

Instituto Superior de Engenharia do Porto
Departamento de Engenharia Electrotécnica
Rua Dr. António Bernardino de Almeida, 431, 4249-015 Porto

SQL Stored Procedures

Mestrado em Engenharia Eletrotécnica e de Computadores

Alexandre Amorim [1161497]

Supervisor: Prof. Maria Benedita Campos Neves Malheiro

Ano Letivo: 2020-2021

Índice

Índice	i
Índice de Figuras	iii
Índice de Tabelas	v
Acrónimos e Abreviaturas	vii
1 Introdução	1
1.1 Enquadramento e Objetivos	1
1.2 Motivação e Metodologia	1
1.3 Estrutura	2
2 Estado de Arte	3
2.1 Relevância no Contexto de RINTE	4
2.2 Vantagens e Desvantagens	4
2.3 Sintaxe	5
2.4 Sumário	5
3 Instalação e Configuração	7
3.1 Servidor Microsoft SQL	8
3.2 SQL Server Management Studio	9
3.3 Sumário	10
4 Biblioteca RINTE	11
4.1 Modelo da Base de Dados	12
4.2 Algoritmo	13
4.3 Interface Gráfica	15
4.4 Testes não funcionais	16
4.5 Sumário	17

5 Conclusão	19
Referências Bibliográficas	21

Índice de Figuras

2.1	Tipos de <i>Stored Programs</i>	3
2.2	Funcionalidade das <i>Stored Procedures</i>	4
2.3	Sintaxe comparativa entre uma operação <i>Linguagem de Consulta Estruturada</i> (SQL) direta e uma <i>Stored Procedure</i> (SP)	5
3.1	Visão holística de instalação e configuração dos <i>softwares</i> em estudo .	7
3.2	<i>Download</i> e configuração do servidor SQL	8
3.3	Finalização da configuração do servidor SQL	8
3.4	<i>Download</i> e configuração do <i>SQL Server Management Studio</i> (SSMS) .	9
3.5	Instalação e inicialização do SSMS	9
3.6	Inicialização das SP	10
3.7	Verificação do nome do sistema	10
4.1	<i>Pipeline</i> desenvolvida para a aplicação	11
4.2	Diagrama da Base de Dados (BD) da biblioteca Redes Inteligentes e Serviços (RINTE)	12
4.3	<i>Overview</i> dos tabelas de dados da biblioteca RINTE	12
4.4	Variáveis participantes na SP de requisição	13
4.5	Testes de validade dos dados de entrada	13
4.6	Resultados obtidos através do sucesso da SP de requisição	14
4.7	Resultados obtidos através do sucesso da SP de devolução	14
4.8	Página inicial da biblioteca RINTE	15
4.9	Interface dos processos realizados com SP	15
4.10	Mensagens de erros	15
4.11	Interface dos dados principais da BD	16
4.12	Interface dos livros que o utilizador dispõe atualmente	16

Índice de Tabelas

2.1	Vantagens e desvantagens das <i>Stored Procedures</i>	5
4.1	Resumo dos resultados obtidos para os testes não funcionais	17

Acrónimos e Abreviaturas

BD Base de Dados

DBMS *Sistema de Gestão de Base de Dados*

DML *Linguagem de Manipulação de Dados*

GUI *Interface Gráfica de Utilizador*

HTML *Linguagem de Marcação de Hipertexto*

HTTP *Protocolo de Transferência de Hipertexto*

MSSQL *Microsoft SQL Server*

SF *Stored Function*

SP *Stored Procedure*

SPg *Stored Program*

SQL *Linguagem de Consulta Estruturada*

SSMS *SQL Server Management Studio*

RINTE *Redes Inteligentes e Serviços*

WWW *World Wide Web*

Capítulo 1

Introdução

1.1 Enquadramento e Objetivos

O trabalho desenvolvido neste relatório, corresponde ao primeiro momento de avaliação da unidade curricular de RINTE, constante no plano curricular do mestrado em Engenharia Eletrotécnica e Computadores.

O objetivo deste trabalho consiste em investigar, explicar e descrever a relevância das *Stored Procedures* no contexto de RINTE. Incluindo a exemplificação dos procedimentos de instalação e configuração da aplicação proposta, assim como o desenvolvimento e a avaliação da mesma.

Relativamente as tecnologias preteridas, recorreu-se ao *Microsoft SQL Server* (MSSQL) como motor de BD, ao SSMS como ferramenta de administração de BD com *Interface Gráfica de Utilizador* (GUI) e ao .NET para o desenvolvimento da *Web App*.

1.2 Motivação e Metodologia

O despotelamento da Era Digital conduziu a uma disrupção tecnológica sem precedentes. Tornando-se fulcral entender o seu cerne: as aplicações e serviços distribuídos para a World Wide Web (WWW). Neste contexto, foi desenvolvida uma *app* que contemple SP, de modo a entender em que medida permitem obter aplicações com respostas mais céleres e com maiores níveis de segurança de informação.

O trabalho realizado segue uma abordagem tradicional de um sistema de engenharia, dispondo de uma estrutura robusta para o desenvolvimento de um projeto desta natureza. Podem ser identificadas diversas etapas dentro desta metodologia, tais como as que se encontram de acordo com a seguinte ordem:

- Análise das metodologias e das tecnologias atuais relevantes para o projeto em causa;
- Desenvolvimento de um manual de instalação e configuração das tecnologias em causa;
- Implementação, testes, validação e avaliação das soluções desenvolvidas;
- Documentação da solução proposta.

De salientar, que existem constantemente atualizações e novas tecnologias para o desenvolvimento de projetos desta natureza, assim espera-se que em propostas futuras seja possível a incorporação de novas metodologias.

