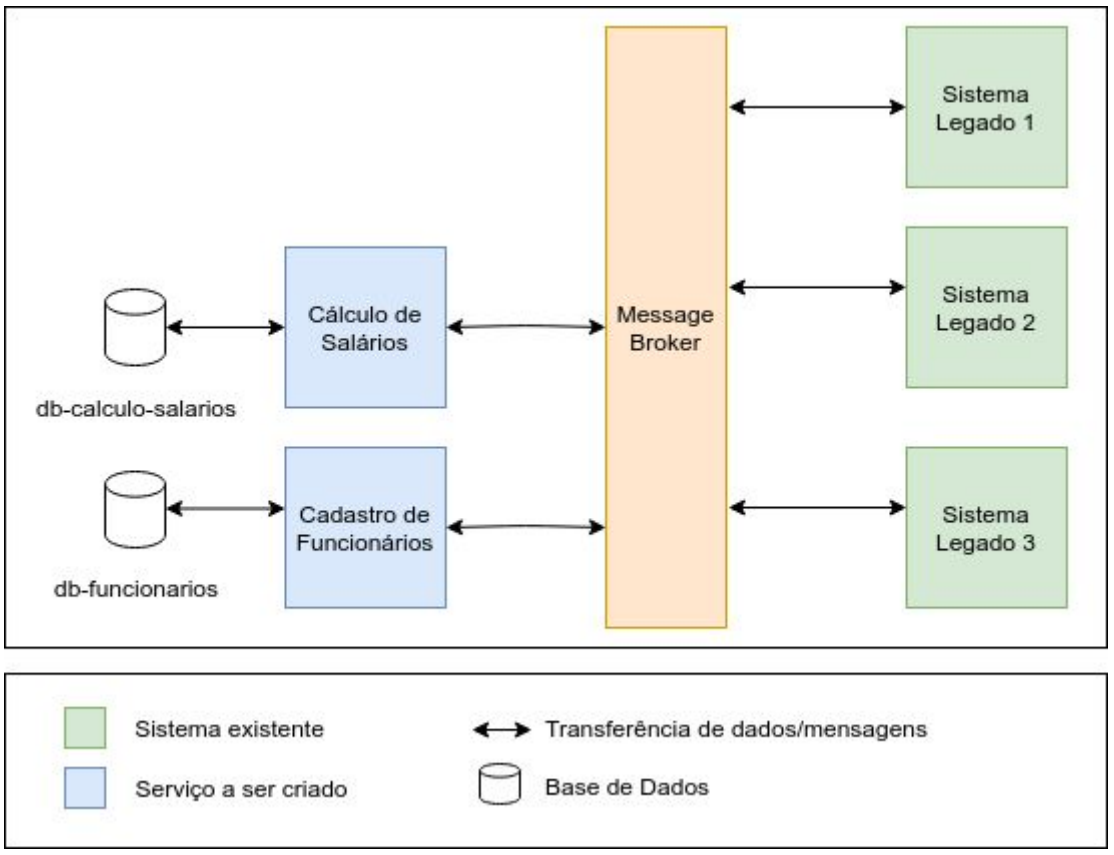
Um possível usuário para o sistema seria um Gerente de RH. Data de contratação, salário e início do contrato podem ser considerados informações para um usuário desse tipo. Já sexo, idade e e-mail podem ser encarados como dados.

A relação de funcionários ativos e com salários entre R$1000,00 a R$5000,00 pode ser considerada uma informação, já que possuem relevância para o funcionário de RH e o propósito de identificar quais profissionais receberão aumento.

2. Proponha uma arquitetura de integração do sistema com os sistemas legados da empresa e justifique a sua resposta, listando as principais características. (Vale 20% do total de pontos)

Pode ser implantado um message broker para intermediar as requisições entre o sistema de salário e os outros sistemas legados. Isso pode diminuir o acoplamento entre esses sistemas. Nele podem ser aplicadas regras de tratamento e roteamento das mensagens para uma melhor coesão na comunicação entre os sistemas.

