

<PROJETO>

## TEST CASE OUTLINE

---

Version <x.0>

<dd/mm/yyyy>

### Histórico de Versões

Version #	Implemented By	Revision Date	Approved By	Approval Date	Reason
1.0	<Author name>	<mm/dd/yy>	<name>	<mm/dd/yy>	draft

## Tabela de Conteúdos

<b>1. Introdução.....</b>	<b>4</b>
1.1. Identificador do documento .....	4
1.2. Âmbito.....	4
1.3. References.....	4
1.4 Glossário .....	4
<b>2. Detalhes.....</b>	<b>4</b>
2.1. Identificador do caso de teste.....	4
2.2. Objetivo ou requisito/funcionalidade associado.....	4
2.3. Inputs .....	4
2.6. Estado inicial ou pré-condições .....	4
2.5. Resultados esperados .....	4
2.6. Procedimentos especiais ou STEPS.....	4
2.7. Dependência de outros casos .....	4

# 1. Introdução

## 1.1. Identificador do documento

[Fornecer um identificador único para este documento. Este identificador único será usado por outros documentos para referenciar este de forma unívoca.]

## 1.2. Âmbito

## 1.3. References

[Fazer referência a outros documentos importantes para uma melhor compreensão deste. Exemplo: documento de requisitos, Master Test Plan, ...]

## 1.4 Glossário

# 2. Detalhes

[cada uma das seções abaixo deverá ser repetida para cada test case. Em alternativa poderá usar-se uma tabela. Uma das principais diferenças entre as seções 2.3 e 2.4 do documento Test Case Design e o test case em si, está diretamente relacionado com os dados, ou seja, este documento pretende apresentar os dados de acordo com as descrições definidas no âmbito do documento test case design]

## 2.1. Identificador do caso de teste

## 2.2. Objetivo ou requisito/funcionalidade associado

## 2.3. Inputs

## 2.6. Estado inicial ou pré-condições

## 2.5. Resultados esperados

## 2.6. Procedimentos especiais ou STEPS

## 2.7. Dependência de outros casos

AddAccount(User user)

Test Case ID	B1.1	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

Pré-condições / estado inicial				
Base de dados iniciada e vazia.				
Procedimento / Passos do user para esta função				
1. Chamar método usando user válido como argumento.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Adicionar conta à base de dados	user	New User ("username", "password")	true	

<b>Test Case ID</b>	B1.2	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados iniciada e com users.				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método usando user com nome já existente como argumento.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Adicionar conta à base de dados	user	New User ("username", "password")	Throws exception	

LogIn(string username, string password)

<b>Test Case ID</b>	B2.1	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com users				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método LogIn com combinação username e password existente na base de dados como argumentos.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Autenticação de conta	String username, string password	("username","password")	True	

<b>Test Case ID</b>	B2.2	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com users				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método LogIn com combinação username e password inexistente na base de dados como argumentos.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Autenticação de conta	String username, string password	("username","password")	False	

GetUserById(int id)

<b>Test Case ID</b>	B3.1	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com users				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetUserById com id correspondente a um user registado na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber user com um id específico	Int id	("id")	true	

<b>Test Case ID</b>	B3.2	<b>Dependências</b>	
---------------------	------	---------------------	--

Test Case outline - <Item>

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com users				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetUserById com id que não correspondente a um user registado na base de dados.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Receber user com um id específico	Int id	("id")	false	

GetAllUsers()

Test Case ID	B4.1	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com users				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetAllUsers.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Receber todos os users registados			true	

Test Case ID	B4.2	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

<b>Pré-condições / estado inicial</b>				
Base de dados existe vazia				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetAllUsers.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Receber todos os users registados			false	

EditUser(User user)

Test Case ID	B5.1	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com users				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método EditUser com user registado na base de dados.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Editar dados de user	user	("user")	true	

Test Case ID	B5.2	Dependências	
--------------	------	--------------	--



Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

**Pré-condições / estado inicial**

Base de dados existe e com users

**Procedimento / Passos do user para esta função**

1. Chamar método EditUser com user inexistente na base de dados.

Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Editar dados de user	user	("user")	false	