1.3 Estrutura

Relativamente a estrutura deste relatório, encontra-se essencialmente dividida em cinco capítulos. O capítulo 1 contém uma breve contextualização do âmbito e dos objetivos do trabalho. Em relação ao capítulo 2, é descrito o conceito de SP, a sua relevância no contexto de RINTE, as suas vantagens/desvantagens em preterimento a outras soluções e por fim a sua sintaxe. Posteriormente, no capítulo 3 encontra-se presente um manual de instalação e configuração das tecnologias necessárias para a realização do trabalho proposto. A implementação, testes, validação e avaliação do exemplo desenvolvido podem ser encontrada no capítulo 4. Ulteriormente, no capítulo 5 estão presentes as principais conclusões advenientes da elaboração do trabalho e de propostas futuras.

Capítulo 2

Estado de Arte

Atualmente, os *Sistema de Gestão de Base de Dados* (DBMS) dominantes são baseados em *SQL*, tal como o *MySQL*¹. As suas funcionalidades resultam de programas computacionais armazenados e executados dentro do servidor de BD, denominados por *Stored Program* (SPg) [1, 2].

Segundo Harrison e Feuerstein [2006] [1], e Deep [2020] [2], os SPg subdividem-se em três tipos principais de programas armazenados em *MySQL* (Figura 2.1):

- *Stored Procedures*: Os SP são um conjunto de instruções *SQL* reutilizáveis, executadas por pedido. Podendo aceitar vários parâmetros de entrada e saída, de modo a satisfazer consultas parametrizáveis.
- *Stored functions*: As *Stored Function* (SF) são semelhantes as SP, retornando imperiosamente um valor. Ademais, aceitam apenas parâmetros de entrada e podem ser chamadas dentro das SP, sendo que o contrário não se verifica.
- *Triggers*: São programas ativados automaticamente em resposta a uma operação *Linguagem de Manipulação de Dados* (DML) (*DELETE*, *UPDATE*, etc.). Sendo que não recebem parâmetros de entrada nem retornam valores.

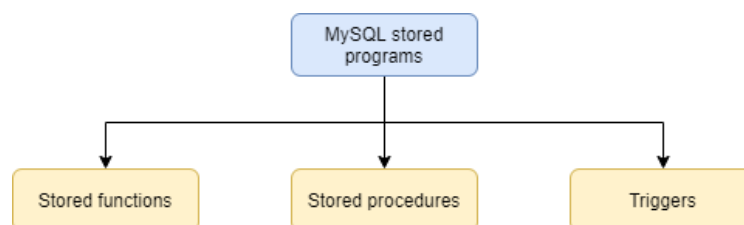


Figura 2.1: Tipos de *Stored Programs*

¹<https://db-engines.com/en/ranking>

2.1 Relevância no Contexto de RINTE

Habitualmente, as *Web Apps* contêm serviços que requerem a execução de conjuntos de operações SQL na sua respetiva BD. De acordo com Shehuh e Ahmetaj [2020], e o próprio *site* da Oracle, as SP são frequentemente empregues neste contexto, sendo de seguida explicado segundo as mesmas fontes como é efetuado este processo.

Quando um utilizador navega numa página Web ou pressiona um botão do tipo *SUBMIT* num formulário *Linguagem de Marcação de Hipertexto* (HTML), um pedido *Protocolo de Transferência de Hipertexto* (HTTP) é gerado para o servidor Web. O que leva a invocação de um SPg ao servidor de BD e sendo lhe transmitido as informações fornecidas no formulário HTML, bem como a identificação dos processos requeridos como um conjunto de parâmetros.

Ulteriormente, as SP retornam o resultado das operações solicitadas ao servidor Web, e por conseguinte ao navegador Web do utilizador. Obtendo os resultados que solicitou na sua interface, como ilustrado na Figura 2.2.

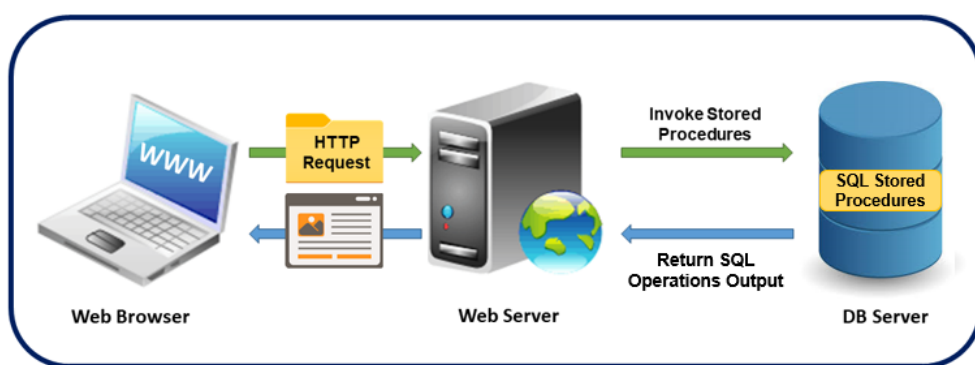


Figura 2.2: Funcionalidade das *Stored Procedures*

2.2 Vantagens e Desvantagens

No entanto, o leitor poderá se questionar quais são as vantagens de optar pelas SP, em contraste a realizar diretamente uma *query* a BD. Neste âmbito, realizou-se uma comparação sobre as suas vantagens e desvantagens relativamente a *query* diretas a BD, como retratado na Tabela 2.1 baseada de [1, 3, 4].

Vantagens	Desvantagens
Elevada segurança	<i>Debug</i> complexo
Mais célere	Elevado <i>overhead</i>
Pode ser testado independentemente	Maiores custos financeiros
Pré-compilado	Não é portátil
Reutilizável	Mão de obra qualificada
Reduz o tráfego de rede	Sintaxe varia de DBMS para DBMS

Tabela 2.1: Vantagens e desvantagens das *Stored Procedures*

De salientar, que a maior celeridade e eficiência, advêm dos SP serem compilados uma vez e armazenados de forma executável (pré-compilados). Adicionalmente, os elevados níveis de segurança resultam do acesso aos dados poderem ser restritos segundo os privilégios do utilizador.

