

## Relatório Final de Estágio Supervisionado

### 1. Identificação

**Estagiário(a):** Isabela Pimentel Loebel

**Disciplina Matriculada :** Estágio Obrigatório

**Título do Projeto:** Implementação de uma ferramenta de relatórios no software Uniplus.

**Quantidade de horas realizadas:** 204h

**Período:** 18/03/2024 à 29/05/2024

### 2. Atividades desenvolvidas no estágio:

#### 2.1. Cronograma de execução e descrição das atividades desenvolvidas

Atividades	2024		
	Março	Abril	Maio
1. Pesquisa Bibliográfica	X		
2. Desenvolvimento – Análise e projeto	X	X	
3. Desenvolvimento – Implementação		X	
4. Testes		X	X
5. Relatórios Parciais		X	X
6. Relatório Final			X

#### Descrição do plano de trabalho e cronograma de execução:

- **Pesquisa Bibliográfica:** Estudo das tecnologias de desenvolvimento a seguir:
  - Banco de dados;
  - UI/UX Design;
  - Jrxml;
  - Jaspersoft Studio IDE.
- **Desenvolvimento – Análise e projeto:** Foi feita a definição dos requisitos do relatório e com este, foi feita a análise da estruturação do relatório e requisitos necessários para busca no banco de dados;
- **Desenvolvimento – Implementação:** Foi iniciado a implementação do relatório com jrxml através da IDE Jaspersoft Studio, inicialmente somente com um modelo simples e posteriormente o modelo definitivo.
- **Testes (verificação e validação):** Execução de testes do relatório desenvolvido: nesta atividade foi feito os testes das funcionalidades do relatório, garantindo os pontos requisitados como filtragem e agrupamentos dos dados para melhor visualização do usuário;
- **Relatórios:** Foi realizado os relatórios parciais e final das atividades realizadas no estágio.

## 2.2. Disciplinas utilizadas

- Computação 1 e 2;
- Banco de dados;
- Introdução à Engenharia de Software;
- Engenharia de software 1 e 2.

## 2.3. Desenvolvimento – Análise e projeto

Para iniciar a implementação deste projeto, foram levantados os requisitos funcionais e não funcionais junto à empresa e ao usuário final. A seguir, estão listados os requisitos identificados:

Requisitos Funcionais:

- **RF01:** A ferramenta para gerar relatórios deve ser desenvolvida em formato JRXML para garantir máxima integração com o software Uniplus.
- **RF02:** A ferramenta deve ser capaz de importar dados do banco de dados do software Uniplus.
- **RF03:** A ferramenta deve aceitar três parâmetros para a geração do relatório: data inicial, data final e grupo de produtos.
- **RF04:** A ferramenta deve ser capaz de gerar relatórios que apresentem todos os produtos com a data de vencimento no período informado quando o grupo de produtos não é especificado.
- **RF05:** A ferramenta deve ser capaz de gerar relatórios que apresentem apenas produtos do grupo especificado quando o grupo de produtos é informado.
- **RF06:** A ferramenta deve ser capaz de exibir a mensagem "Nenhum dado correspondente encontrado" quando nenhum dado é encontrado.

Requisitos Não Funcionais:

- **RNF01:** O relatório deve incluir as seguintes informações: CNPJ da empresa, razão social da empresa, título "Lista de Produtos por Validade", período buscado (período: data inicial à data final), o dia que aquele relatório foi gerado, o grupo do produto, o código do produto, o código de barras (EAN) do produto, a descrição do produto (nome do produto) e a validade do produto, assim como o número de páginas.
- **RNF02:** A ferramenta deve ser compatível com o software Uniplus.
- **RNF03:** A ferramenta deve ser capaz de lidar com grandes volumes de dados do banco de dados do software Uniplus de maneira eficiente.
- **RNF04:** A ferramenta deve ter uma interface de usuário intuitiva e fácil de usar.

## 2.4. Desenvolvimento – Implementação

Para garantir os requisitos RF01, RF02, RNF02 e RNF03, foi utilizada a IDE Jaspersoft Studio como editor de código-fonte para a implementação, onde foi criado um modelo de relatório no formato Jasper Report, conforme ilustrado na Figura 1, visto que é o formato compatível com o mencionado no RF01.

