



**UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO**  
**CENTRO TECNOLÓGICO**  
**COLEGIADO DO CURSO DE CIÊNCIA DA COMPUTAÇÃO**

Matheus De Oliveira Lima

## **Título do Trabalho**

Vitória, ES

2025

Matheus De Oliveira Lima

## **Título do Trabalho**

Anteprojeto apresentado ao Colegiado do Curso de Ciência da Computação do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito para aprovação na Disciplina Projeto de Graduação I.

Universidade Federal do Espírito Santo – UFES

Centro Tecnológico

Colegiado do Curso de Ciência da Computação

Orientador: Prof. Dr. Vítor Estêvão Silva Souza

Coorientador: Prof. Dr. Ciclano de Tal

Vitória, ES

2025

# 1 Introdução

Ao observar os períodos de entrada de estudantes (início de cada semestre estudantil), fazer os processos de alocação de professores a cargos internos, colocar estudantes de Graduação e Pós-Graduação em disciplinas, entre outras tarefas administrativas, se torna um processo muito trabalhoso manualmente; Com intuito de facilitar esse processo está sendo desenvolvido o Marvin, para gerenciar tarefas de ensino e pesquisa dentro da universidade.

Este trabalho tem como foco melhorar a interface do Sistema Marvin, que atualmente funciona com CRUD's padrões e pouco adaptados aos diferentes tipos de usuários. Esse sistema foi criado para ajudar na gestão de tarefas de ensino e pesquisa dentro da universidade, mas hoje apresenta páginas padronizadas, sem diferenciação para os distintos papéis de usuários (como professores, coordenadores, alunos, entre outros).

## 1.1 Motivação e Justificativa

A motivação para a escolha deste tema foi o interesse pessoal em atuar na área de desenvolvimento web. Desde quando as disciplinas começaram a ficar mais específicas eu comecei a me identificar mais com desenvolvimento web, e essa afinidade influenciou diretamente na decisão de buscar um projeto que permitisse aplicar na prática os conhecimentos adquiridos ao longo da graduação. Além disso, ter a oportunidade de contribuir para uma aplicação que tem um potencial alto de ser usado dentro da universidade trouxe uma motivação a mais para participar.

A justificativa para esse trabalho está na necessidade de melhorar a experiência dos usuários do Sistema Marvin. Atualmente, ele funciona com páginas genéricas baseadas em operação CRUD, sem levar em conta o papel específico que cada usuário exerce dentro da instituição. Isso acaba tornando o uso do sistema mais confuso e pouco eficiente, especialmente para quem não tem familiaridade com o funcionamento interno das atividades administrativas e acadêmicas.

Desenvolver interfaces personalizadas, que se adaptem ao perfil de cada usuário, pode aumentar a usabilidade do sistema e agilizar tarefas rotineiras. Os trabalhos anteriores deram mais ênfase e esforço nas funcionalidades dentro do sistema administrativo, mas também é importante a experiência do usuário e a adaptação do sistema ao contexto real de uso. Por isso, é relevante continuar pesquisando e desenvolvendo soluções que não apenas funcionem tecnicamente, mas que também sejam intuitivas e úteis no dia a dia dos usuários.

## 1.2 Objetivos

O **objetivo geral** é desenvolver interfaces personalizadas no sistema Marvin, adaptando as páginas de acordo com o papel (Role) de cada usuário, com o objetivo de tornar a navegação mais intuitiva e funcional para diferentes perfis dentro da universidade.

## 1.3 Método de Desenvolvimento do Trabalho

O desenvolvimento deste trabalho será feito em etapas, seguindo uma abordagem prática baseada no processo de engenharia de software, com foco em desenvolvimento web. A metodologia utilizada envolve análise dos requisitos, análise dos papéis dos usuários, planejamento das interfaces, implementação e validação.

