



SISTEMA DE GESTÃO DE TROCAS DE TURNOS

Trabalho Prático - 1ª fase



Cesário Perneta

A73883

Duarte Freitas

A63129

Pedro Marta

A75587

Ricardo
Cardante

A75368

Tiago Fraga

A74092

Conteúdo

1.	Introdução	3
2.	Modelo de Domínio.....	4
3.	Descrição dos Use Case	5
3.1.	Solicitar troca direta	6
3.2.	Troca com estatuto.....	7
3.3.	Registar na lista de trocas.....	7
3.4.	Aceita/registar trocas	8
3.5.	Marcar Faltas	9
3.6.	Restantes	10
3.	Conclusão	14

1. Introdução

O seguinte trabalho prático advém da necessidade de aplicação e consequente cimentação de conhecimentos adquiridos nas aulas relativas à unidade curricular de Desenvolvimento de Sistemas de Software, visando temas introdutórios à cadeira, nomeadamente, modelação do sistema proposto.

A tarefa visa o desenvolvimento de um sistema de gestão dos turnos práticos de um curso (mestrado ou licenciatura), sendo apenas apresentada, nesta primeira fase, o respetivos diagramas de domínio, use cases e consequentes especificações.

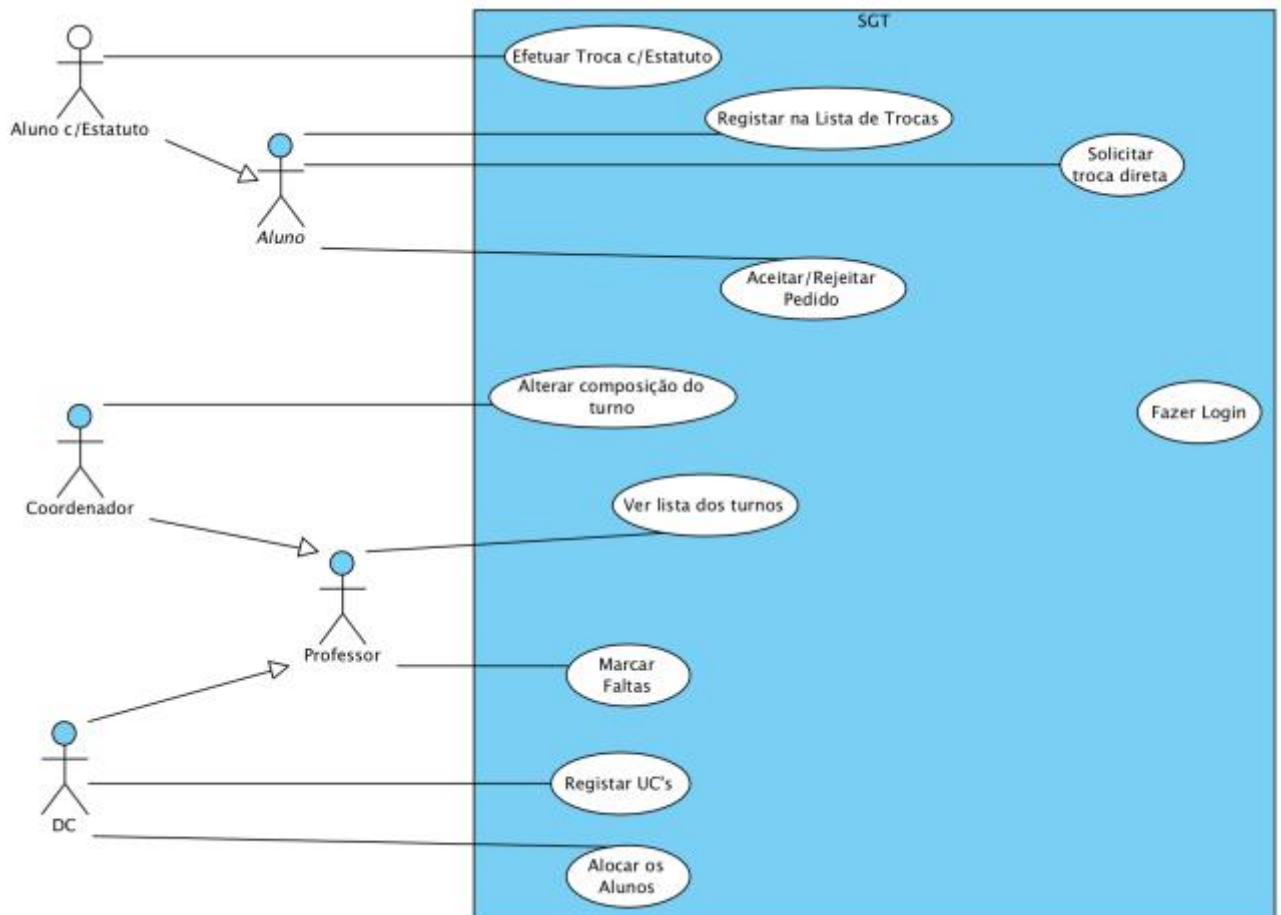
No final deste trabalho, é esperado que tenham sido corretamente aplicados conceitos introdutórios de Desenvolvimento de Sistemas de Software, de um modo organizado e estruturado, enquadrado com o problema proposto, a fim de lançar as fundações adequadas à execução dos restantes segmentos da componente prática de avaliação da unidade curricular, bem como da respetiva apreciação teórica sumativa.