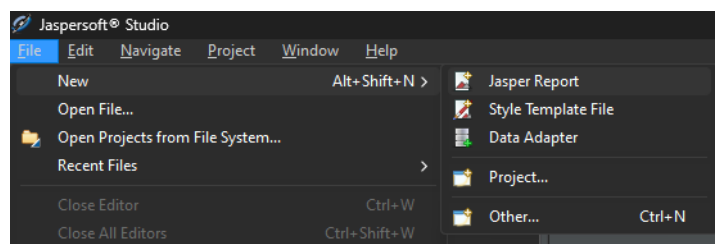


Figura 1 - Criação de um modelo Jasper Report.

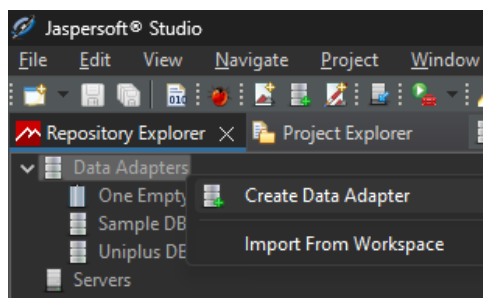


Figura 2 - Criação do adaptador de dados.

Após a criação do arquivo do modelo de relatório, foi necessário desenvolver o adaptador de dados, ilustrado na Figura 2. Este adaptador estabeleceu a conexão com uma base de ensaios disponibilizada pela empresa para testes, composta por um banco de dados PostgreSQL do software Uniplus. A configuração do adaptador incluiu a inserção da URL JDBC, nome do banco de dados a ser consultado, usuário e senha necessários para acesso, conforme ilustrado na Figura 3, garantindo o RF02.

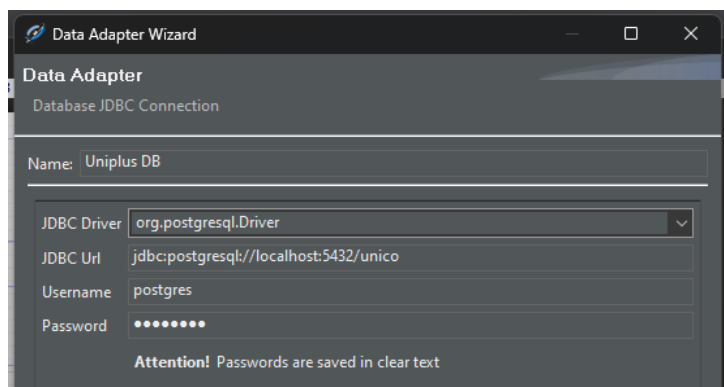


Figura 3 - Configuração do adaptador de dados.

#### 2.4.1. Configuração e Modelagem de Dados

Após a configuração do adaptador, foi necessário moldar a consulta SQL que alimentaria o relatório com os dados do banco de dados do software Uniplus. Foi realizada uma investigação do Modelo Entidade-Relacionamento (M-ER) utilizado pelo software. Concluiu-se que três tabelas estariam envolvidas na consulta: filial, produto e hierarquia.

- Tabela Filial: Contém dados da empresa, como CNPJ e razão social;
- Tabela Hierarquia: Utilizada para retornar o nome do grupo do produto, pois retém informações relacionadas ao grupo;

- Tabela Produto: Utilizada para consultar informações sobre os produtos, incluindo código, código de barras (EAN), nome, grupo e campos adicionais "extra1" a "extra10".

A necessidade dos campos adicionais foi justificada pela empresa AlternativaFoz, devido à flexibilidade oferecida pelo software Uniplus, permitindo personalizações. Observou-se que não havia um campo dedicado para a data de vencimento do produto, levando a empresa AlternativaFoz a utilizar os campos adicionais para essa finalidade. Cada cliente do software utilizava diferentes campos adicionais para armazenar a data de vencimento, necessitando a verificação de todos os campos em busca de valores no formato "DD/MM/AAAA".

#### 2.4.2. Filtragem de Dados e Geração de Relatórios

Após compreender estas condições, iniciou-se a filtragem dos dados para cumprir os requisitos RF03 a RF06 e RNF01, resultando no modelo de relatório demonstrado na Figura 4, que é populado pelo SQL disposto na Figura 5.