2.3 Sintaxe

Uma das vertentes essenciais da programação computacional é a sintaxe da linguagem em estudo. Deste modo, realizou-se uma comparação entre a sintaxe de uma operação SQL direta e de uma SP, como retratado na Figura 2.3.

Numa primeira análise, o utilizador poderá interpretar que uma SP necessita de mais código que o seu némesis. Em que na realidade, a cada execução basta apenas realizar o comando **EXECUTE**, necessitando assim de menos código. Ademais, pode se verificar os três grandes constituintes de uma SP (nome, corpo e execução do procedimento), segundo Harrison e Feuerstein [2006] [1] e Forta [2020] [4].

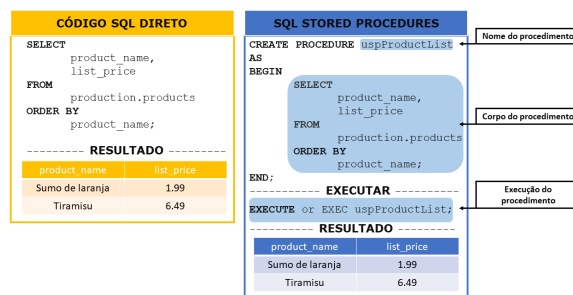


Figura 2.3: Sintaxe comparativa entre uma operação SQL direta e uma SP

2.4 Sumário

Neste capítulo, foram estabelecidos bases de conhecimentos necessários para compreender os restantes capítulos. Este conhecimento inclui a definição de SP, a sua relevância no contexto de RINTE, as suas vantagens e por fim a sua sintaxe.

Capítulo 3

Instalação e Configuração

Neste capítulo, irão ser descritos os procedimentos de instalação e de configuração do MSSQL (motor de BD) e do SSMS (sistema de administração de BD com GUI). Deste modo, propõe-se duas abordagens para este fim: uma visão holística como representado no fluxograma da Figura 3.1, e de seguida uma abordagem mais detalhada.

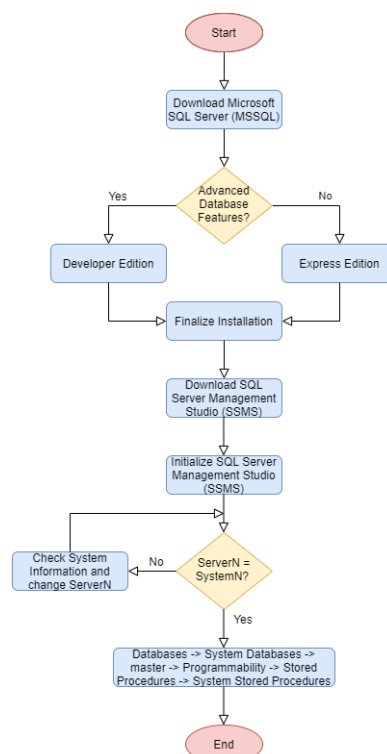


Figura 3.1: Visão holística de instalação e configuração dos *softwares* em estudo

3.1 Servidor Microsoft SQL

Inicialmente, o utilizador deverá navegar no *link*³, optar pelo *download* da edição do servidor SQL 2019 *Developer* (passo 1) e quando finalizado executar o ficheiro SQL2019-SSEI-Dev.ex (passo 2), como ilustrado na Figura 3.2(a). Relativamente a escolha do tipo de instalação, a Figura 3.2(b) evidencia a sugestão de optar pela edição básica, de modo a tornar o processo de configuração/instalação mais célere.

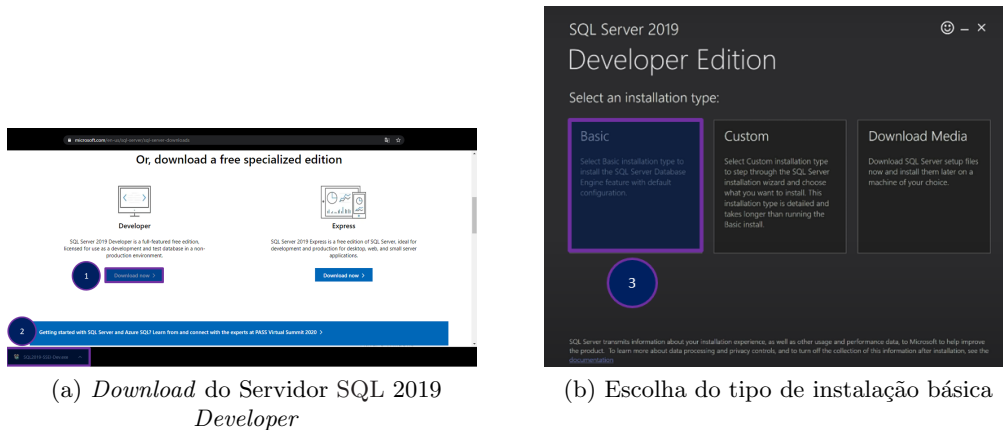


Figura 3.2: Download e configuração do servidor SQL

A *posteriori*, deverá realizar a aceitação dos termos de licença (passo 4 - Figura 3.3(a)) e especificar o local de instalação (passo 5 - Figura 3.3(b)). Sendo que se sugere analisar o *link*⁴ para obtenção de mais informações concernente as restantes opções não seleccionadas.