Por turno haverá uma única sala, que assim como o tipo de aula e o turno, se encontra limitada por uma capacidade própria. A esse mesmo turno são lecionadas múltiplas aulas, às quais podem ser dadas faltas pelos alunos nelas inscritos, o que leva a que quando ultrapassado certo limite, ocorra a perda de lugar nesse turno.

O aluno, assim como o professor, tem que levar a cabo o seu registo na plataforma. A si poderá ter associado um estatuto especial caso o mesmo cumpra funções de trabalho/estudante, desportista ou outras, o que trará a mais valia da troca c/ estatuto, a qual permite a transferência de um turno para outro (da mesma) UC, bastando haver vagas disponíveis no destino (depende da capacidade do dito turno). Estando um aluno inserido num contexto normal, sem estatuto, poderá realizar tais trocas de turno a partir de permutas, quer por via de troca normal (contacto directo com um outro aluno com interesse no turno contrário), quer por via de inscrição na lista de trocas, a qual é relativa a um determinado turno.

Restará referir que um professor vai lecionar uma unidade curricular, podendo ou não ser o coordenador da mesma. Cada unidade curricular fica limitada a um coordenador, assim como o curso só pode ter um diretor.

3. Descrição dos Use Case



A necessidade de Use Cases chega com a identificação dos vários cenários de utilização do software, podendo assim ser estipulado o contexto de interação entre utilizador e sistema, bem como todos os atores e serviços presentes no mesmo.

Assim, com a definição de cada Use Case, é adicionada uma unidade coerente de funcionalidade, peça a peça e sem revelar a estrutura interna do programa, de modo em que última instância, como agrupamento de todos estes serviços, se obtenha a funcionalidade do sistema.

Dos Use Cases definidos no contexto do problema, destacam-se:

3.1. Solicitar troca direta

Permite que os alunos possam trocar de turno entre si, sem necessidade de estatuto especial ou intervenção por parte de outro ator.

Assumindo o pré-povoamento do sistema, alocação dos turnos e estado do aluno autenticado, bastará indicar ao sistema o nº do aluno com o qual pretende fazer a troca. Daí, será efetuada a verificação da existência do aluno e, se tal caso se comprovar, o aluno que solicitou a troca será notificado, requerendo confirmação por parte do mesmo, para que o aluno “destinatário” possa ser notificado por parte do sistema. Caso o aluno indicado para troca de turno não seja válido, o primeiro é notificado e a ação termina a sua execução.

Use Case: Solicitar uma troca direta.		
Descrição:	O Aluno notifica o sistema, para solicitar outro aluno que este pretende trocar de turno com ele.	
Pré-Condição:	Sistema está povoado, os turnos já foram alocados, o aluno fez o login.	
Pós-Condição:	O Aluno envia pedido ao outro aluno.	
Comportamento Normal	Ator(Aluno)	Sistema
	1. Notifica o sistema com que aluno pretende fazer a troca.	
		2. O sistema verifica se o aluno existe.
		3. Notifica o aluno que o pedido pode ser efetuado.
	4. Aluno confirma	
Exceção [O Aluno não existe]. (passo2)		5. Notifica o outro aluno do pedido da troca.
		2.1 - Notifica o aluno que o colega pretendido não está registado no sistema.

3.2. Troca com estatuto

Permite a um aluno com estatuto especial (trabalhador/estudante, desportista, entre outros) trocar de turno desde que haja vagas no mesmo, sem necessidade de indicar um outro aluno com quem fazer a troca, ou intervenção por parte de outro ator do sistema.

Assim, basta ao aluno indicar o turno pretendido, para que o sistema passe à verificação de existência de vagas no mesmo. Em caso positivo, o aluno é notificado, havendo a necessidade de confirmação por parte do mesmo, de modo a que a troca seja efetivada pelo sistema. Caso o turno venha a revelar falta de capacidade, o aluno é notificado sobre o facto e a ação termina a sua execução.