O modelo de relatório "Lista de Produtos por Validade" apresenta os seguintes elementos:

- Cabeçalho:**
  - Gerado em: "+new java.util.Date()"
  - Período: new à new
- Filtros:**
  - CNPJ: \$F{cnpj}
  - \$F{razaosocial}
  - GRUPO: \$F{hierarquia}
- Tabela de Dados:**

Código	Código de barras	Descrição do produto	Validade
\$F{codigo}	\$F{ean}	\$F{nomeproduto}	(\$F{extra1} != null &&
- Página:** "Página "+\$V" " + \$V
- Mensagem de Erro:** NENHUM DADO CORRESPONDENTE ENCONTRADO

Figura 4 - Modelo de relatório criado.

```

1 SELECT filial.cnpj,
2       filial.razaosocial,
3       t2.codigo,
4       t2.ean,
5       t2.nome as nomeproduto,
6       t2.extra1 as extra1,
7       t2.extra2 as extra2,
8       t2.extra3 as extra3,
9       t2.extra4 as extra4,
10      t2.extra5 as extra5,
11      t2.extra6 as extra6,
12      t2.extra7 as extra7,
13      t2.extra8 as extra8,
14      t2.extra9 as extra9,
15      t2.extra10 as extra10,
16      hierarquia.nome as hierarquia,
17      hierarquia.id AS idhierarquia
18 FROM filial, produto t2
19 INNER JOIN hierarquia ON (hierarquia.id = t2.idhierarquia)
20 WHERE (
21      TO_DATE(extra1, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
22      OR TO_DATE(extra2, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
23      OR TO_DATE(extra3, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
24      OR TO_DATE(extra4, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
25      OR TO_DATE(extra5, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
26      OR TO_DATE(extra6, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
27      OR TO_DATE(extra7, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
28      OR TO_DATE(extra8, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
29      OR TO_DATE(extra9, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
30      OR TO_DATE(extra10, 'DD-MM-YYYY') between $P{DATA_INICIAL} and $P{DATA_FINAL}
31 )
32 AND ( $P{GRUPOPRODUTO}::text IS NULL
33      OR $P{GRUPOPRODUTO}::text = ''
34      OR t2.idhierarquia = $P{GRUPOPRODUTO}
35 )
36 ORDER BY hierarquia

```

Figura 5 - Consulta SQL construída.

- Figura 5: A construção da consulta SQL garante que os parâmetros de período do Jasper Report (\$P{DATA\_INICIAL} e \$P{DATA\_FINAL}) sejam corretamente interpretados e comparados com cada campo adicional. A verificação do grupo do produto assegura o cumprimento dos requisitos RF04 e RF05.
- No Data Section: A seção "No Data" do relatório foi configurada para exibir a mensagem "Nenhum dado correspondente encontrado" conforme RF06.
- Exibição do Campo de Validade: Definiu-se que, quando houverem múltiplos campos com datas de validade, somente o primeiro campo encontrado será exibido.

## 2.5. Testes e Validação

O modelo de relatório foi compilado e importado no software Uniplus para testes. Após a verificação e validação dos dados pela empresa, concluiu-se que o relatório gerado estava coerente e atendia a todos os requisitos necessários.

Esta abordagem detalhada garantiu a implementação eficiente do projeto, alinhando-se às expectativas dos clientes e aos padrões estabelecidos para a geração de relatórios no contexto do software Uniplus.

## 2.6. Apresentação e discussão sucinta dos principais resultados e produtos gerados

Na reunião final com a empresa, juntamente com o *feedback* de três clientes que utilizaram a ferramenta, concluiu-se que o modelo de relatório desenvolvido foi um instrumento valioso para a tomada de decisões estratégicas e para a obtenção de *insights* de negócio. O relatório gerado demonstrou ser de fácil manuseio, apresentando informações

claras e precisas, além de possibilitar aos clientes um entendimento mais aprofundado sobre seu estoque atual.