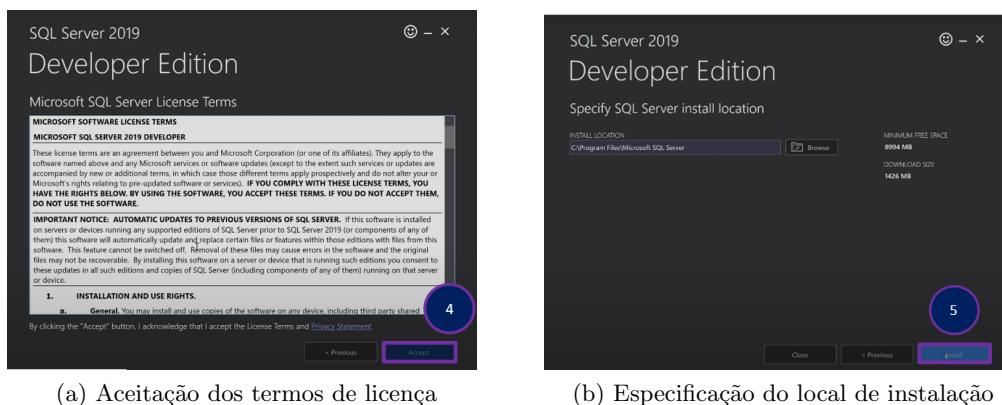


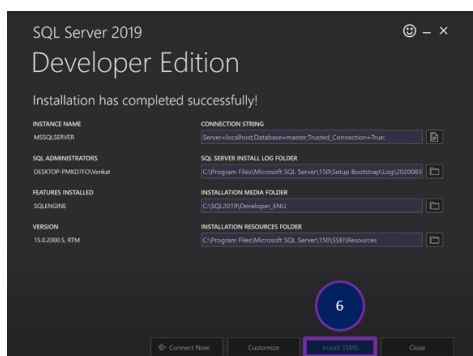
Figura 3.3: Finalização da configuração do servidor SQL

³<https://www.microsoft.com/en-us/sql-server/sql-server-downloads>

⁴<https://docs.microsoft.com/en-us/sql/sql-server/editions-and-components>

3.2 SQL Server Management Studio

Ulteriormente, é necessário realizar o mesmo processo para o SSMS. Ao finalizar as etapas anteriores, o utilizador será confrontado com a interface presente na Figura 3.4(a), onde deverá premir o botão *Install SSMS* (passo 6). Posteriormente, será reencaminhado para o *site*⁵ ilustrado na Figura 3.4(b), no qual efetuará o *download* do SSMS (passo 7) e executará o ficheiro obtido (passo 8).



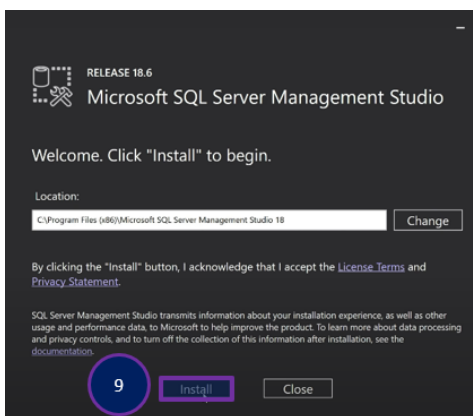
(a) Transição para o *download* do SSMS



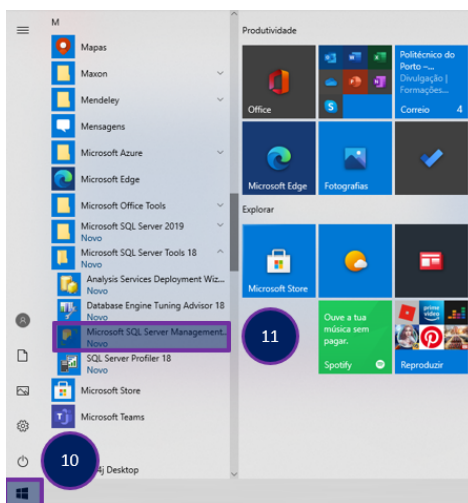
(b) *Download* do SSMS

Figura 3.4: *Download* e configuração do SSMS

Seguidamente, o utilizador deverá especificar o local de instalação pretendido SSMS (passo 8 - Figura 3.5(a)) e realizar o caminho de inicialização do SSMS (passo 9), como pode ser visualizado na Figura 3.5(b).



(a) Especificação do local de instalação do SSMS

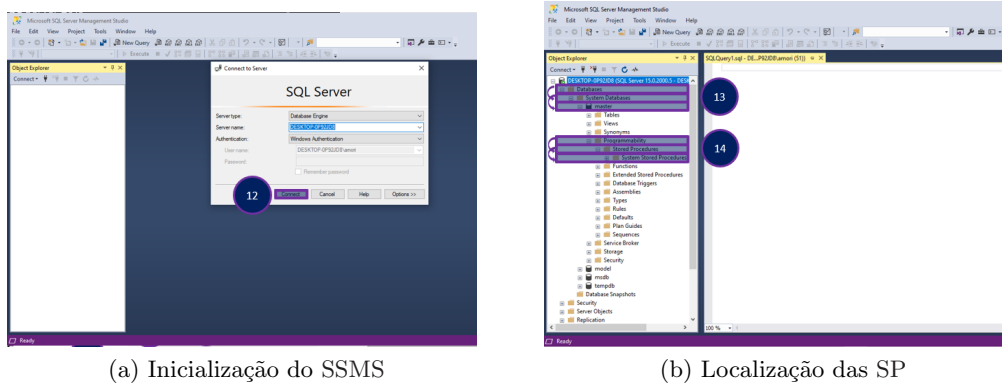


(b) Endereço de inicialização SSMS

Figura 3.5: Instalação e inicialização do SSMS

⁵<https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio>

Finalmente, resta apenas iniciar o SSMS (passo 10 - Figura 3.6(a)) e efetuar o caminho ilustrado na Figura 3.6(b): Databases -> System Databases -> master -> Programmability -> Stored Procedures -> System Stored Procedures. De salientar que é recomendável verificar se o nome do servidor é homólogo ao do sistema em operação. Podendo este último ser identificado através dos passos explicitados na Figura 3.7.