É possível utilizar o MessageBroker como um buffer para as mensagens, o que trará maior disponibilidade aos serviços, uma vez que, esse mecanismo não exige que o receptor da mensagem esteja ativo exatamente no momento do envio, garantindo a entrega da mensagem quando a operação for normalizada.



Cálculo de Salários: Serviço responsável pelo armazenamento das regras de cálculo de salários, bem como, parametrizações que permitam identificar os funcionários que receberão aumento. Com base nos dados cadastrais dos funcionários e nas regras cadastradas, o serviço "Cálculo de Salários" calcula o aumento salarial, e envia uma mensagem solicitando a atualização dos mesmos. Com essa abordagem, pretende-se tornar o sistema mais flexível, permitindo que o próprio usuário realize calibragens nas regras de cálculo de aumento sem a necessidade de alterar o sistema.

Cadastro de Funcionários: Serviço responsável pelo armazenamento dos dados cadastrais dos funcionários. Ao receber uma mensagem de atualização de salário, o serviço persiste as informações em sua base de dados.

3. Aplique a metodologia do TDD criando um novo teste unitário onde irá cobrir a regra do item 5 do enunciado. Altere o cálculo de aumento dos funcionário para

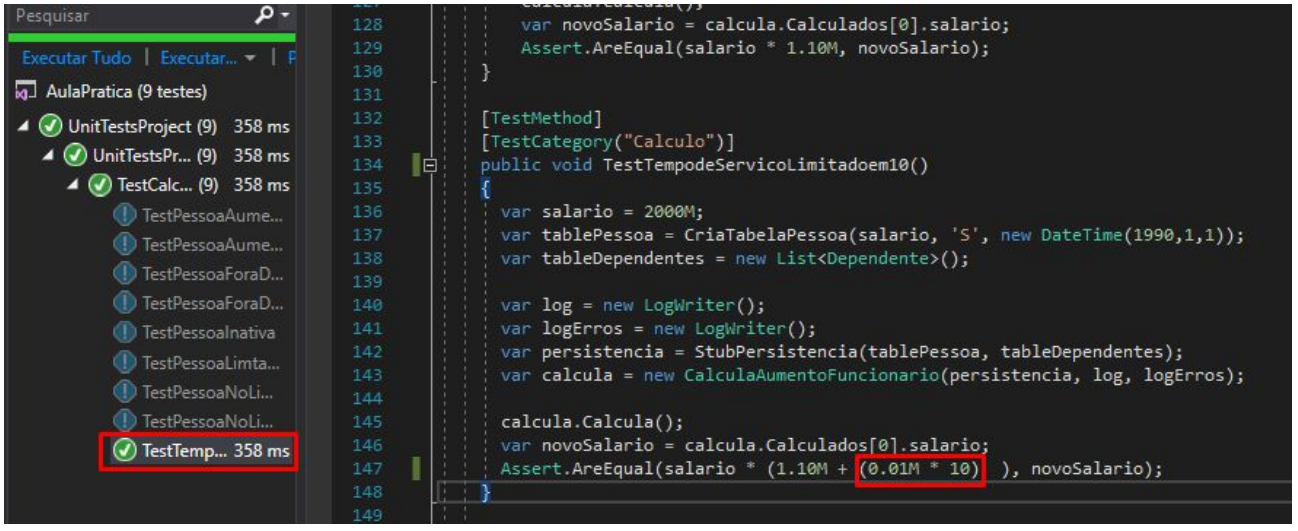
que atenda ao item 5. Envie os prints dos códigos feitos referentes a nova funcionalidade, dos testes unitários e sendo executados com sucesso tanto os testes unitários como a tela do sistema web mostrando os novos cálculos. (Vale 25% do total de pontos)

