

CENTRALMUSIC - LDS

TEST CASE OUTLINE

Version **1.3**

16/02/2021

Histórico de Versões

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	<i>Samuel Cunha</i>	<i>16/02/2021</i>	<i>Samuel Cunha</i>	<i>16/02/2021</i>	Adaptação do documento para o projeto
1.1	<i>Samuel Cunha</i>	<i>16/02/2021</i>	<i>Samuel Cunha</i>	<i>16/02/2021</i>	Adicionadas todas as secções para o começo da descrição dos cenários de testes
1.2	<i>Samuel Cunha</i>	<i>16/02/2021</i>	<i>Samuel Cunha</i>	<i>16/02/2021</i>	Adicionados casos de teste para todos os componentes
1.3	<i>Samuel Cunha</i>	<i>16/02/2021</i>	<i>Samuel Cunha</i>	<i>16/02/2021</i>	Finalização dos componentes UserDAO, LoginDAO, PublicationDAO e ChatDAO

Tabela de Conteúdos

1. Introdução.....	5
1.1. Identificador do documento	5
1.2. Âmbito.....	5
1.3. Referencias.....	5
1.4 Glossário	5
2. Detalhes.....	6
2.1 Test Cases do componente <i>UserDAO</i>	6
2.1.1 Método UpdatePassword	7
2.1.2 Método FindUserByEmail	8
2.1.3 Método FindByld	11
2.2 Test Cases do componente <i>EmployerDAO</i>	13
2.2.1 Método Create	13
2.2.2 Método Update	14
2.2.3 Método FavoriteList	Erro! Marcador não definido.
2.2.4 Método AddFavorite	Erro! Marcador não definido.
2.2.5 Método RemoveFavorite	Erro! Marcador não definido.
2.2.6 Método FindEmployerByld	Erro! Marcador não definido.
2.3 Test Cases do componente <i>MateDAO</i>	15
2.3.1 Método Create	15
2.3.2 Método CategoriesList	Erro! Marcador não definido.
2.3.3 Método AddCategory	Erro! Marcador não definido.
2.3.4 Método RemoveCategory	Erro! Marcador não definido.
2.3.5 Método Update	Erro! Marcador não definido.
2.3.6 Método FindMateByld	Erro! Marcador não definido.
2.3.7 Método PendingJobsList	Erro! Marcador não definido.
2.3.8 Método GetMates	Erro! Marcador não definido.
2.3.9 Método IgnoreJobPost	Erro! Marcador não definido.
2.4 Test Cases do componente <i>JobDAO</i>	18
2.4.1 Método AddPayment	18
2.4.2 Método RemovePayment	18
2.4.3 Método GetJobs	19
2.4.4 Método MakeOfferOnJob	Erro! Marcador não definido.
2.4.5 Método Create	Erro! Marcador não definido.
2.4.6 Método Delete	Erro! Marcador não definido.
2.4.7 Método FindByld	Erro! Marcador não definido.
2.4.8 Método GetEmployerPosts	Erro! Marcador não definido.
2.4.9 Método UpdatePostDetails	Erro! Marcador não definido.
2.5 Test Cases do componente <i>WorkDAO</i>	Erro! Marcador não definido.
2.5.1 Método MarkJobAsDone	Erro! Marcador não definido.
2.5.2 Método Create	Erro! Marcador não definido.
2.5.3 Método FindByld	Erro! Marcador não definido.
2.5.4 Método Delete	Erro! Marcador não definido.
2.5.5 Método Update	Erro! Marcador não definido.
2.6 Test Cases do componente <i>LoginDAO</i>	Erro! Marcador não definido.
2.6.1 Método Authenticate	Erro! Marcador não definido.
2.7 Test Cases do componente <i>ReportDAO</i>	Erro! Marcador não definido.
2.7.1 Método ReportUser	Erro! Marcador não definido.

2.8 Test Cases do componente <i>ReviewEmployerDAO</i>	Erro! Marcador não definido.
2.8.1 Método ReviewEmployer	Erro! Marcador não definido.
2.9 Test Cases do componente <i>ReviewMateDAO</i>	Erro! Marcador não definido.
2.9.1 Método ReviewMate	Erro! Marcador não definido.
2.10 Test Cases do componente <i>DistanceHelper</i>	Erro! Marcador não definido.
2.10.1 Método GetAddressFromCoordinates	Erro! Marcador não definido.
2.11 Test Cases do componente <i>PaymentDAO</i>	Erro! Marcador não definido.
2.11.1 Método GetInvoiceById	Erro! Marcador não definido.
2.11.2 Método MakePayment	Erro! Marcador não definido.
2.11.3 Método ConfirmPayment	Erro! Marcador não definido.
2.12 TestCases do componente ChatDAO	Erro! Marcador não definido.
2.12.1 Método AddMessage	Erro! Marcador não definido.
2.12.2 Método CreateChat	Erro! Marcador não definido.
2.12.3 Método GetMessageList	Erro! Marcador não definido.
2.12.4 Método GetMessageList	Erro! Marcador não definido.
2.12.5 Método GetChats	Erro! Marcador não definido.