(a) Inicialização do SSMS

(b) Localização das SP

Figura 3.6: Inicialização das SP

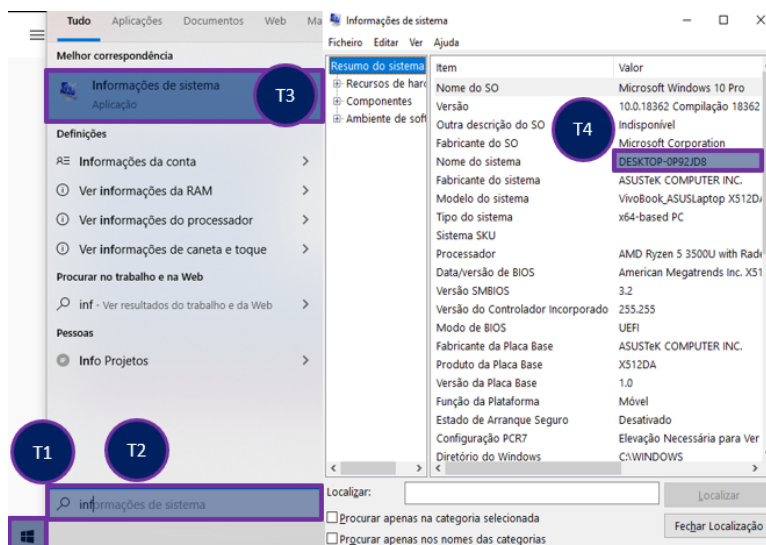


Figura 3.7: Verificação do nome do sistema

3.3 Sumário

Este capítulo, descreveu as tecnologias que permitiram a realização da *Web App* exemplificativa (servidor SQL e SSMS), bem como os seus processos de instalação e configuração.

Capítulo 4

Biblioteca RINTE

Posteriormente a conclusão do Capítulo 3, reúnem-se os elementos necessários para a realização de SP. Deste modo, elaborou-se uma demonstração exemplificativa das mesmas na ótica de uma *Web App*, que permite requisitar e devolver um livro.

O método de requisição é caracterizado através de um sistema de pontos. Cada utilizador possui 10 pontos, podendo requisitar tantos livros quanto o número de pontos que dispõe. Sendo que a cada livro foi atribuído uma determinada raridade, que se converte em pontos (de 1 à 5, sendo 1 para os mais comuns e 5 para os mais raros) necessários para os requisitar. Em contraste, o processo de devolução consiste em atualizar a disponibilidade do livro na BD.

Assim, este capítulo encontra-se dividido consoante a ordem de trabalhos efetuada para o fim anteriormente proposto: criação da BD, desenvolvimento das SP e por fim a sua integração com a *Web App* concebida, como presente na *pipeline* da Figura 4.1.

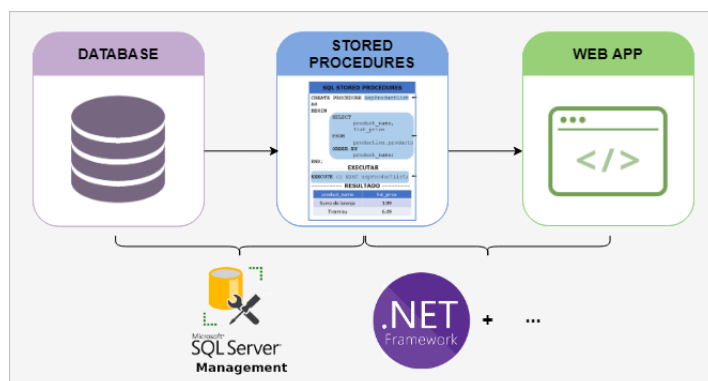


Figura 4.1: *Pipeline* desenvolvida para a aplicação

4.1 Modelo da Base de Dados

Inicialmente, foi elaborada uma base de dados constituída por 6 tabelas: identificação do autor (id, primeiroNome e últimoNome); identificação do livro (id, idAutor, ISBN, idRaridade, título e disponível); identificação do membro (id, primeiroNome, últimoNome e pontosDisponíveis); identificação da raridade (id, pontos e descrição); identificação da requisição (id, idMembro, idLivro, dataRequisição e dataDevolução) e identificação Book/Autor, como ilustrado no diagrama da Figura 4.2.

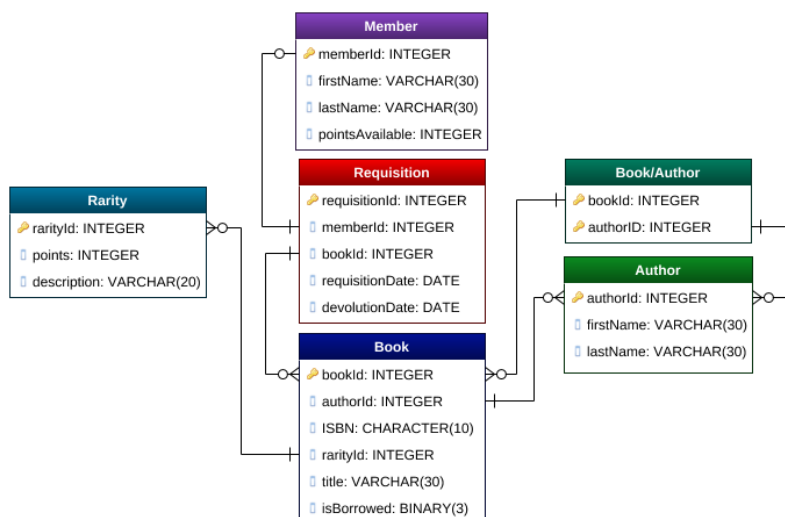


Figura 4.2: Diagrama da BD da biblioteca RINTE

Sendo possível visualizar os dados intrínsecos a cada tabela através da Figura 4.3. De referir que estes dados são de cariz meramente exemplificativos, não violando assim qualquer diretiva do Regulamento Geral sobre a Proteção de Dados.

authorid	firstName	lastName
1	Miguel	Cervantes
2	Antoine	Saint-Exupéry
3	Lewis	Carroll
4	Fernando	Pessoa
5	José	Saramago
6	Eça	Queirós
7	Gil	Vicente
8	Luís	Camões
9	Floribela	Espanca
10	Miguel	Torga