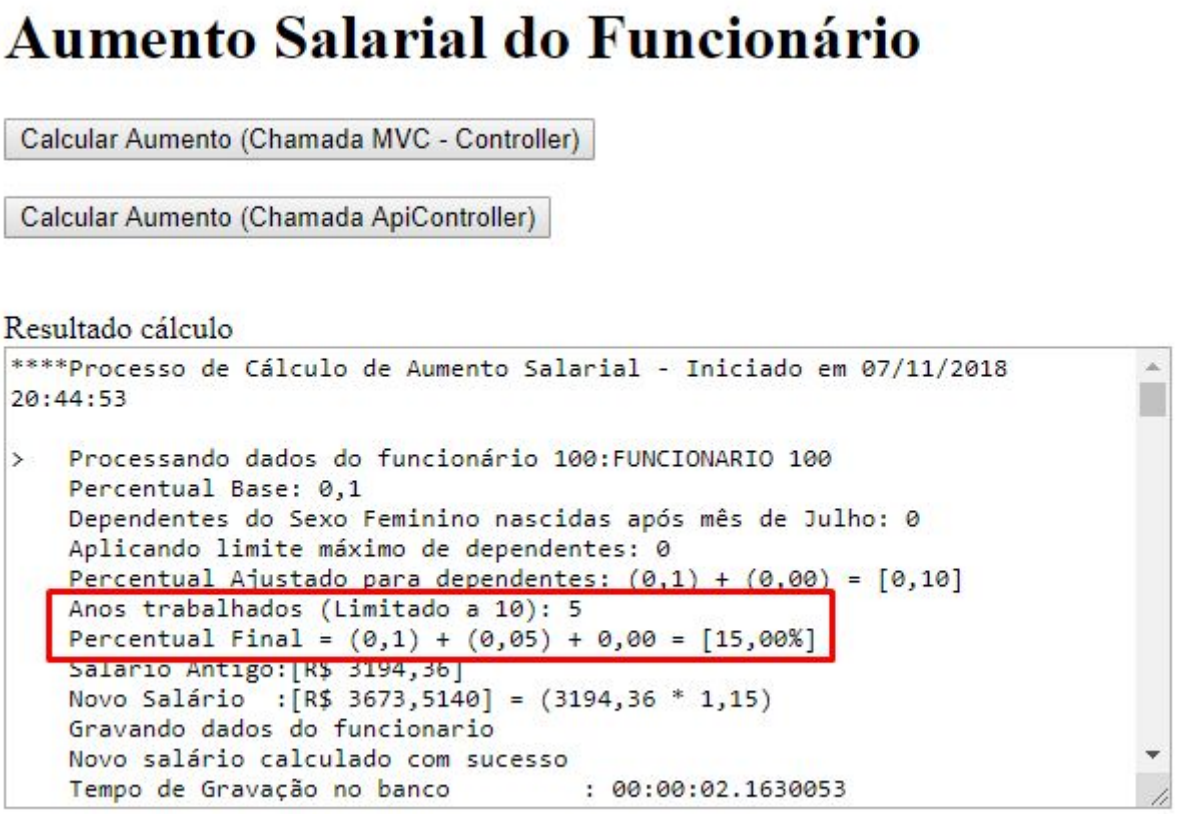
```
protected virtual decimal Calcula(Funcionario funcionario, out TimeSpan duracao)
{
  var incio = DateTime.Now;
  decimal percentual = 1M;
  if (Ativo(funcionario) && AtendeFaixaSalarial(funcionario))
  {
    WriteLog($">   Processando dados do funcionário {funcionario.id}:{funcionario.nome}");
    percentual += 0.1M;
    WriteLog($"    Percentual Base: {percentual - 1}");
    var dependentesMulheres = RetornaDependentesValidosParaCalculo(funcionario);
    WriteLog($"    Percentual Ajustado para dependentes: ({percentual-1}) + ({dependentesMulheres * 0.01M}) = [{percentual-1 + dependentesMulheres * 0.01M}]");
    var percentualDep = (0.01M * dependentesMulheres);

    var anosTrabalhados = RetornaAnosTrabalhados(funcionario);
    WriteLog($"    Anos trabalhados (Limitado a 10): {anosTrabalhados}");
    var percentualAnos = (0.01M * RetornaAnosTrabalhados(funcionario));

    PercentualCalculado = percentual + percentualAnos + percentualDep;
    WriteLog($"    Percentual Final = ({percentual-1}) + ({percentualAnos}) + {percentualDep} = [{(PercentualCalculado-1) * 100}%]  ");
    percentual += percentualAnos + percentualDep;
  }
  duracao = DateTime.Now - incio;
  return percentual;
}
```

```
protected int RetornaAnosTrabalhados(Funcionario funcionario)
{
  var anosTrabalhados = (((DateTime.Now - funcionario.inicioContrato)).Days / 365);
  anosTrabalhados = Tops(10, anosTrabalhados);
  return anosTrabalhados;
}
```