Inicialmente, será feita uma análise do sistema Marvin, com foco no subsistema administrativo, para entender como os diferentes tipos de usuários (como administradores, gestores de departamento, coordenadores de curso e programas de pós-graduação) interagem com as funcionalidades existentes. A partir disso, serão identificadas as principais funções associadas a cada papel e os pontos em que a interface atual não atende bem às necessidades específicas desses perfis.

Com base nessa análise, serão projetadas novas interfaces personalizadas, utilizando ferramentas modernas de desenvolvimento frontend. A implementação será feita por meio de tecnologias web já utilizadas no projeto Marvin, respeitando a estrutura já existente e buscando garantir compatibilidade com o sistema atual.

Por fim, será realizada uma etapa de testes com usuários reais ou simulações de uso, para avaliar a usabilidade das novas interfaces. A documentação do processo completo, incluindo decisões técnicas e dificuldades encontradas, também fará parte do trabalho final.

## 1.4 Cronograma

O cronograma de execução apresenta a distribuição no tempo das atividades que serão realizadas ao longo do desenvolvimento do trabalho. As atividades foram definidas com base nos objetivos específicos apresentados na Subseção 1.2, garantindo que cada uma delas contribua diretamente para a obtenção do objetivo geral.

- **Atividade 1: Levantamento de requisitos e análise de papéis do sistema**  
Analisar o sistema Marvin, especialmente o subsistema administrativo, e identificar os diferentes tipos de usuários e suas respectivas funções.
- **Atividade 2: Avaliação da usabilidade atual**

Estudar as interfaces existentes no sistema e levantar as principais dificuldades enfrentadas por cada perfil de usuário.

- **Atividade 3: Projeto das interfaces personalizadas**

Definir a estrutura, layout e fluxo de navegação das novas interfaces específicas para cada Role.

- **Atividade 4: Implementação das interfaces - Parte 1**

Desenvolver e integrar as interfaces personalizadas de 1 ou 2 papéis principais de usuário, como Administrador e Gestor de Curso.

- **Atividade 5: Implementação das interfaces - Parte 2**

Finalizar o desenvolvimento das interfaces dos demais papéis e garantir a integração com o sistema existente.

- **Atividade 6: Testes, validação e documentação**

Realizar testes com usuários (ou simulações), ajustar problemas encontrados e documentar todo o processo de desenvolvimento.

Tabela 1 – Cronograma de execução das atividades

	Mês 1	Mês 2	Mês 3	Mês 4	Mês 5	Mês 6
<b>Atividade 1</b>	X					
<b>Atividade 2</b>		X				
<b>Atividade 3</b>			X			
<b>Atividade 4</b>			X	X		
<b>Atividade 5</b>				X	X	
<b>Atividade 6</b>						X

## 2 Fundamentação Teórica e Tecnologias Utilizadas

A seção de **Fundamentação Teórica e Tecnologias Utilizadas** deve apresentar os aspectos relativos ao conteúdo de disciplinas cursadas, ao conteúdo do material bibliográfico consultado e às tecnologias que irão auxiliar o desenvolvimento do trabalho em busca de uma solução para o problema abordado. Deve ser apresentada em forma de texto, **entre 8 e 15 páginas**, que demonstre conhecimento básico da literatura técnico-científica sobre o tema abordado.

Utilizaremos este capítulo para apresentar alguns exemplos de uso de  $\text{\LaTeX}$  que podem ser úteis para aqueles que possuem pouca experiência com a ferramenta e vão escrever o Anteprojeto e, futuramente, a monografia usando  $\text{\LaTeX}$ . Para mais informações sobre  $\text{\LaTeX}$ , você pode consultar a [documentação do overleaf](#) ou vários materiais disponíveis online, como [esse minicurso da USP](#).

### 2.1 Seções e Subseções

O documento é organizado em capítulos (`\chapter{}`), seções (`\section{}`), subseções (`\subsection{}`), sub-subseções (`\subsubsection{}`) e assim por diante. Atenção, porém, a não criar estruturas muito profundas (sub-sub-sub-...) pois o documento não fica bem estruturado.