(a) Autor

bookid	authorid	ISBN	requisitionid	title	isBorrowed
1	1	900000000	4	Don Quixote	False
2	1	900000001	2	La Golem	False
3	2	900000002	3	The Little Prince	False
4	2	900000003	4	Night Flight	False
5	3	900000004	5	Underneath	False
6	3	900000005	1	Alice in Wonderland	False
7	4	900000006	5	Mensagem	False
8	4	900000007	5	The Book of Job	False
9	5	900000008	4	Barbarian and the	False
10	5	900000009	3	Land of Sin	False
11	6	900000010	2	The Cane of Mr. F.	False
12	6	900000011	1	The Mole	False
13	7	900000012	2	Radio da Índia	False
14	7	900000013	3	The Best Place	False
15	8	900000014	1	Skeletons in the	False
16	8	900000015	2	The Lovers	False
17	9	900000016	3	The Making of	False
18	9	900000017	4	Moon in Bloom	False
19	10	900000018	1	The Creation of	False
20	10	900000019	5	Animals	False

(b) Livros

memberid	firstName	lastName	pointsAvailable
1	Lucas	Taveira	10
2	Pedro	Santos	10
3	Maria	Carneiro	10
4	Sara	Lotus	10
5	Ricardo	Organista	10
6	Rita	Galego	10
7	Tiago	Lopes	10
8	Vasco	Figueiredo	10
9	Vitória	Carmen	10
10	Xavier	Cardoso	10

(c) Membro

rarityid	points	description
1	1	Very Common
2	2	Common
3	3	Uncommon
4	4	Rare
5	5	Extremely Rare

(d) Raridade

requisitionid	memberid	bookid	requisitionDate	devolutionDate
1	1	1	2020-10-20	2020-10-30
1038	2	2	2020-10-20	2020-10-30
1039	3	3	2020-10-20	2020-10-30
1040	4	4	2020-10-20	2020-10-30
1041	5	5	2020-10-20	2020-10-30
1042	6	6	2020-10-20	2020-10-30
1043	7	7	2020-10-20	2020-10-30
1044	8	8	2020-10-20	2020-10-30
1045	9	9	2020-10-20	2020-10-30
1046	10	10	2020-10-20	2020-10-30

(e) Requisição

Figura 4.3: Overview dos tabelas de dados da biblioteca RINTE

4.2 Algoritmo

A *posteriori* da criação da base de dados, foram desenvolvidas as SP: uma que simula o processo de requisição e outra de devolução. Sendo de seguida explicado os processos anteriormente referidos, bem como os seus algoritmos.

Stored Procedure - Requisição

O processo de requisição consiste em alterar os valores das variáveis de entrada @idMembro, @idLivro e de saída, @sucesso e @detalhe, consoante as variáveis condicionais @emprestado, @nPontosLivro, @nPontoUsados, @nPontoMaximo e @existe, como ilustrado respetivamente nas Figuras 4.4 (a) e (b).

```
USE [libraryDB]
GO
/***** Object: StoredProcedure [dbo].[RequisicaoLivroPessoa]    Script Date: 11/11/2020 11:14:20 *****/
SET ANSI_NULLS ON
SET QUOTED_IDENTIFIER ON
-- Select statement that uses WHERE clause = null returns zero rows even if there are null values in column_name
GO
SET QUOTED_IDENTIFIER ON
-- Identifiers can be delimited by double quotation marks, and literals must be delimited by single quotation marks
GO

-- =====
-- Author:      @author, Alexandre
-- Create date: 11/11/2020 11:14:20
-- Description: Description, orderbookstoreprocedure
-- =====
ALTER PROCEDURE [dbo].[RequisicaoLivroPessoa]
    -- Input/output variables fed into the conditions
    @memberID int,
    @bookID int,
    @success BIT OUTPUT, -- 0: if failure, 1: if success,
    @detail varchar(50) OUTPUT -- If failure show detail
AS
BEGIN
    -- Declare condition variables
    DECLARE @isBorrowed BIT
    DECLARE @nPontosLivro int
    DECLARE @nPontosUsados int
    DECLARE @nPontoMaximo int
    DECLARE @exist BIT
    SET NOCOUNT ON -- SET NOCOUNT ON added to prevent extra result sets from interfering with SELECT statements.
END
```

(a) Entrada e Saída

(b) Condicionantes

Figura 4.4: Variáveis participantes na SP de requisição

Contudo, antes de alterar estes valores é necessário submeter as variáveis de entrada a um conjunto de testes, tais como: a existência do livro solicitado (Figura 4.5(a)), a sua disponibilidade (Figura 4.5(b)), existência do membro (Figura 4.5(c)) e se dispõe de pontos suficientes para requisitar o livro (Figura 4.5(d)).

```
-- Check the book existence
IF (@isBorrowed IS NULL)
BEGIN
    SET @success=0
    SET @Detail='The book does not exist';
    RETURN
END

-- Check if the book was borrowed
IF @isBorrowed = 1
BEGIN
    SET @success=0
    SET @Detail='Book not available in the library'
    RETURN
END

-- Check the member existence with the exist SQL operator
IF (@exist=0)
BEGIN
    SET @success=0
    SET @Detail='The member does not exist';
    RETURN
END

-- Check if the member has enough points to order the book
if (@nPontoUsados+@nPontosLivro) > @nPontoMaximo
BEGIN
    SET @success=0
    SET @Detail='You do not have enough points to order this book'
    RETURN
END
```

(a) Existência do livro

(b) Disponibilidade do livro

(c) Existência do membro

(d) Pontos suficientes

Figura 4.5: Testes de validade dos dados de entrada

Por fim, a Figura 4.6 demonstra que se os requisitos anteriores forem cumpridos, serão introduzidos na tabela de requisições os valores do idMembro, do idLivro, da dataRequisição e do detalhe. Assim como a sua disponibilidade é alterada para o estado emprestado.

```

-- Insert the requisition data to the tblRequisition
INSERT INTO [dbo].[tblRequisition]
    ([memberId]
    ,[bookId]
    ,[requisitionDate]
    ,[devolutionDate])
VALUES
    (@memberId,
    @bookId,
    CAST( GETDATE() AS Date ), -- Converts the value of GETDATE() into a Date datatype
    null)

-- Update the state of the book to borrowed
UPDATE [dbo].[tblBook]
    SET [isBorrowed] = 1
WHERE [dbo].[tblBook].bookId=@bookId

-- Retrieve success message
SET @success=1
SET @Detalhe='Success'