4. Utilize o BDD para descrever os cenários e testar o item 7 do enunciado. Envie os prints dos testes do SpecFlow e sendo executado com sucesso. (Vale 20% do total de pontos)

Cenário 1 : Somente funcionários Ativos (Coluna Pessoa.Ativo = “S”) poderão ter seu salário aumentado.

- Dado um funcionário com status de inativo e que tenha um salário;

- O reajuste é aplicado a toda folha e o este funcionário passa a ser NÃO CONTEMPLADO com o aumento de salário;

- Dado um funcionário com status ativo e que tenha um salário;

- O reajuste é aplicado a toda folha e o este funcionário passa a ser CONTEMPLADO com o aumento de salário;

Cenário 2: Somente funcionários com salários entre R$1000,00 a R$5000,00 deverá receber aumento salarial.

- Dado um funcionário que receba um salário maior ou igual a 1000 reais e menor ou igual a 5000 reais

- O reajuste é aplicado a toda a folha e este funcionário passa a ser CONTEMPLADO com o aumento de salário;

- Dado um funcionário que receba um salário menor do que 1000 reais ou maior do que 5000 reais

- O reajuste é aplicado a toda a folha e este funcionário passa a ser NÃO CONTEMPLADO com o aumento de salário;

Cenário 3: Para todos os funcionários que atendam aos cenários 1 e 2 deverá ser aplicado um fator base de aumento de 10%.

- O aumento concedido pela empresa aos funcionários que estejam CONTEMPLADOS com aumento de salário será de 10%

- O reajuste é aplicado a toda folha e o salário de um funcionário que esteja CONTEMPLADO com um aumento de salário e receba 1000 reais será reajustado para 1650 reais

- O reajuste é aplicado a toda folha e o salário de um funcionário que esteja NÃO CONTEMPLADO com um aumento de salário receba 5001 reais, permanecerá 800 reais;

Cenário 4: Para todos os funcionários que atendam a regra 4 e para cada ano trabalhado na empresa, o funcionário deverá receber um adicional de + 1% limitado ao máximo de 10%.

O cálculo de anos trabalhados pode ser feito usando a coluna *IniciodeContrato* da tabela Pessoa.

Exemplo: ((DateTime.Now - row.Field<DateTime>("INICIOCONTRATO")).Days / 365).

- Dado um funcionário CONTEMPLADO com aumento de salário, com 3 anos de contrato e com um salario de 1500;

- O reajuste é aplicado a toda folha e o salário deste funcionário passa a ser de 1695 reais (reajuste de 13%)

- Dado um funcionário CONTEMPLADO com aumento de salário, com 10 anos de contrato e com um salario de 1500;

- O reajuste é aplicado a toda folha e o salário deste funcionário passa a ser de 1800 reais (reajuste de 20%)

- Dado um funcionário CONTEMPLADO com aumento de salário, com 11 anos de contrato e com um salario de 1500;

- O reajuste é aplicado a toda folha e o salário deste funcionário passa a ser de 1800 reais (reajuste de 20%)

- Dado um funcionário não CONTEMPLADO com aumento de salário, com 3 anos de contrato e com um salario de 1500;

- O reajuste é aplicado a toda folha e o salário deste funcionário permanece o mesmo (reajuste de 0%)

Cenário 5: Criar uma funcionalidade para deletar do banco de dados os funcionários inativos e
retornar a quantidade deletada para a página web.