#### 2.1.1 Referências a seções

Cada parte do documento (capítulo, seção, etc.) deve possuir um rótulo logo abaixo de sua definição. Por exemplo, este capítulo é definido com `\chapter{Introdução}` seguido por `\label{sec-fundteo}`. Assim, podemos fazer referências cruzadas usando o comando `\ref{rótulo}`: “O Capítulo 2 começa com a Seção 2.1, que é ainda subdividida nas subseções 2.1.1 e 2.1.2.

Para melhor organização das partes do documento, sugere-se primeiro utilizar o prefixo `sec-` (para diferenciar de referências à figuras, tabelas, etc. quando usarmos o comando `\ref{}`) e também representar a hierarquia das seções nos rótulos. Por exemplo, o Capítulo 2 tem rótulo `sec-fundteo`, sua Seção 2.1 tem rótulo `sec-fundteo-secoes` e a Subseção 2.1.1 tem rótulo `sec-fundteo-secoes-refs`.

### 2.1.2 Sobre referências cruzadas

Nas próximas seções, veremos que é possível fazer referência cruzada não só a seções mas também a listagens de código, figuras, tabelas, etc. Em todos estes casos, quando nos referimos à Seção X, Listagem Y ou Figura Z, consideramos que estes são os nomes próprios destes elementos e, portanto, usa-se a primeira letra maiúscula. Isso pode ser visto na Subseção 2.1.1, acima. A exceção é quando nos referimos a vários elementos ao mesmo tempo, por exemplo: “as subseções 2.1.1 e 2.1.2”.

Por fim, ao usar o comando `\ref{}`, sugere-se separá-lo da palavra que vem antes dele com um `~` ao invés de espaço. Por exemplo: o `Capítulo~\ref{sec-fundteo}`. Isso faz com que o  $\text{\LaTeX}$  não quebre linha entre a palavra capítulo e o número do capítulo.

## 2.2 Citações Bibliográficas

Este documento utiliza a ferramenta de gerenciamento de referências bibliográficas do  $\text{\LaTeX}$ , chamada *BibTeX*. O arquivo `bibliografia.bib`, referenciado no arquivo  $\text{\LaTeX}$  principal deste documento, contém algumas referências bibliográficas de exemplo. Assim como capítulos, seções, etc., tais referências também possuem rótulos, especificados como primeiro parâmetro de cada entrada (ex.: `@incollection{albert1999internet, ...}`).

Sugere-se um padrão para rótulos de referências bibliográficas para que fique claro também no código  $\text{\LaTeX}$  qual referência está sendo citada. Por exemplo, ao citar a referência `dijkstra1959note`, sabemos que é um artigo escrito por *Souza* e outros, publicado no *SESAS* em *2013* (geralmente a pessoa que citou sabe que publicação é *SESAS* e quem é *Souza*).

Para citar uma referência bibliográfica contida no arquivo *BibTeX*, basta usar seu rótulo como parâmetro de um de dois comandos possíveis de citação:

- O comando `\cite{}` efetua uma citação tradicional, colocando o nome do(s) autor(es) e o ano entre parênteses. Por exemplo, `\cite{albert1999internet}` é transformado em (ALBERT; JEONG; BARABÁSI, 1999);
- O comando `\citeonline{}` efetua uma citação integrada ao texto, colocando o nome do(s) autor(es) direto no texto e somente o ano entre parênteses. Por exemplo, “de acordo com `\citeonline{albert1999internet}`” é transformado em: de acordo com Albert, Jeong e Barabási (1999);

Também é possível citar vários trabalhos de uma só vez, separando os rótulos das referências bibliográficas com uma vírgula dentro do comando apropriado. Por exemplo, `\cite{dijkstra1959note,lattes1947observations}` (DIJKSTRA, 1959; LATTES; OCCHIALINI; POWELL, 1947).

Os trabalhos citados são automaticamente incluídos na seção de referências bibliográficas, ao final do documento. Tudo é formatado automaticamente segundo padrões da ABNT.