```

Figura 4.6: Resultados obtidos através do sucesso da SP de requisição

Stored Procedure - Devolução

Por outro lado, no processo de devolução são necessários as mesmas variáveis de entrada/saída e condicionantes, a exceção das condicionantes relativas ao número de pontos. Seguindo esta linha de pensamento, apenas os testes concernentes as pontuações e a disponibilidade do livro diferem do processo de requisição. No caso de ser bem sucedido, a tabela de requisições é atualizada e o livro passa a estar disponível. Caso contrário, o utilizador irá ser notificado com uma mensagem de falha.

```

-- TRY/CATCH statement: tries and specifies the process of devolution and throws an exception in case of failure
BEGIN TRY

-- Set the devolutionDate in tblRequisition
UPDATE [dbo].[tblRequisition]
    SET [devolutionDate] = CAST( GETDATE() AS Date )
    WHERE
        [memberId] = @memberId and
        [bookId] = @bookId and
        [requisitionDate] is not null and
        [devolutionDate] is null

-- UPDATE the isBorrowed state to available in tblRequisition
UPDATE [dbo].[tblBook]
    SET [isBorrowed] = 0
    WHERE [dbo].[tblBook].bookId=@bookId

-- Return to the interface a success message
SET @success=1
SET @Detalhe = 'Success'
END TRY

-- Return to the interface a failure message
BEGIN CATCH
    SET @success=0
    SET @Detalhe = 'Failure'
END CATCH;
END

```

Figura 4.7: Resultados obtidos através do sucesso da SP de devolução

4.3 Interface Gráfica

Posteriormente, a BD foi interligada com a *Web App*, sendo possível visualizar a sua página inicial na Figura 4.8.

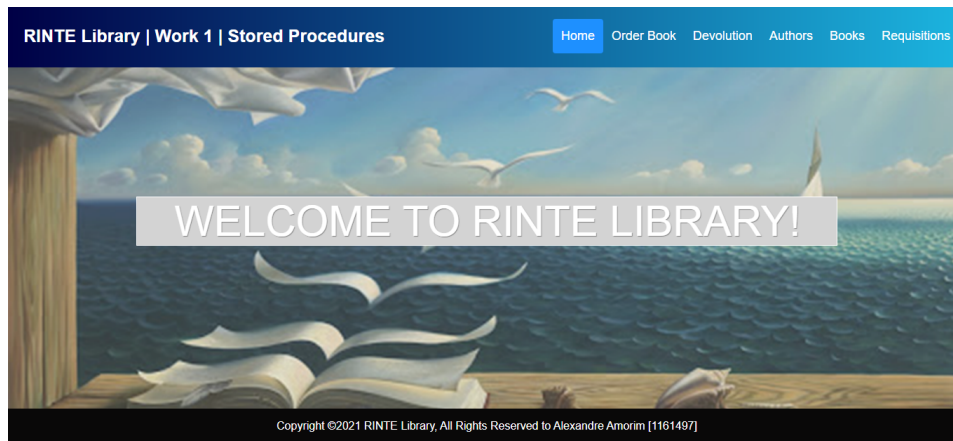


Figura 4.8: Página inicial da biblioteca RINTE

Posteriormente, é possível observar a interface de requisição e devolução de livros exemplificando cenários reais, respetivamente na Figura 4.9(a) e 4.9(b).

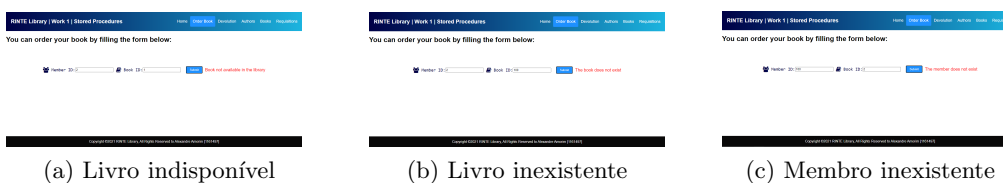


(a) Interface da requisição de livros

(b) Interface da devolução de livros

Figura 4.9: Interface dos processos realizados com SP

Sendo as múltiplas mensagens de erro anteriormente explicadas, ilustradas nas Figuras 4.10.



(a) Livro indisponível

(b) Livro inexistente

(c) Membro inexistente

Figura 4.10: Mensagens de erros

Adicionalmente, foram criadas interfaces para se poderem visualizar os autores (Figura 4.11(a)) e livros (Figura 4.11(b)) que se encontram na BD. De modo a mitigar a necessidade de visualizar os mesmos através do SSMS.

RINTE Library | Work 1 | Stored Procedures Home Order Book Devolution Authors Books Requisitions

You can check the authors list below:

authorId	firstName	lastName
1	Nogueira	Cervantes
2	Andr�e	Saint-Exup�ry
3	Lewis	Carroll
4	Fernando	Pessoa
5	Jo�o	Saramago
6	E�a	Guerilla
7	Gi�	Vomito
8	Luis	Cardoso
9	F�rula	Esquiv�s
10	Miguel	Torga

Copyright  2021 RINTE Library. All Rights Reserved to Alexandre Amorim [1161497]

RINTE Library | Work 1 | Stored Procedures Home Order Book Devolution Authors Books Requisitions

You can check the book list below:

bookId	authorId	ISBN	rarityId	title	isBorrowed
1	1	9000000000	4	Don Quixote	Yes
2	1	9000000001	2	Le G�n�ral	
3	2	9000000002	3	The Little Prince	
4	2	9000000003	4	Night Flight	
5	3	9000000004	5	Jabberwocky	Yes
6	3	9000000005	1	Alice in the Wonderland	Yes
7	4	9000000006	5	Message	
8	4	9000000007	5	The Book of Disquiet	
9	5	9000000008	4	Saltwater and Diamonds	
10	5	9000000009	3	Land of Sin	
11	6	9000000010	2	The G�n�ral of Earth & Air	

Copyright  2021 RINTE Library. All Rights Reserved to Alexandre Amorim [1161497]

(a) Interface dos autores


(b) Interface dos livros

Figura 4.11: Interface dos dados principais da BD

Ademais, tamb m   poss vel que o utilizador analise os livros que disp e atualmente, atrav s da interface da Figura 4.12.