- Funcionários de todos os status possíveis são exibidos e o seu total é de 100;

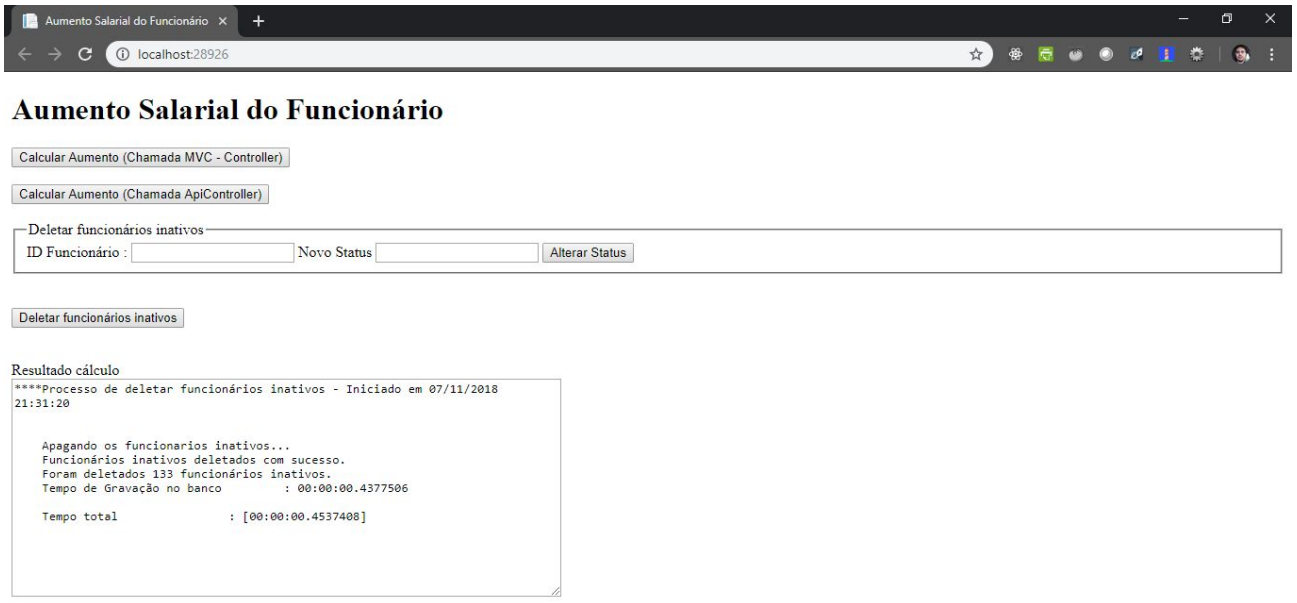
- Uma solicitação de exclusão dos funcionários com status inativos é solicitada;

- Passam a ser exibidos somente funcionários com o status ATIVO e o seu total é de 80;

- Os funcionários excluídos corresponde a 20 (100 - 80);

5. Implementar o item 7 do enunciado. Envie os prints dos códigos feitos referentes a nova funcionalidade e das telas da nova funcionalidade do sistema web sendo executado com sucesso.
    - Uma sugestão é criar um novo serviço do tipo Delete (Rest), um novo botão na tela Web com o texto Deletar Inativos e exibir uma mensagem com a

quantidade de inativos deletados que foi retornado do serviço criado. (Vale 25% do total de pontos).



6. Para verificar se o sistema web AulaPratica.sln está funcionando conforme as atividades envie o projeto compactado (.zip).
    - Lembre-se: no Canvas só é permitido fazer o upload de um arquivo. Portanto compacte todo o seu trabalho em um único arquivo (.zip).