## 2.3 Listagens de Código

O pacote `listings`, incluído neste template, permite a inclusão de listagens de código. Análogo ao já feito anteriormente, listagens possuem rótulos para que possam ser referenciadas e sugerimos uma regra de nomenclatura para tais rótulos: usar como prefixo o rótulo do capítulo, substituindo `sec-` por `lst-`.

A Listagem 2.1, por exemplo, possui o rótulo `lst-intro-exemplo` e representa o código que foi usado no próprio documento para exibir as listagens desta seção. Como podemos ver, a sugestão é que os arquivos de código sejam colocados dentro da pasta `codigos/` e tenham nome idêntico ao rótulo, colocando a extensão adequada ao tipo de código.

Listagem 2.1 – Exemplo de código L<sup>A</sup>T<sub>E</sub>X para inclusão de listagens de código.

```
1 \lstinputlisting[label=lst-intro-exemplo, caption=Exemplo de código \latex para
   inclusão de listagens de código., float=htpb]{codigos/lst-intro-exemplo.tex}
2
3 \lstinputlisting[label=lst-intro-outror exemplo, caption=Exemplo de código \java
   especificando linguagem utilizada., language=Java]{codigos/lst-intro-
   outror exemplo.java}
```

A Listagem 2.2 mostra um exemplo de listagem com especificação da linguagem utilizada no código. O pacote `listings` reconhece algumas linguagens<sup>1</sup> e faz “coloração” de código (na verdade, usa **negrito** e não cores) de acordo com a linguagem. O parâmetro `float=htpb` incluído em ambos os exemplos impede que a listagem seja quebrada em diferentes páginas.

Listagem 2.2 – Exemplo de código Java<sup>TM</sup> especificando linguagem utilizada.

```
1 public class HelloWorld {
2     public static void main(String[] args) {
3         System.out.println("Hello , World!");
4     }
5 }
```

<sup>1</sup> Veja a lista de linguagens suportadas em [http://en.wikibooks.org/wiki/LaTeX/Source\\_Code\\_Listings#Supported\\_languages](http://en.wikibooks.org/wiki/LaTeX/Source_Code_Listings#Supported_languages).



## 2.4 Figuras

Figuras podem ser inseridas no documento usando o ambiente `figure` (ou seja, `\begin{figure}` e `\end{figure}`) e o comando `\includegraphics{}`. Existem alguns outros elementos e propriedades úteis de serem configuradas, resultando no código exibido na Listagem 2.3.

Listagem 2.3 – Código L<sup>A</sup>T<sub>E</sub>X utilizado para inclusão das figuras na Seção 2.4.

```
1 \begin{figure}
2   \centering
3   \includegraphics[width=.25\textwidth]{figuras/fig-fundteo-exemplo.png}
4   \caption{Exemplo de figura.}
5   \label{fig-fundteo-exemplo}
6 \end{figure}
7
8 \begin{sidewaysfigure}
9   \centering
10  \includegraphics[width=\textwidth]{figuras/fig-fundteo-exemplosideways}
11  \caption{Exemplo de figura em modo paisagem: um modelo de objetivos~\cite{souza
      -mylopoulos:spe13}.}
12  \label{fig-fundteo-exemplosideways}
13 \end{sidewaysfigure}
```

O comando `\centering` centraliza a figura na página. A opção `width` do comando `\includegraphics{}` determina o tamanho da figura e usa-se `\textwidth` (opcionalmente multiplicado por um número) para se referir à largura da página.

O parâmetro do comando `\includegraphics{}` indica onde a imagem pode ser encontrada. Foi criado o diretório `figuras/` para conter as figuras do documento, dando uma melhor organização aos arquivos. Ao abrir esta pasta, repare que as figuras possuem duas versões—uma em `.eps` e outra em `.pdf`—e que o comando `\includegraphics{}` não especifica a extensão. Isso se dá porque o L<sup>A</sup>T<sub>E</sub>X possui um compilador para formato PostScript (`latex`) que espera as imagens em `.eps` e um compilador para PDF (`pdflatex`) que espera as imagens em `.pdf`.