RINTE Library | Work 1 | Stored Procedures Home Order Book Devolution Authors Books Requisitions

You can confirm the books your have order below:

 Member ID:

bookId	authorId	ISBN	rarityId	title	isBorrowed
5	3	9000000004	5	Jabberwocky	true
4	2	9000000003	4	Night Flight	true
6	3	9000000005	1	Alice in the Wonderland	true

Copyright  2021 RINTE Library. All Rights Reserved to Alexandre Amorim [1161497]

Figura 4.12: Interface dos livros que o utilizador disp e atualmente

4.4 Testes n o funcionais

Ulteriormente, foram realizados testes n o funcionais de modo a avaliar a performance da aplica  o, como apresentado na Tabela 4.1. Como expet vel, a p gina inicial possui a maior carga de dados, devido a possuir uma imagem de fundo. Ademais, os tempos de resposta s o bastante semelhantes dado a pequena diferen a da ordem de grandeza de dados que cada p gina requer. Por fim,   poss vel concluir que a carga de dados da p gina de requisico es   regida pela seguinte equa  o:

$$c = 203,6 + 162 \times n \quad (4.1)$$

sendo n o número de livros emprestados ao utilizador em causa.

Tabela 4.1: Resumo dos resultados obtidos para os testes não funcionais

Nome da Página	Carga de Dados (kB)	Tempo de Resposta (ms)
Home	923,2	11
Borrow Book	202,9	8
Borrow Book*	202,9 + 134B	12
Return Book	202,9	8
Return Book*	202,9 + 92B	12
Authors	94,5	7
Books	96,3	7
Requisitions 0	203,6	8
Requisitions 1	203,6 + 162B	12

* após efetuar a respetiva operação para um determinado livro

4.5 Sumário

Neste capítulo, demonstrou-se qual a *pipeline* desenvolvida para levar a avante esta *Web App* exemplificativa, bem como o seu diagrama de base de dados, os seus algoritmos e a sua respetiva interface. Ademais, foram realizados testes não funcionais que permitiram entender a *performance* da mesma.

Capítulo 5

Conclusão

Posteriormente a realização do trabalho em causa, foi possível concluir diversos aspetos relativos ao ecossistema das bases de dados relacionais, tais como as suas diferentes plataformas, os seus rankings de popularidade e os seus tipos de *Stored Programs*. Sendo o tipo principal e o tema deste trabalho, as *Stored Procedures*.

Primeiramente, explicou-se em que consistem as SP. Isto é, que são um conjunto de instruções SQL reutilizáveis executadas a pedido. Sendo que podem aceitar vários parâmetros de entrada e saída, de modo que a tarefa seja efetuada, de acordo com a necessidade do programador.

Posteriormente, contextualizou-se a sua relevância no contexto de RINTE. Em que se verificou que as SP são frequentemente empregues para executar conjuntos de operações SQL, necessários para o funcionamento dos serviços das *Web Apps*. Além disso, foi explicado como este processo ocorre.

Relativamente as suas vantagens e desvantagens, destacam-se a sua celeridade (pré-compilado) e segurança (restrição dos dados que os utilizadores tem acesso), e em contrapartida a sua complexidade (requerem mais conhecimentos técnicos).

Ademais, foi analisada a sua sintaxe comparativamente a códigos SQL diretos. Em que apesar de necessitarem de um maior número de linhas de código para a sua concepção, acabam por serem executados com um comando que ocupa apenas uma linha de código denominada de EXECUTE.

Adicionalmente, foram descritos os *softwares* empregues para a aplicação das SP, que foram o servidor SQL e o SSMS. O processo é bastante direto, contudo requer especial atenção no momento de iniciar o SSMS. Isto pois, o nome do servidor necessita de ser coerente com o sistema onde é operado.

Ulteriormente, desenvolveu-se um exemplo ilustrativo denominado de biblioteca RINTE, que se resume em três etapas: criação da base de dados, desenvolvimento das

SP e por fim a sua integração com a *Web App* concebida. Tendo sido desenvolvidas duas SP uma de requisição e outra de devolução, que foram submetidas a testes não funcionais, de modo a avaliar a sua performance.

Deste modo, verificou-se que todos os objetivos e pressupostos delineados inicialmente foram cumpridos. Ademais, obteve-se uma visão holística das SP e consciência da sua importância acrescida no contexto de RINTE.

Referências Bibliográficas

- [1] G. Harrison and S. Feuerstein, *MySQL Stored Procedure Programming*. O'Reilly Media, 2006. [citado na p. 3, 4, 5]
- [2] J. Deep, *SQL for Beginners: The Practice Guide to Learn SQL in 1 Day + 10 Tips + Exercises, Projects, and Applications*. O'Reilly Media, 2020. [citado na p. 3]
- [3] Oracle, "Advantages of stored procedures." https://docs.oracle.com/cd/F49540_01/DOC/java.815/a64686/01_intr3.htm [Acedido a 14 de Outubro 2020]. [citado na p. 4]
- [4] B. Forta, *SQL in 10 Minutes a Day, Sams Teach Yourself*. Sams Publishing, 5th ed., 2020. [citado na p. 4, 5]