1. Introdução

1.1. Identificador do documento

TCO16022021V1.3

Descrição do identificador do projeto:

TCO - TestCaseOutline

16-Numero do mês(dois dígitos seguidos dos dois dígitos iniciais)

02-Dia do mês(dois dígitos iniciais)

2021-Ano de criação(quatro dígitos finais)

V1.3-Numero de versão do projeto(dois caracteres sendo "V" o identificador de versão através de sigla seguido do numero de versão "1" e numero de alteração ".3")

1.2. Âmbito

Este documento encontra-se enquadrado no âmbito do documento *Test Design Specification*, referido na secção Referências deste documento. Porém este documento apresenta um objetivo um pouco diferente, ou seja, visa a apresentar os dados de acordo com as descrições definidas no âmbito do documento *Test Design Specification*.

1.3. Referencias

Lista de referências relevantes para este test case outline:

- a) Test Design Specification.
- b) <https://gitlab.estg.ipp.pt/8160526/centralmusic>

1.4 Glossário

MC – Marcelo Carvalho

SC – Samuel Cunha

2. Detalhes

2.1 Test Cases do componente *UserDAO*

2.1.1 Método Create

Test Case ID	TC1	Dependências	---		
Classes de Equivalência Válidas	EC1				
Classes de Equivalência Inválidas	EC2				
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha	
16/02/21	19:16	SC	Pass	N/A	
Pré-condições / estado inicial					
Tem de ser possível qualquer utilizador não registado utilizar a rota que usa este método					
Procedimento / Passos do user para esta função					
1. O utilizador regista os seus dados 2. A API executa a rota com o método Create 3. O utilizador é registado no sistema					
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested	
Realizar registo	Objeto de utilizador	Strings: email, password, primeiro nome, último nome, morada, imagem	Utilizador registado (True)	Pass	

2.1.2 Método Update

Test Case ID	TC1	Dependências	---		
Classes de Equivalência Válidas	EC1				
Classes de Equivalência Inválidas	EC2				
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha	
16/02/21	19:25	SC	Pass	N/A	
Pré-condições / estado inicial					
Tem de estar instanciado um utilizador					
Procedimento / Passos do user para esta função					
1. O utilizador faz o Login na aplicação					
2. O utilizador visita o seu perfil e seleciona a opção de editar dados pessoais					
3. Edita os dados e guarda					

Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Editar dados pessoais	Objeto de utilizador, ID do utilizador	Strings: Email, primeiro nome, último nome e morada. Inteiro: ID do utilizador	Utilizador atualiza dados (True)	Pass

2.1.3 Método UpdatePassword

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	19:29	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador				
Procedimento / Passos do user para esta função				
1. O utilizador faz o Login na aplicação 2. O utilizador visita o seu perfil e seleciona a opção de alterar password. 3. O utilizador faz a alteração da sua palavra passe, introduzindo a palavra passe atual e de seguida a nova. 3. O utilizador verifica se foi retornado True.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Alterar password	Objeto de UserUpdatePassword, Id do utilizador	Password atual e nova password, ID do utilizador	Alteração da password (True)	Pass

2.1.4 Método GetPublications

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	19:35	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado algum tipo de utilizador e o mesmo tem de ter pelo menos uma publicação				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador visita o seu perfil e seleciona as suas publicações ativas				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Visualizar anúncios do próprio	ID do utilizador	ID do utilizador (inteiro)	Visualização dos anúncios (True)	Pass

2.1.5 Método UpdatePublications

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	19:39	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado algum tipo de utilizador e o mesmo tem de ter pelo menos uma publicação				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador visita o seu perfil e seleciona as suas publicações ativas 3. O mesmo edita a publicação que pretende				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Editar anúncio	Objeto de Publication	Título, Descrição, booleano trocável, Categoria, double preço, imagem, Condição, Morada e raio	Publicação editada (True)	Pass

2.1.6 Método DeletePublications

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	20:48	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado algum tipo de utilizador e o mesmo tem de ter pelo menos uma publicação				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador visita o seu perfil e seleciona as suas publicações ativas 3. O mesmo elimina a publicação que deseja				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Remover anúncio	ID da publicação	ID da publicação (inteiro)	Eliminação da publicação (True)	Pass