DeleteUser(int id)

Test Case ID	B5.1	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

**Pré-condições / estado inicial**

Base de dados existe e com users

**Procedimento / Passos do user para esta função**

1. Chamar método DeleteUser com id correspondente a um user registado na base de dados.

Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Eliminar user	id	("id")	true	

Test Case ID	B5.2	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

**Pré-condições / estado inicial**

Base de dados existe e com users

**Procedimento / Passos do user para esta função**

1. Chamar método DeleteUser com id correspondente que não corresponde a um user registado na base de dados.

Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Eliminar user	id	("id")	false	

GetMatches(int id)

<b>Test Case ID</b>	B6.1	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

**Pré-condições / estado inicial**

Base de dados existe e com users

**Procedimento / Passos do user para esta função**

1. Chamar método GetMatches com id correspondente a um user registado na base de dados.

<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber todos os Matches de um certo user	id	("id")	true	

<b>Test Case ID</b>	B6.2	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

**Pré-condições / estado inicial**

Base de dados existe e com users

**Procedimento / Passos do user para esta função**

1. Chamar método GetMatches com id que não correspondente a um user registado na base de dados.

<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber todos os Matches de um certo user	id	("id")	false	

AddBook(Book book);

Test Case ID	B7.1	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

Pré-condições / estado inicial				
Base de dados iniciada				
Procedimento / Passos do user para esta função				
1. Chamar método usando book válido como argumento.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Adicionar book à base de dados.	book	New Book ("name", "Author", "State")	true	

Test Case ID	B7.2	Dependências	
--------------	------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

Pré-condições / estado inicial				
Base de dados iniciada				
Procedimento / Passos do user para esta função				
1. Chamar método usando book sem o campo name preenchido como argumento.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Adicionar book à base de dados.	book	New Book ("Author", "State")	true	

<b>Test Case ID</b>	B7.3	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados iniciada				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método usando book sem o campo Author preenchido como argumento.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Adicionar book à base de dados.	book	New Book ("name", "State", "Photo")	true	

<b>Test Case ID</b>	B7.4	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados iniciada				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método usando book sem o campo State preenchido como argumento.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Adicionar book à base de dados.	book	New Book ("name", "Author", "Photo")	true	

<b>Test Case ID</b>	B7.5	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados iniciada				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método usando book sem o campo Photo preenchido como argumento.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Adicionar book à base de dados.	book	New Book ("name", "Author", "State")	true	

GetBooks());

<b>Test Case ID</b>	B8.1	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com books				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetBooks.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber todos os books registados			true	

<b>Test Case ID</b>	B8.2	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe vazia				
<b>Procedimento / Passos do <i>user</i> para esta função</b>				
1. Chamar método GetBooks.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber todos os books registados			false	

GetBookById(int id);

<b>Test Case ID</b>	B9.1	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com books				
<b>Procedimento / Passos do <i>user</i> para esta função</b>				
1. Chamar método GetBook(int Id) com id correspondente a um book registado na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber book com um id específico	Int id	("id")	true	

<b>Test Case ID</b>	B9.2	<b>Dependências</b>	
---------------------	------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com books				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetBook com id que não correspondente a um book registado na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber book com um id específico	Int id	("id")	false	

DeleteBook(int id);

<b>Test Case ID</b>	B10.1	<b>Dependências</b>	
---------------------	-------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com books				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método DeleteBook com id correspondente a um book registado na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Eliminar book	id	("id")	true	

<b>Test Case ID</b>	B10.2	<b>Dependências</b>	
---------------------	-------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com users				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método DeleteBook com id correspondente que não corresponde a um book registado na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Eliminar book	id	("id")	false	

#### EditBook(int id)

<b>Test Case ID</b>	B11.1	<b>Dependências</b>	
---------------------	-------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com books				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método EditBook com book registado na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Editar dados de book	user	("user")	true	