Use Case:	Efetuar uma troca com estatuto	
Descrição:	O Aluno solicita ao sistema para efetuar uma troca para o turno pretendido, sem ter de trocar com outro aluno, pois tem estatuto.	
Pré-Condição:	Sistema está povoado, os turnos já foram alocados e o aluno fez o login.	
Pós-Condição:	O Aluno fica com o turno pretendido.	
Comportamento Normal	Ator(Aluno)	Sistema
	1. Notificar qual o turno pretendido.	
		2.Recebe o pedido e verifica se o turno tem capacidade para o receber.
		3.Notifica o aluno que o turno pode recebe-lo.
	4.Aluno confirma.	
Excepção [O turno nao tem capacidade]. (passo2)		5.Efetua a troca
		2.1 - Notifica o aluno que o turno está lotado e nao o pode receber.

3.3. Registrar na lista de trocas

Permite a um aluno submeter a sua inscrição numa lista de trocas de turnos.

Assumindo o pré-povoamento do sistema, alocação dos turnos e estado do aluno autenticado, bastará indicar ao sistema o turno para o qual o aluno se pretende transferir. Daí, por parte do sistema, será registado o turno pretendido (anteriormente indicado), bem



como o atual turno do aluno, seguido da notificação da inscrição na lista ao aluno, que posteriormente terá que confirmar a ação.

Use Case:	Registrar na Lista de Trocas	
Descrição:	O Aluno solicita ao sistema para ser introduzido numa lista de trocas de turnos.	
Pré-Condição:	Sistema está povoado, os turnos ja foram alocados e o aluno fez o login.	
Pós-Condição:	O Aluno fica inscrito na lista.	
Comportamento Normal	Ator(Aluno)	Sistema
	1. Notificar o sistema qual o turno que pretende	
		2. O sistema regista qual o turno pretendido e qual o turno a que pertence no momento
		3.Notifica o aluno que esta inscrito na lista de trocas.
	4.Aluno confirma.	

3.4. Aceita/registar trocas

Permite a um aluno responder positivamente/negativamente a um pedido de troca de turno solicitado por outro aluno.

Assumindo o pré-povoamento do sistema, alocação dos turnos e estado do aluno autenticado, é iniciado pelo sistema o processo de comunicação, notificando o aluno de que existe um pedido de troca de turno pendente. O aluno deverá aceitar/rejeitar a proposta, ao que o sistema, com a receção de tal informação, deverá transmiti-la ao aluno responsável pela proposta da troca.

Use Case:	Aceitar ou Rejeitar um pedido de troca de turno.	
Descrição:	O Aluno notifica o sistema que pretende aceitar ou rejeitar o pedido de troca de turno que outro aluno lhe fez.	
Pré-Condição:	Sistema está povoado, os turnos já foram alocados, o aluno fez o login, e outro aluno fez-lhe um pedido de troca	
Pós-Condição:	O Aluno fica com o turno pretendido.	
Comportamento Normal	Ator(Aluno)	Sistema
		1.O sistema notifica o aluno que tem um pedido de troca de turno.
	2. O Aluno aceita ou rejeita esse pedido.	
		3. O sistema recebe a informação.
		4. Notifica o outro aluno com a resposta correspondente.

3.5. Marcar Faltas

Permite a realização do registo de faltas referentes a determinado aluno, por parte de um docente.

Assumindo o pré-povoamento do sistema, alocação dos turnos e estado do professor autenticado, bastará indicar o aluno ao qual deverá ser marcada a falta. Daí, o sistema receberá o pedido, executá-lo-á, restando ao professor confirmar o pedido. Caso o aluno não integre a base de dados do sistema, o sistema deverá notificar o professor, levando posteriormente ao término da execução desta ação.



Use Case: Marcar Faltas		
Descrição:	O professor solicita ao sistema para registar as faltas de um determinado aluno.	
Pré-Condição:	Sistema está povoado, os turnos ja foram alocados e o professor fez login.	
Pós-Condição:	O professor marca uma falta a um aluno.	
Comportamento Normal	Ator(Professor)	Sistema
	1. Notificar qual o aluno a quem pretende marcar falta.	
		2. O sistema recebe o pedido.
		3. O sistema executa o pedido.
	4. O professor confirma.	
Excepção. [O aluno pretendido nao existe no sistema]. (passo2)		2.1 O aluno a quem o professor pretende marcar falta nao está registado.

3.6. Restantes

Entre as restantes especificações, por não concentrarem em si particular interesse, por já se encontrarem explanadas no âmbito de diferentes Use Cases de comportamento homólogo ou por apresentarem um carácter generalista e direto, não relevante ao problema proposto, segue apenas a especificação tabular dos mesmos elaborada pelo grupo.

- Alocação de alunos

Use Case: Alocar os Alunos		
Descrição:	O Diretor de Curso solicita ao sistema para fazer a alocação automática dos alunos.	
Pré-Condição:	Sistema está povoado e o diretor de curso fez login.	
Pós-Condição:	Os alunos ficam distribuídos pelos turnos.	
Comportamento Normal	Ator(Diretor de Curso)	Sistema
	1. Notificar que pretende gerar a alocação automática dos turnos.	
		2. O sistema recebe o pedido.
		3. O sistema efetua o pedido.
Exceção. [Nao ha alunos no sistema]. (passo2)		2.1 Notifica o utilizador, indicando-lhe que nao existe alunos no sistema.

