

## Modelo de Documentação de Software – 5W2H

## **SUMÁRIO**

<b>1. INTRODUÇÃO</b>	<b>3</b>
<b>2. OBJETIVO</b>	<b>3</b>
<b>3. ESTRUTURA</b>	<b>3</b>
<b>3.1 SEÇÃO 1: VISÃO GERAL</b>	<b>3</b>
<b>3.2 SEÇÃO 2: ARQUITETURA DO SOFTWARE</b>	<b>4</b>
<b>3.3 SEÇÃO 3: FUNCIONALIDADES</b>	<b>4</b>
<b>3.4 SEÇÃO 4: INSTALAÇÃO E CONFIGURAÇÃO</b>	<b>4</b>
<b>3.5 SEÇÃO 5: SUPORTE TÉCNICO</b>	<b>4</b>
<b>3.6 SEÇÃO 6: GLOSSÁRIO</b>	<b>4</b>
<b>3.7 SEÇÃO 7: APÊNDICE</b>	<b>5</b>
<b>4. REQUISITOS ADICIONAIS</b>	<b>5</b>

## **1. Introdução**

A documentação de software é um elemento crucial para o sucesso de qualquer projeto de desenvolvimento. Ela serve como guia para desenvolvedores, usuários finais e equipe de suporte técnico, garantindo a compreensão clara do software, sua funcionalidade e como utilizá-lo de forma eficaz. É uma etapa do desenvolvimento do produto que consiste em registrar em texto, e de forma precisa o que há de essencial a saber sobre um sistema. A ausência de uma documentação para o software está associada a problemas de usabilidade e legibilidade, entre outros.

Este documento apresenta um modelo de documentação de software com ênfase na técnica 5W2H (O quê, Por quê, Quem, Quando, Onde, Como e Quanto), além de outros requisitos importantes para garantir a clareza, completude e organização da documentação.

## **2. Objetivo**

O objetivo deste modelo é estabelecer um padrão para a documentação de software, facilitando a comunicação entre os envolvidos no projeto e garantindo a qualidade do software entregue.

## **3. Estrutura**

A documentação de software deve ser estruturada de forma lógica e organizada, abrangendo todos os aspectos relevantes do software. Uma estrutura recomendada é a seguinte:

### **3.1 Seção 1: Visão Geral**

3.1.1 O quê: Descrever o nome do software, sua versão e objetivo principal.

3.1.2 Por quê: Explicar a razão de criação do software e o problema que ele resolve.

3.1.3 Quem: Identificar o público-alvo da documentação, incluindo desenvolvedores, usuários finais e equipe de suporte.

## **3.2 Seção 2: Arquitetura do Software**

3.2.1 O quê: Descrever os componentes principais do software e suas interações.

3.2.2 Como: Detalhar a arquitetura do software, incluindo diagramas, fluxogramas e outras representações visuais.

## **3.3 Seção 3: Funcionalidades**

3.3.1 O quê: Descrever todas as funcionalidades do software, com casos de uso e exemplos.

3.3.2 Como: Explicar como cada funcionalidade funciona, passo a passo, com instruções claras e concisas.

3.3.3 Quando: Indicar em quais situações cada funcionalidade deve ser utilizada.

## **3.4 Seção 4: Instalação e Configuração**

3.4.1 O quê: Descrever os requisitos de sistema para a instalação do software.

3.4.2 Como: Detalhar o processo de instalação do software, passo a passo, com *screenshots* e instruções claras.

3.4.3 Onde: Indicar o local de download do software e os arquivos necessários para a instalação.

3.4.4 Quanto: Informar o tempo estimado para a instalação do software.

## **3.5 Seção 5: Suporte Técnico**

3.5.1 Quem: Identificar a equipe de suporte técnico e seus canais de contato.

3.5.2 Como: Descrever os procedimentos para solicitar suporte técnico, incluindo informações sobre relatórios de bugs e soluções de problemas.

3.5.3 Quando: Indicar o horário de atendimento da equipe de suporte técnico.

## **3.6 Seção 6: Glossário**

3.6.1 O quê: Definir os termos técnicos utilizados na documentação do software.

### **3.7 Seção 7: Apêndice**

3.7.1 Qualquer informação adicional relevante, como documentação de APIs, logs de erros e histórico de versões.

## **4. Requisitos Adicionais**

Para garantir a qualidade da documentação de software, é importante considerar os seguintes requisitos adicionais:

- Clareza: A linguagem utilizada na documentação deve ser clara, concisa e objetiva, evitando jargões técnicos desnecessários.
- Completude: A documentação deve abranger todos os aspectos do software, desde a instalação até o uso das funcionalidades avançadas.
- Atualização: A documentação deve ser atualizada regularmente para refletir as mudanças no software.
- Organização: A documentação deve ser organizada de forma lógica e fácil de navegar, utilizando índices, tabelas e outros recursos visuais.
- Acessibilidade: A documentação deve estar disponível em formatos acessíveis, como HTML, PDF e ePub.
- Autenticidade: A documentação deve ser precisa e confiável, revisada por especialistas no software.