Por fim, o comando `\caption{}` especifica a descrição da figura e `\label{}`, como de costume, estabelece um rótulo para permitir referência cruzada de figuras. Note ainda que é utilizada a mesma estratégia de nomenclatura de rótulos usada nas listagens, porém utilizando o prefixo `fig-`.

As figuras 1 e 2 mostram o resultado do código da Listagem 2.3. A Figura 2, em particular, utiliza o pacote `rotating` para mostrar figuras largas em modo paisagem. Basta usar o ambiente `sidewaysfigure` ao invés de `figure`.



Figura 1 – Exemplo de figura.

Tabela 2 – Exemplo de tabela com diferentes alinhamentos de conteúdo.

Centralizado	Esquerda	Direita	Parágrafo
C	L	R	Alinhamento de tipo parágrafo especifica largura da coluna e quebra o texto automaticamente.
Linha 2	Linha 2	Linha 2	Linha 2

Tabela 3 – Exemplo que especifica largura de coluna e usa lista enumerada (adaptada de (SOUZA; MYLOPOULOS, 2013)).

<i>AwReq</i>	Adaptation strategies	Applicability conditions
AR1	1. <i>Warning</i> (“AS Management”) 2. <i>Reconfigure</i> ( $\emptyset$ )	1. Once per adaptation session; 2. Always.
AR2	1. <i>Warning</i> (“AS Management”) 2. <i>Reconfigure</i> ( $\emptyset$ )	1. Once per adaptation session; 2. Always.

2.5 Tabelas

Tabelas são um ponto fraco do L<sup>A</sup>T<sub>E</sub>X. Elas são complicadas de fazer e, dependendo da complexidade da tabela (muitas células mescladas, por exemplo), vale a pena construi-las em outro programa (por exemplo, em seu editor de texto favorito), converter para PDF e inclui-las no documento como figuras. Mostramos, no entanto, alguns exemplos de tabela a seguir. O código utilizado para criar as tabelas encontra-se nas listagens 2.4 e 2.5.

