

Relatório de Projeto LP2 - 2019.1

Lourival Netto - 119111236

Rodrigo Estrela - 118210418

Lucas Abrantes - 118111727

Gabriel Dantas - 118210140

Design geral:

O design do nosso projeto tem como foco manter o máximo de alta coesão possível, simplicidade e organização, criando packages e as classes necessárias e seus respectivos controllers, onde utilizamos em cada caso a coleção mais apropriada. Fizemos uso da delegação, onde criamos métodos internos privados que poderia resolver a lógica separadamente. Para a separação de Pessoa, decidimos utilizar herança, onde decorreria em Pessoa sem partido e Pessoa com partido, essa no qual poderia vir a se tornar um deputado posteriormente, e para isso fizemos uso do padrão strategy. Criamos uma classe à parte intitulada "Validador" que contém os métodos necessários para verificar a maioria dos parâmetros passados pelo usuário, lançado as exceções coesas quando algum parâmetro é inválido. Abaixo especificamos mais sobre cada caso.

Caso 1:

O caso 1 solicitava que pudesse ser realizado o cadastro de pessoas no sistema, que podem se tornar um deputado posteriormente, para isso criamos a classe abstrata Pessoa, que continha todos os atributos necessários para uma pessoa normal, ela contém duas classes filhas, Pessoa sem partido e Pessoa com partido essa no qual tem os atributos Partido e Função a mais. Foi criado também um Controller para pessoa que contém um HashMap que usa o DNI como chave e a Pessoa como valor.

Caso 2:

No caso 2 foi feito a solicitação para que pudesse realizar o cadastro de Deputado, onde somente uma pessoa com partido poder ser deputado, assim se faz uso da classe filha de Pessoa, Pessoa com partido, para cumprir esse requisito, para ser Deputado atribuímos isso ao atributo Função de Pessoa, onde Deputado é uma classe que implementa Função.

Caso 3:

No caso 3 foi desenvolvido o método de exibir pessoa, onde cada classe filha contém seu próprio toString, para poder realizar a exibição de forma correta.

Caso 4:

O caso 4 solicita o cadastramento de partidos governistas, para tal criamos um Controller que contém um TreeSet, que armazena todos os partidos cadastrados. Também foi implementado o método para exibição da base.

Caso 5:

No caso 5 é necessário realizar o cadastro de Comissões, onde uma comissão era composta por um tema e por uma lista de Deputados participantes, para armazenar todos os Deputados da comissão utilizamos um ArrayList, foi feito também um Controller para

armazenar e cuidar de todas as comissões, foi utilizado um HashMap para organizar tais, onde o tema da comissão é a chave e a própria comissão é o valor.

Caso 6:

No caso 6 foi requisitado um meio de cadastrar e exibir Propostas Legislativas, sendo que uma proposta legislativa pode decorrer em Projeto de Lei, Projeto de Lei Complementar ou uma Proposta de Emenda à Constituição, para cumprir tais requisitos, foi criada a classe abstrata LeiAbstract, que contém os atributos e métodos básicos necessários em qualquer proposta, e foi criada três classes filhas (PL,PLP,PEC) que contém seus atributos e métodos específicos, sendo um deles o método de exibir.

Caso 7:

No caso 7 é necessário um sistema que possa realizar a votação das Propostas Legislativas, em comissão ou no plenário posteriormente, ela inicia a votação votando primeiro na comissão da CCJC e verifica no caso da PL se ela é conclusiva ou não, caso não, ela passa ao próximo local até chegar ao plenário. A contagem de votos na comissão é realizada com base no status governista da proposta e se o deputado que vai votar simpatiza com o status, essa verificação é feita acessando o partido do deputado e vendo se condiz com o status, caso o status seja LIVRE, a contagem é feita com base nos interesses, onde é pego os interesses da proposta e os interesses do deputado, e verifica se a algum igual. Para votar no plenário a diferença é que invés de ser votada por uma comissão, a proposta é votada agora por uma lista de deputados presentes, mas que faz as mesmas verificações para contar. A classe contém métodos para verificar se a votação é válida ou não.

Considerações:

O projeto foi feito tomando cumprir o máximo de requisitos, fizemos bom uso do auxílio do monitor e serviu de grande aprendizagem sobre o trabalho com sistemas "maiores" como também sobre o trabalho em grupo.

Link para o repositório no GitHub:

https://github.com/abedess/Projeto_LP2