<b>Test Case ID</b>	B11.2	<b>Dependências</b>	
---------------------	-------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com books				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método EditBook com book inexistente na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Editar dados de book	user	("user")	false	

SubmitReport(Report report);

<b>Test Case ID</b>	B12.1	<b>Dependências</b>	
---------------------	-------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados iniciada				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método usando um report válido como argumento.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Submeter Report na base de dados	report	New Report ("reporterId", "reportedId", "Text");	true	

Test Case ID	B12.2	Dependências	
--------------	-------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

<b>Pré-condições / estado inicial</b>				
Base de dados iniciada				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método usando um reporterId inválido como argumento.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Submeter Report na base de dados	report	New Report ("reportedId", "Text");	false	

## GetReports()

Test Case ID	B13.1	Dependências	
--------------	-------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com reports				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetReports.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber todos os reports registados			true	

<b>Test Case ID</b>	B13.2	<b>Dependências</b>	
---------------------	-------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe vazia				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetReports.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber todos os reports registados			false	

GetReport(int id);

<b>Test Case ID</b>	B14.1	<b>Dependências</b>	
---------------------	-------	---------------------	--

<b>Data de Execução</b>	<b>Hora de Execução</b>	<b>Tester</b>	<b>Pass/Fail</b>	<b>Resumo da Falha</b>

<b>Pré-condições / estado inicial</b>				
Base de dados existe e com reports				
<b>Procedimento / Passos do user para esta função</b>				
1. Chamar método GetReport(int Id) com id correspondente a um report registado na base de dados.				
<b>Requisito/Use Case/ Funcionalidade</b>	<b>Inputs</b>	<b>Valores do Inputs</b>	<b>Resultados Esperados</b>	<b>Pass/Fail/Untested</b>
Receber report com um id específico	Int id	("id")	true	

Test Case ID	B14.2	Dependências	
--------------	-------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

Pré-condições / estado inicial				
Base de dados existe e com reports				
Procedimento / Passos do user para esta função				
1. Chamar método GetReport(int Id) com id que não corresponde a um report registado na base de dados.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Receber report com um id específico	Int id	("id")	false	

SendMessage(Message message)

Test Case ID	B17.1	Dependências	
--------------	-------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

Pré-condições / estado inicial				
Base de dados existe com Users válidos				
Procedimento / Passos do user para esta função				
1. Chamar método SendMessage() com message válido como argumento.				
Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Enviar mensagem dada como argumento	Message	New Message {"text","senderId","receiverId","time"}	true	

Test Case ID	B17.2	Dependências	
--------------	-------	--------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

Pré-condições / estado inicial				
Base de dados existe com Users válidos				
Procedimento / Passos do user para esta função				

Test Case outline - <Item>

1. Chamar método SendMessage() com message como argumento que corresponde a utilizador inexistente na base de dados.

Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untestado
Enviar mensagem dada como argumento	Message	New Message {"text","senderId","receiverId","time"} }	false	

ChangeStatusById(int id);

<b>Test Case ID</b>	B16.1	<b>Dependências</b>	
---------------------	-------	---------------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

**Pré-condições / estado inicial**

Base de dados existe e com "messages"

**Procedimento / Passos do user para esta função**

1. Chamar método ChangeStatusById(int id)

Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Mudar Status da Message com o id dado	Int id	{"id"}	true	

<b>Test Case ID</b>	B16.2	<b>Dependências</b>	
---------------------	-------	---------------------	--

Data de Execução	Hora de Execução	Tester	Pass/Fail	Resumo da Falha

**Pré-condições / estado inicial**

Base de dados existe vazia

**Procedimento / Passos do user para esta função**

1. Chamar método ChangeStatusById(int id) com id que não corresponde a um Message registado na base de dados.

Requisito/Use Case/ Funcionalidade	Inputs	Valores do Inputs	Resultados Esperados	Pass/Fail/Untested
Mudar Status da Message com o id dado	Int id	{"id"}	false	