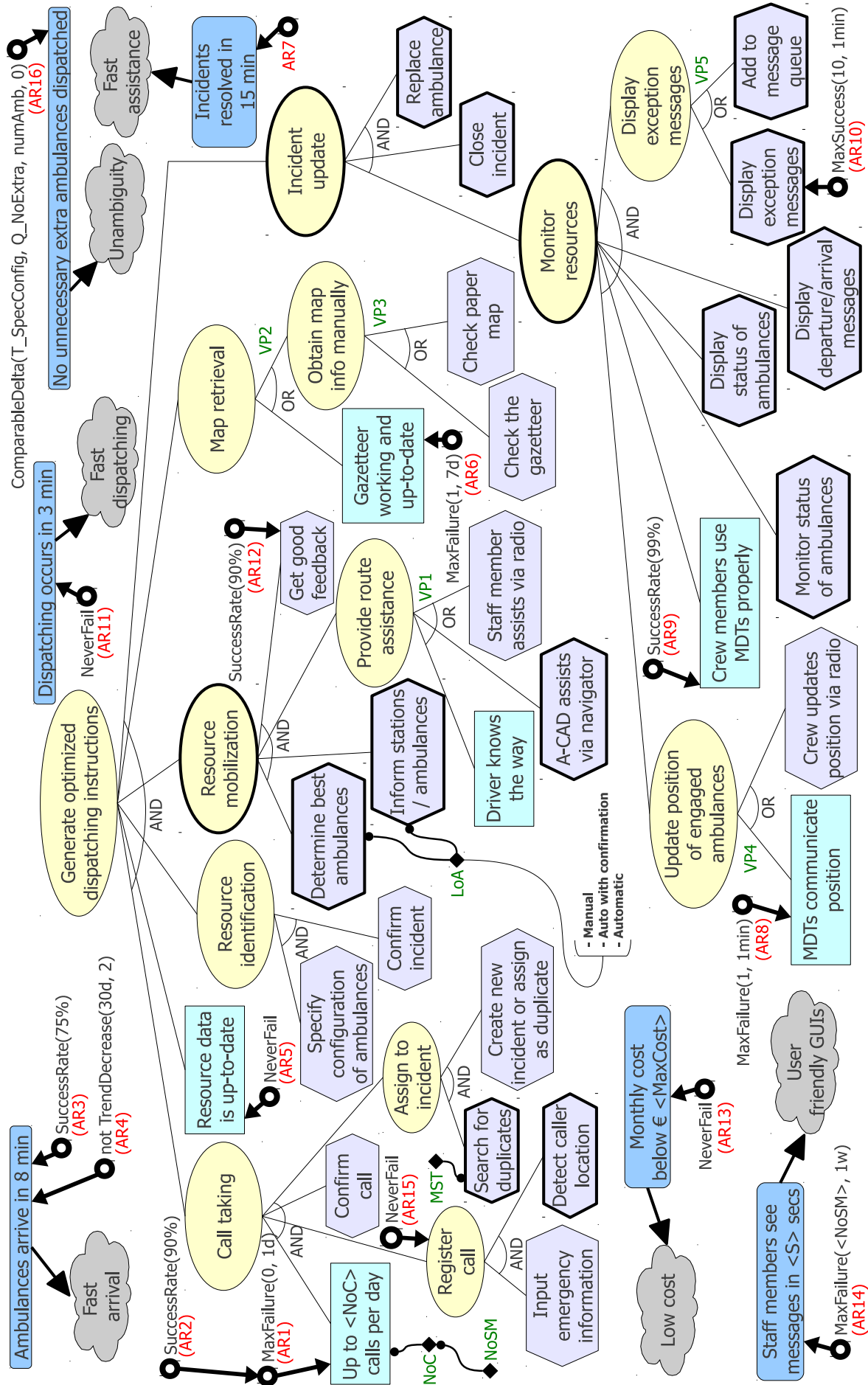


Figura 2 – Exemplo de figura em modo paisagem: um modelo de objetivos (SOUZA; MYLOPOULOS, 2013).

Listagem 2.4 – Código L<sup>A</sup>T<sub>E</sub>X utilizado para inclusão das tabelas 2 e 3.

```

1 % Exemplo de tabela 01:
2 \begin{table}
3 \caption{Exemplo de tabela com diferentes alinhamentos de conteudo.}
4 \label{tbl-intro-exemplo01}
5 \centering
6 \begin{tabular}{| c | l | r | p{40mm} |}\hline
7 \textbf{Centralizado} & \textbf{Esquerda} & \textbf{Direita} & \textbf{Parágrafo}
8 C & L & R & Alinhamento de tipo parágrafo especifica largura da coluna e quebra o
9 \hline
10 Linha 2 & Linha 2 & Linha 2 & Linha 2\\
11 \hline
12 \end{tabular}
13 \end{table}
14
15 % Exemplo de tabela 02:
16 \begin{table}
17 \caption{Exemplo que especifica largura de coluna e usa lista enumerada (adaptada
18 de~\cite{souza-mylopoulos:spe13}).}
19 \centering
20 \renewcommand{\arraystretch}{1.2}
21 \begin{small}
22 \begin{tabular}{| p{15mm} | p{77mm} | p{55mm} |}\hline
23 \textbf{\textit{AwReq}} & \textbf{Adaptation strategies} & \textbf{Applicability}
24 conditions}\hline
25 AR1 &
26 \vspace{-2mm}\begin{enumerate}[topsep=0cm, partopsep=0cm, itemsep=0cm, parsep=0cm
27 , leftmargin=0.5cm]
28 \item \textit{Warning}('AS Management')
29 \item \textit{Reconfigure}($\varnothing$)
30 \end{enumerate}\vspace{-4mm} &
31 \vspace{-2mm}\begin{enumerate}[topsep=0cm, partopsep=0cm, itemsep=0cm, parsep=0cm
32 , leftmargin=0.5cm]
33 \item Once per adaptation session;
34 \item Always.
35 \end{enumerate}\vspace{-4mm}
36 \hline
37 AR2 &
38 \vspace{-2mm}\begin{enumerate}[topsep=0cm, partopsep=0cm, itemsep=0cm, parsep=0cm
39 , leftmargin=0.5cm]
40 \item \textit{Warning}('AS Management')
41 \item \textit{Reconfigure}($\varnothing$)
42 \end{enumerate}\vspace{-4mm} &
43 \vspace{-2mm}\begin{enumerate}[topsep=0cm, partopsep=0cm, itemsep=0cm, parsep=0cm
44 , leftmargin=0.5cm]
45 \item Once per adaptation session;
46 \item Always.
47 \end{enumerate}\vspace{-4mm}
48 \hline
49 \end{tabular}
50 \end{small}
51 \end{table}

