

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA

ANA CLARA MATIVI DE SOUZA  
LARISSA ROZALES GONÇALVES  
MATHEUS CEZIMBRA BARBIERI  
RICARDO STADTLOBER SABEDRA

Trabalho Prático TCP – Etapa 2  
DESIGNER - GRUPO 1

Porto Alegre, 6 de Maio de 2016

## Class Paper

Esta classe representa os artigos de uma conferência, utilizando dados da *Database*. Está associada com as classes *Researcher*, onde o atributo é o autor do artigo, e *Review*, onde fica a lista de review (ver a classe *Review*). Setters e getters são necessários para a manipulação de dados. O método *getMeanGrade* implementa o cálculo da média das revisões associadas ao artigo. O método *checkReviews* retorna True caso todas as reviews estejam completas.

## Class Review

Representa as revisões recebidas por artigos. Os atributos da classe associam o objeto com um *Researcher*. Para a manipulação da nota, existe setter e getter, bem como um *setReviewStatus*, que modifica o atributo tipo *boolean* sinalizando que a revisão está completa (com nota). O método *isConcluded* retorna o True quando a revisão está completa.

## Class Conference

Representa as conferências que serão alocadas. O construtor recebe os dados informados pelo usuário na interface, e o método *allocate* recebe o número de revisores que cada artigo receberá e começa a execução da alocação. O método *generateOrderedPaperList* é executado “número de revisores” vezes, criando a lista de artigos ordenada por id.

- I. O método *checkCandidate* remove da lista de candidatos aqueles que não atendem os requisitos para ser revisor. *generateOrderedReviewerList* ordena a lista de candidatos possíveis e usa *getFirstCandidate* para atribuir o primeiro candidato para revisar o artigo atual através do método *addReviewerToTempList*, pois as informações só serão salvas após a conclusão da alocação. O método *removeTopPaper* retira o artigo que já recebeu o revisor nesta iteração. A lista de revisores é reordenada de acordo com os ids e número de artigos já recebidos, utilizando o método *reorderReviewerList*. Após isso, o sistema passa para o próximo artigo, e segue até a lista de artigos estar vazia.

O passo I é repetido até o método *allocate* terminar. Após isto, os dados são salvos nos objetos *Paper* através do método *saveReviewers*.

## Class Researcher

Representa os pesquisadores, utilizando dados da Database. Ele possui métodos de comparação, que serão usados na validação de um revisor para um Paper. Quando um revisor é avaliado para ser alocado para um Paper, ele chama estes métodos para saber se este mesmo pesquisador não é o autor deste Paper, se ele possui tópicos de pesquisa de interesse na área, e se ele ainda não está alocado para este Paper.

## Class ConferenceAllocateCommand

Classe responsável por chamar o método *execute()* para alocação de artigos a membros do comitê de programa. Faz leitura de informações fornecidas pelo usuário, e chama método de alocação para alocar Papers da conferência de acordo com o algoritmo fornecido. Ao fim de sua execução, chama o método *saveReviews()* que salva os dados alocados nos objetos Paper e exibe um log contendo uma lista dos artigos da conferência e seus respectivos revisores.

## Class ReviewPaperCommand

Classe responsável por chamar o método *execute()* para atribuição de notas a artigos. Exibe a lista de artigos pendentes e solicita que o usuário escolha um para ser avaliado. Após feita essa escolha, o usuário é solicitado novamente a escolher um revisor de uma lista exibida pelo método. O usuário então informa a nota a ser atribuída, que deve estar entre -3 e 3. Caso a nota esteja fora deste limite, a exceção *ArgumentOutOfRangeException* é lançada. Se não, salva a nota com o método *setReviewGrade()* da classe Paper.

## Class ConferenceReviewsResultCommand

Classe responsável por chamar o método *execute()* para seleção de artigos. Solicita que o usuário informe uma conferência. Se existir algum artigo que ainda necessita de revisão, exibe um alerta. Se não, são exibidas duas listas após o cálculo da média de todos os artigos: uma para os artigos aceitos, com média maior ou igual a zero, e uma para os artigos rejeitados, com média menor que zero.