- Alteração de composição de turno

Use Case: Alterar a composição de um turno.		
Descrição:	O Coordenador solicita o sistema para fazer alterações nos turnos que pertencem à UC que coordena.	
Pré-Condição:	Sistema está povoado, os turnos ja foram alocados e o professor fez o login.	
Pós-Condição:	O Coordenador faz as trocas que pretende.	
Comportamento Normal	Ator(Professor Coordenador)	Sistema
	1. Notifica o sistema com o numero de aluno que pretende mover.	
		2. O sistema recebe a informação.
	3. Indica o numero do turno para onde pretende mudar o aluno.	
		4. O sistema recebe a informação e confirma que pode ser efetuada a troca.
Exceção [O aluno nao existe]. (passo2)		2.1 - O sistema recebe a informação, no entanto o aluno nao existe no sistema.
Exceção. [O turno, onde pretende colocar o aluno, está cheio]. (passo 4)		4.1 O sistema recebe a informação e notifica o professor que esse turno está lotado.



- Listagem de turnos

Use Case:	Ver lista dos turnos	
Descrição:	O professor solicita ao sistema para ver a lista dos alunos do turno que leciona	
Pré-Condição:	Sistema está povoado, os turnos já foram alocados e o professor fez login.	
Pós-Condição:	O professor visualiza os alunos dos turnos que leciona.	
Comportamento Normal	Ator(Professor)	Sistema
	1. Notificar qual o turno pretendido.	
		2. O sistema recebe o pedido.
		3. Apresenta a lista dos alunos do turno selecionado.
	4. Professor sai do sistema.	
Excepção. [O professor não leciona nenhum turno]. (passo2)		2.1 O sistema recebe o pedido mas notifica o professor que ele não leciona nenhum turno.
Alternativa. [Visualizar mais turnos] (passo4)	4.1 Voltar ao passo 1.	

- Autenticação de utilizador

Use Case: Fazer Login	
Descrição:	O ator solicita ao sistema para efetuar o login.
Pré-Condição:	O ator está registado no sistema.
Pós-Condição:	O ator entra no sistema.
Comportamento Normal	Ator(Professor ou Aluno)
	Sistema
	1. Notifica o sistema que pretende efetuar o login.
	2. O sistema recebe o pedido.
	3. Solicita ao ator os dados de acesso.
	4. Introduz os dados de acesso ao sistema.
Exceção [Dados inválidos] (passo5)	5. O sistema recebe os dados.
	6. Valida o acesso ao ator.
Exceção [Ator inexistente] (passo5)	5.1 Indica que os dados introduzidos estão incorrectos.
	5.1 Indica que o ator que tentou aceder ao sistema nao está registado.

3. Conclusão

Com o término da presente fase, é admissível afirmar que todos os objetivos requeridos pelo docente no enunciado cedido foram alcançados, nomeadamente, um auxílio à compreensão da realidade e estrutura do projeto, de modo a planear de forma rigorosa e metódica o seu desenvolvimento, com o fim último de prevenir possíveis riscos/erros de código e fracas implementações.

Assim, nesta fase, foram elaborados dois tipos de diagramas: o modelo de domínio e o de Use Cases (com respetivas especificações). O primeiro permitiu obter uma visão mais topográfica da aplicação, descrevendo todas as entidades a estar integradas no produto final, bem como todos os relacionamentos entre as mesmas. Já o segundo, cumpriu o propósito de modelar as ações possíveis por parte de cada tipo de ator, bem como o comportamento de tais interações, a um nível mais específico, com a troca de intervenções e respostas entre o respetivo ator e o sistema.

A maior dificuldade com que nos deparamos durante a realização do trabalho foi a de conjugar a modelação dos diagramas com o que futuramente vai ser a nossa aplicação, tentando nesta fase executar as tarefas num nível mais amplo, mas tentando não fugir ao que já idealizamos para a realização da tarefa seguinte.

A inclusão de ferramentas como o software Visual Paradigm, bem como folhas de cálculo Excel, permitiram ao grupo agilizar todo o processo associado modelação do sistema proposto, ferramentas essas de significativa utilidade para várias outras unidades curriculares, mas com especial destaque para a que aqui se encontra em foco.

A qualidade do trabalho apresentado agrada ao grupo, tendo havido tempo suficiente para o desenvolvimento das soluções necessárias aos exercícios propostos, sem precipitações e com espaço de maneio suficiente para que todos os membros pudessem inteirar-se do trabalho realizado.