

Relatório Leitor QR-code



José Moura – 12917 – josemouracoc@gmail.com

João Pereirinha – 11201 – 11201@ismt.pt

Curso – Licenciatura em Informática Desenvolvimento de Software



Conteúdo

Resumo	2
1 Introdução	3
2 Funcionalidades	4
3 Aplicação	5
4 Testes	8
5 Conclusão	10
Link do Git	11

Resumo

Este relatório descreve o desenvolvimento de uma aplicação que utiliza uma interface gráfica para ler e armazenar informações de QR-Code em imagens de ficheiros. A aplicação permite ao utilizador abrir ficheiros de imagem, extrair os detalhes do código QR e guardar as informações no formato CSV. Utilizamos a biblioteca ZXing para a decodificação dos QR-Codes e a biblioteca Java Swing para a construção da interface gráfica.

Capítulo 1

Introdução

O objetivo deste projeto é desenvolver uma aplicação que possa usar uma interface gráfica para ler e armazenar informações de QR-Code em imagens de ficheiros de imagem. O aplicativo deve permitir que o utilizador abra o ficheiro de imagem e extraia os detalhes do código QR.

Para que o utilizador verifique as informações que serão guardadas no formato CSV, que incluem o nome do arquivo, a data-hora e os dados extraídos, a leitura dos QR Codes é exibida em uma tabela.

Para criar esta aplicação, utilizamos ferramentas e bibliotecas que permitam a leitura e processamento de arquivos de imagem, como a biblioteca ZXing para decodificação de QR-Codes.

Foi criada uma interface gráfica amigável para que o utilizador possa interagir com a aplicação e realizar as tarefas desejadas de forma simples e intuitiva.

Capítulo 2

Funcionalidades

As funcionalidades desenvolvidas neste projeto incluem:

1. Leitura de ficheiro de imagem QR-Code;
2. Detecção de erros;
3. Apresentação dos dados extraídos ao utilizador;
4. Exportação de dados para o formato CSV, contendo informação do nome do ficheiro de imagem QR Code, data-hora e dados extraídos da imagem QR Code;
5. Capacidade para guardar informação de vários QR Codes para o ficheiro CSV numa única operação.

Capítulo 3

Aplicação

A aplicação foi implementada utilizando a linguagem de programação Java e a biblioteca ZXing para a leitura e decodificação dos QR-Codes. Para usar a biblioteca ZXing, necessitamos de fazer o download de:

- `core-3.5.3.jar`;
- `javase-3.5.3.jar`;
- `postgresql-42.7.3.jar`.

Após termos o necessário para usar esta biblioteca, juntamos estes 3 ficheiros a uma pasta chamada `lib` na pasta do projeto e ficamos com acesso à biblioteca por parte do nosso código.

De seguida, desenvolvemos a interface gráfica utilizando a biblioteca Java Swing, com a adição de elementos como botões, caixas de texto e tabelas. A nossa aplicação ficou com o seguinte aspeto:

Para a leitura dos ficheiros de imagem e seleção do CSV, foi implementado um sistema de seleção de ficheiros utilizando a classe `JFileChooser` da biblioteca Swing. Uma vez selecionado o ficheiro de imagem, a aplicação faz uso da biblioteca ZXing para decodificar o QR-Code e extrair os dados contidos no código. No caso da escolha da seleção do ficheiro CSV, utilizamos a função do `PrintWriter`, que é um tipo de `FileWriter`, para guardar a informação no ficheiro CSV.

Agora apresentamos alguns erros que podem acontecer quando Ao selecionar um imagem que não seja QRCode aparece a seguinte mensagem:

Ao selecionar um imagem QRCode aparece a informação no excel em csv:

Ao atingir o limite definido vai aparecer este erro:

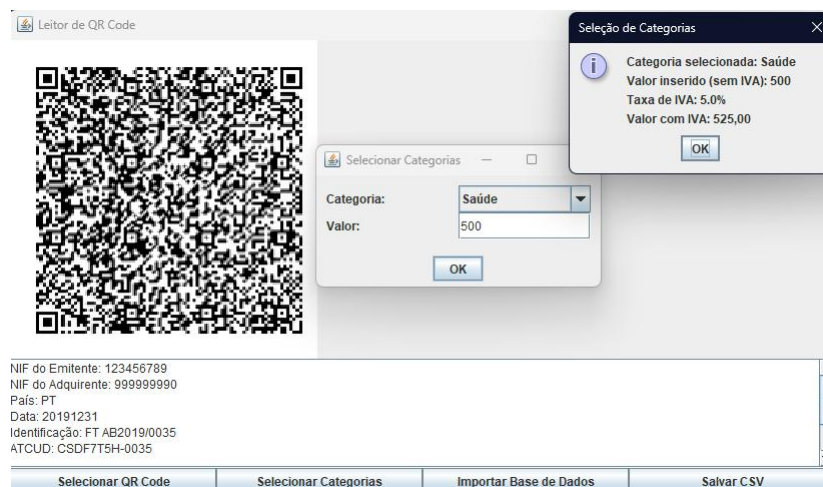


Figura 3.1: Interface gráfica da aplicação

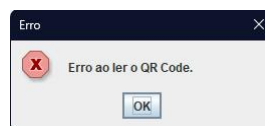


Figura 3.2: Mensagem de erro ao selecionar uma imagem não QR Code



Figura 3.3: Informações do QR Code em CSV



Figura 3.4: Erro ao atingir limite definido

```
org.postgresql.util.PSQLException: ERROR: new row for relation "categorias" violates check constraint "teto_despesa_check"
Detail: Failing row contains (1, Supermercado, 1000.26, 500.00).
Where: SQL statement "UPDATE categorias
SET total_despesas = total_despesas + NEW.valor
WHERE categorias.id = NEW.id_categoria"
```

Figura 3.5: Exemplo de saída no terminal

E no terminal aparece assim:

Os dados extraídos são apresentados ao utilizador numa tabela na interface gráfica, permitindo que o utilizador visualize facilmente os dados.

Para exportar os dados para um arquivo CSV, foi implementado um sistema de escrita em arquivo utilizando a classe `PrintWriter`.

Os dados extraídos de todos os ficheiros são armazenados em uma única tabela e podem ser exportados para um único arquivo CSV. O CSV guardado fica desta maneira:



```
123.csv > data
1  nifemidente;nifadquirente;pais;data;identificacao;atcud;categoria;valorsemlva;taxa;valorcomiva
2  123456789;999999999;PT;20191231;FT AB2019/0035;CSDFT5H-0035;Supermercado;1;13.0%;999,13;
3  |
```

Figura 3.6: Exemplo de arquivo CSV gerado

Capítulo 4

Testes

Neste teste, será feita a leitura e implementação de um código QR de uma fatura médica de 400€.



Figura 4.1: Leitor de QR Code em ação

Data OutputMessagesNotifications

	id [PK] integer	id_categoria integer	valor numeric	nif_emitente character varying (100)	nif_adquirente character varying (100)	pais character varying (100)	data character varying (100)	identificacao character varying (100)	atcud character varying (100)
1		8	2	420	123456789	999999990	PT	20191231	FT AB2019/0035 CSDFT5H-0035

Total rows: 1 of 1Query complete 00:00:00.183Ln 1, Col 1

Figura 4.2: Tabela com os dados da fatura

Data OutputMessagesNotifications

	id [PK] integer	nome character varying (100)	total_despesas numeric (10,2)	teto_despesa numeric (10,2)
1	1	Supermercado	0.00	500.00
2	2	Saúde	420	500.00
3	3	Estudos	0.00	500.00

Figura 4.3: Tabela com as categorias

Capítulo 5

Conclusão

Neste trabalho, desenvolvemos uma aplicação em Java para leitura e extração de dados de arquivos de imagem QR-Code. Utilizamos a biblioteca ZXing para decodificação dos códigos e a biblioteca Swing para construção da interface gráfica. Implementamos validação dos dados extraídos e exportação dos dados para um arquivo CSV. A aplicação atendeu aos requisitos propostos e apresentou uma interface gráfica intuitiva e fácil de usar.

Em conclusão, o trabalho foi bem-sucedido na criação de uma solução prática e eficiente para a leitura e extração de dados de arquivos QR-Code, permitindo-nos aprofundar os conhecimentos na parte da criação gráfica da aplicação e deu a conhecer como funcionam os QR Codes e como utilizá-los.

Link do Git

`https://github.com/JoseComSono/qrcodereader`