2.1.7 Método GetPublicationFromFavorites

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	20:50	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado algum tipo de utilizador e o mesmo tem de ter pelo menos uma publicação nos favoritos				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador visita o seu perfil e seleciona as suas publicações favoritas 3. O mesmo escolhe a publicação que deseja visualizar				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Visualizar anúncios favoritos	ID do utilizador	ID do utilizador (inteiro)	Visualização da publicação (True)	Pass

2.1.8 Método RemovePublicationFromFavorites

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	20:52	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado algum tipo de utilizador e o mesmo tem de ter pelo menos uma publicação nos favoritos				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador visita o seu perfil e seleciona as suas publicações favoritas 3. O mesmo elimina a publicação que deseja				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Remover anúncios favoritos	ID do utilizador, ID da publicação	ID do utilizador (inteiro), ID da publicação (inteiro)	Remoção da publicação (True)	Pass

2.1.9 Método FindUserByEmail

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	20:55	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador				
Procedimento / Passos do user para esta função				
1. O utilizador regista-se na plataforma. 2. O utilizador verifica se as suas informações retornadas pelo método correspondem ao pedido 3. Retorna o utilizador pretendido				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Encontrar utilizador	String com um email	email	Utilizador pretendido (True)	Pass

Test Case ID	TC2	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	20:56	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador				
Procedimento / Passos do user para esta função				
1. O utilizador regista-se na plataforma. 2. O utilizador verifica se as suas informações retornadas pelo método correspondem ao pedido 3. Retorna nulo				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Encontrar utilizador	String com um email	email	Nulo	Pass

2.1.10 Método FindByld

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	20:57	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador.				
Procedimento / Passos do user para esta função				
1. Um utilizador regista-se na plataforma. 2. Esse utilizador tenta procurar outro utilizador específico. 3. A api retorna o utilizador encontrado 4. O Utilizador consegue ver os detalhes da sua procura.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Encontrar utilizador	ID do utilizador	ID do utilizador (inteiro)	Utilizador pretendido (True)	Pass

Test Case ID	TC2	Dependências	---	
Classes de Equivalência Válidas	EC3			
Classes de Equivalência Inválidas	EC4			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	20:57	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador.				
Procedimento / Passos do user para esta função				
1. Um utilizador regista-se na plataforma. 2. Esse utilizador tenta procurar outro utilizador específico. 3. A api retorna uma resposta de utilizador não encontrado.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Encontrar utilizador	ID do utilizador	ID do utilizador (inteiro)	Null	Pass

2.2 Test Cases do componente *LoginDAO*

2.2.1 Método Authenticate

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	21:01	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador				
Procedimento / Passos do user para esta função				
1. O utilizador regista-se na plataforma 2. O utilizador tenta fazer login na aplicação 3. O mesmo entra na aplicação				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Realizar Login	String com email, string com password	Email, password	True	Pass

Test Case ID	TC2	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
08/12/20	18:55	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador				
Procedimento / Passos do user para esta função				
1. O utilizador regista-se na plataforma 2. O utilizador tenta fazer login na aplicação 3. O utilizador não consegue entrar na aplicação				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Realizar Login	String com email, string com password	Email, password	False	Pass

2.2.2 Método RecoverPassword

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	21:04	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador				
Procedimento / Passos do user para esta função				
1. O utilizador carrega na opção para recuperar a password 2. O utilizador segue as instruções para alterar a password				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Recuperar password	Objeto de RecoverPasswordModel, email do utilizador	Email, Password, Confirm Password, Token	True	Pass

2.3 Test Cases do componente *PublicationDAO*

2.3.1 Método Create

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	21:08	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de ser instanciado um utilizador				
Procedimento / Passos do <i>user</i> para esta função				
1. O utilizador faz login na plataforma 2. O utilizador carrega na opção para criar uma publicação 3. O mesmo submete a publicação com os dados que inseriu				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Publicar anúncio	Objeto de Publication	Título, descrição, trocável, categoria, preço, imagem. condição, morada	Publicação criada (True)	Pass

2.3.2 Método FindById

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	21:11	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador e pelo menos uma publicação.				
Procedimento / Passos do user para esta função				
1. O utilizador faz login na aplicação 2. Esse utilizador tenta procurar uma publicação específica.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested

Encontrar publicação pelo ID	ID da publicação	ID da publicação (inteiro)	Publicação pretendida (True)	Pass
------------------------------	------------------	----------------------------	------------------------------	------