Um feedback particularmente significativo veio de um dos usuários da ferramenta. Este cliente observou que, com o auxílio do relatório, identificou um alimento específico próximo ao vencimento, bem como uma bebida. Utilizando essa informação, o cliente pôde criar uma estratégia de venda, montando um "kit festa" que combinava a compra desses dois produtos com um desconto. Esta estratégia resultou em um aumento significativo nas vendas e na redução do estoque de produtos próximos ao vencimento, diminuindo assim o potencial prejuízo para o cliente.

Este exemplo ilustra o impacto positivo do relatório, destacando sua utilidade prática e a capacidade de transformar dados em ações estratégicas concretas, proporcionando benefícios tangíveis para os usuários.

## **2.7. Leitura bibliográfica utilizada**

O treinamento das funcionalidades do software Uniplus foi realizado pela equipe da AlternativaFoz, com o suporte do portfólio de documentação disponibilizado pelo site do software.

INTELIDATA. Central do Uniplus. Sistemas de Gestão Empresarial. 2024. Disponível em: <<https://centraldouniplus.intelidata.inf.br/>>.

## **2.8. Relação dos fatores positivos e negativos durante a execução do projeto**

Durante a fase de elicitação dos casos de uso, observou-se a adversidade representada pelo número excessivo de campos adicionais e pela inflexibilidade do software Uniplus em relação ao Modelo de Entidade-Relacionamento (M-ER) adotado. Esses fatores negativos, embora desafiadores, impuseram a necessidade de desenvolver um modelo de relatório simplificado. Este esforço culminou na criação de um produto precisamente alinhado com as expectativas dos clientes finais.

A análise detalhada de um software proprietário, como o abordado neste estudo, destacou a necessidade de uma projeção de software precisa e coerente. Mesmo sendo um software já consolidado no mercado, foram identificadas várias contrariedades que são discutidas nas disciplinas de Engenharia de Software e Banco de Dados como aspectos a serem evitados. Este processo resultou em um ganho significativo de experiência, permitindo uma compreensão aprofundada das razões pelas quais certos requisitos, como os observados no software em questão, não são geralmente adotados. Essa experiência contribuirá para a melhoria da prática de desenvolvimento de software em projetos futuros.

# **3. Conhecimento Teórico e Prático Exigido**

## **3.1. Conhecimento Teórico e Prático Adquirido no Curso**

- Banco de Dados:
  - Conceitos de Banco de Dados;
  - Modelagem de Dados;
  - SQL.
- Introdução a Engenharia de Software:
  - Ciclo de Vida de Software;

- Boas Práticas de Programação.
- Engenharia de Software I:
  - Análise de Requisitos de Software;
  - Modelagem de Sistemas;
  - Análise e Projeto Orientado a Objetos;
  - Introdução a Verificação e Validação de Software.
- Engenharia de Software II:
  - Arquitetura de Software;
  - Qualidade de Software.
- Computação I e II:
  - Programação;
  - Paradigma de programação orientada a objetos;
  - Linguagem Java;
  - Tópicos Avançados.

### 3.2. Conhecimento Teórico e Prático Adquirido no Estágio

- Jrxml:
  - Criação e edição de arquivos Jrxml para a geração de relatórios personalizados;
  - Compreensão da estrutura XML utilizada pelo JasperReports para definir layouts de relatórios;
  - Integração de Jrxml com outras tecnologias para a personalização de relatórios conforme as necessidades do projeto;
- Utilização de Java com Jrxml:
  - Implementação de funcionalidades em Java para manipulação e processamento de dados a serem exibidos nos relatórios;
  - Integração de bibliotecas Java com JasperReports para a geração e exportação de relatórios em diversos formatos (PDF, Excel, entre outros).
- Desenvolvimento de Relatórios:
  - Análise e definição de requisitos para relatórios conforme a necessidade dos stakeholders;
  - Criação de *templates* reutilizáveis para diferentes tipos de relatórios;
- Aprofundamento de todos os tópicos já vistos no curso.

Foz do Iguaçu, 29 de maio de 2024.

---

**Assinatura do Estagiário**  
Isabela Pimentel Loebel

---

**Assinatura do Professor  
Orientador de Estágio**  
Camile Frazão Bordini

---

**Assinatura do Supervisor  
Técnico**  
Analú Gabriela Costa de  
Medeiros