```

Listagem 2.5 – Código L<sup>A</sup>T<sub>E</sub>X utilizado para inclusão da Tabela 4.

```

1 % Exemplo de tabela 03:
2 \begin{table}
3 \caption{Exemplo que mostra equações em duas colunas (adaptada de~\cite{souza-
   mylopoulos:spe13}).}
4 \label{tbl-intro-exemplo03}
5 \centering
6 \vspace{1mm}
7 \fbox{\begin{minipage}{.98\linewidth}
8 \begin{minipage}{0.51\linewidth}
9 \vspace{-4mm}
10 \begin{eqnarray}
11 \Delta \left( I_{AR1} / NoSM \right) \left[ 0, maxSM \right] > 0\\
12 \Delta \left( I_{AR2} / NoSM \right) \left[ 0, maxSM \right] > 0\\
13 \Delta \left( I_{AR3} / LoA \right) < 0\\
14 \end{eqnarray}
15 \vspace{-6mm}
16 \end{minipage}
17 \hspace{2mm}
18 \vline
19 \begin{minipage}{0.41\linewidth}
20 \vspace{-4mm}
21 \begin{eqnarray}
22 \Delta \left( I_{AR11} / VP2 \right) < 0\\
23 \Delta \left( I_{AR12} / VP2 \right) > 0\\
24 \Delta \left( I_{AR6} / VP3 \right) > 0\\
25 \end{eqnarray}
26 \vspace{-6mm}
27 \end{minipage}
28 \end{minipage}}
29 \end{table}

```

Tabela 4 – Exemplo que mostra equações em duas colunas (adaptada de (SOUZA; MYLOPOULOS, 2013)).

$\Delta(I_{AR1}/NoSM)[0, maxSM] > 0$	(2.1)	$\Delta(I_{AR11}/VP2) < 0$	(2.5)
$\Delta(I_{AR2}/NoSM)[0, maxSM] > 0$	(2.2)	$\Delta(I_{AR12}/VP2) > 0$	(2.6)
$\Delta(I_{AR3}/LoA) < 0$	(2.3)	$\Delta(I_{AR6}/VP3) > 0$	(2.7)
	(2.4)		(2.8)

## Referências

ALBERT, R.; JEONG, H.; BARABÁSI, A.-L. Internet: Diameter of the world-wide web. *nature*, Nature Publishing Group, v. 401, n. 6749, p. 130, 1999. Citado na página 6.

DIJKSTRA, E. W. A note on two problems in connexion with graphs. *Numerische mathematik*, Springer, v. 1, n. 1, p. 269–271, 1959. Citado na página 6.

LATTES, C. M. G.; OCCHIALINI, G. P.; POWELL, C. F. Observations on the tracks of slow mesons in photographic emulsions. *Nature*, Nature Publishing Group, v. 160, n. 4067, p. 486, 1947. Citado na página 6.

SOUZA, V. E. S.; MYLOPOULOS, J. Designing an adaptive computer-aided ambulance dispatch system with Zanshin: an experience report. *Software: Practice and Experience* (online first: <http://dx.doi.org/10.1002/spe.2245>), Wiley, 2013. Citado 3 vezes nas páginas 9, 10 e 12.