2.3.3 Método GetPublications

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	21:15	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador e tem de haver pelo menos uma publicação				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador pesquisa uma publicação usando o filtro de categoria				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Filtrar por categorias	Categorie	Enum de Categoria	Visualização dos anúncios (True)	Pass

Test Case ID	TC2	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Data de Execução
16/02/21	21:17	SC	Pass	16/02/21
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador e tem de haver pelo menos uma publicação				
Procedimento / Passos do <i>user</i> para esta função				
1. O utilizador faz Login na plataforma				
2. O utilizador pesquisa uma publicação usando o filtro de categoria				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Requisito/Use Case/ Funcionalidade
Filtrar por distância	Distância	Inteiro de distância	Visualização dos anúncios (True)	Filtrar por categorias

Test Case ID	TC3	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Data de Execução
16/02/21	21:18	SC	Pass	16/02/21
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador e tem de haver pelo menos uma publicação				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador pesquisa uma publicação usando a barra de pesquisa				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Requisito/Use Case/ Funcionalidade
Filtrar por barra de pesquisa	String	String com o nome do instrumento a pesquisar	Visualização dos anúncios (True)	Filtrar por distância

2.3.4 Método AddPublicationToFavorites

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	21:21	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado um utilizador e tem de haver pelo menos uma publicação				
Procedimento / Passos do user para esta função				
1. O utilizador faz Login na plataforma 2. O utilizador ao pesquisar uma publicação, adiciona a mesma à lista de favoritos				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Adicionar aos favoritos	ID do utilizador, ID da publicação	Inteiros: ID do utilizador, ID da publicação	Publicação adicionada aos favoritos (True)	Pass

2.4 Test Cases do componente *ChatDAO*

2.4.1 Método AddMessage

Test Case ID	TC1	Dependências	---		
Classes de Equivalência Válidas	EC1				
Classes de Equivalência Inválidas	EC2				
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha	
16/02/21	21:25	SC	Pass	N/A	
Pré-condições / estado inicial					
Tem de estar pelo menos dois utilizadores, um Chat e tem de haver pelo menos uma publicação					
Procedimento / Passos do user para esta função					
1. Um utilizador carrega na opção para enviar mensagem ao outro utilizador através da publicação 2. O utilizador envia uma mensagem 3. O outro utilizador recebe a mensagem					
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested	
Chat para negociação	Objeto de mensagem	ID do chat, string com a mensagem, ID do remetente, Data	True	Pass	

2.4.2 Método CreateChat

Test Case ID	TC1	Dependências	---		
Classes de Equivalência Válidas	EC1				
Classes de Equivalência Inválidas	EC2				
Data de Execução	Hora de Execução	Tester	Pass/Fail	Data de Execução	
16/02/21	21:28	SC	Pass	11/01/21	
Pré-condições / estado inicial					
Tem de estar pelo menos dois utilizadores					
Tem de haver pelo menos uma publicação					
Procedimento / Passos do user para esta função					
1. O utilizador carrega na opção para enviar mensagem a partir da publicação					
2. O utilizador é enviado para um chat com o utilizador alvo					
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested	
Chat para negociação	Objeto de Chat	ID do utilizador, ID do chat	Chat adicionado à BD com os utilizadores (True)	Pass	

2.4.3 Método GetMessageList

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Data de Execução
16/02/21	21:31	SC	Pass	11/01/21
Pré-condições / estado inicial				
Tem de estar instanciado dois utilizadores, um Chat e tem de haver pelo menos uma publicação				
Procedimento / Passos do user para esta função				
1. O Employer da Match com o Mate 2. O Employer acede a lista de Match e verifica que foi adicionado uma nova conversa 3. O utilizador ao carregar na <i>bottom bar</i> para aceder a todas as conversas, ao carregar num chat, verifica que todas as mensagens da conversa foram carregadas				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Requisito/Use Case/ Funcionalidade
Chat para negociação	ID do Chat	ID do Chat	Lista de Mensagens (True)	Pass

2.4.4 Método GetChats

Test Case ID	TC1	Dependências	---	
Classes de Equivalência Válidas	EC1			
Classes de Equivalência Inválidas	EC2			
Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha
16/02/21	21:37	SC	Pass	N/A
Pré-condições / estado inicial				
Tem de estar instanciado dois utilizadores e um Chat				
Procedimento / Passos do user para esta função				
1. O utilizador acede à lista de chats 2. A app mobile pede os chats à API 3. O utilizador recebe os chats a que pertence				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Chat para negociação	ID do utilizador	ID do utilizador	Array de chats (True)	Pass