

# **Biblioteca Comunitária**

## **Documento de Arquitetura de Software**

**Versão 3.0**

## HISTÓRICOS DE REVISÕES

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
04/10/2012	1.0	Criação do artefato.	Francisco Ernesto, Jady Pâmella
05/10/2012	2.0	Revisão do artefato.	Francisco Ernesto, Jady Pâmella
20/10/2012	3.0	Revisão do artefato.	Francisco Ernesto, Jady Pâmella

## SUMÁRIO

<b>1. INTRODUÇÃO.....</b>	<b>4</b>
<b>2. REPRESENTAÇÃO ARQUITETURAL.....</b>	<b>4</b>
<b>3. OBJETIVOS ARQUITETURAIS E RESTRIÇÕES.....</b>	<b>5</b>
<b>4. VISÃO DE CASOS DE USO.....</b>	<b>6</b>
<b>5. VISÃO LÓGICA.....</b>	<b>7</b>
<b>6. VISÃO DE PROCESSO.....</b>	<b>9</b>
<b>7. VISÃO DE IMPLANTAÇÃO.....</b>	<b>9</b>
<b>8. VISÃO DE IMPLEMENTAÇÃO.....</b>	<b>9</b>
<b>9. TAMANHO E PERFORMANCE.....</b>	<b>10</b>
<b>10. QUALIDADE.....</b>	<b>10</b>
<b>11. ASSINATURAS.....</b>	<b>10</b>

## 1. INTRODUÇÃO

O documento de arquitetura apresenta uma visão geral abrangente da arquitetura do sistema e utiliza uma série de visões arquiteturais diferentes para ilustrar os diversos aspectos do sistema. Sua intenção é capturar e transmitir as decisões significativas do ponto de vista da arquitetura que foram tomadas em relação ao sistema. A arquitetura a ser implementada foi definida pelo cliente para o desenvolvimento de seus sistemas.

Este documento fornece uma descrição da arquitetura do projeto Biblioteca Comunitária, mostrando e exemplificando a solução arquitetural utilizada para atender os requisitos funcionais e não funcionais do sistema.

## 2. REPRESENTAÇÃO ARQUITETURAL

A modelagem, implementação e documentação de um sistema requerem que o sistema seja visualizado de diferentes perspectivas. Optamos por desenvolver o sistema da Biblioteca Comunitária utilizando o framework GWT (Google Web Toolkit) que é um conjunto de ferramentas desenvolvido pela Google para criar aplicativos em AJAX com Java e hospedar na infraestrutura gratuita da própria empresa (Google AppEngine). A arquitetura do GWT divide a aplicação em 2 partes: Client-Side Code e Server-Side Code e é organizada segundo a estrutura da figura 1.

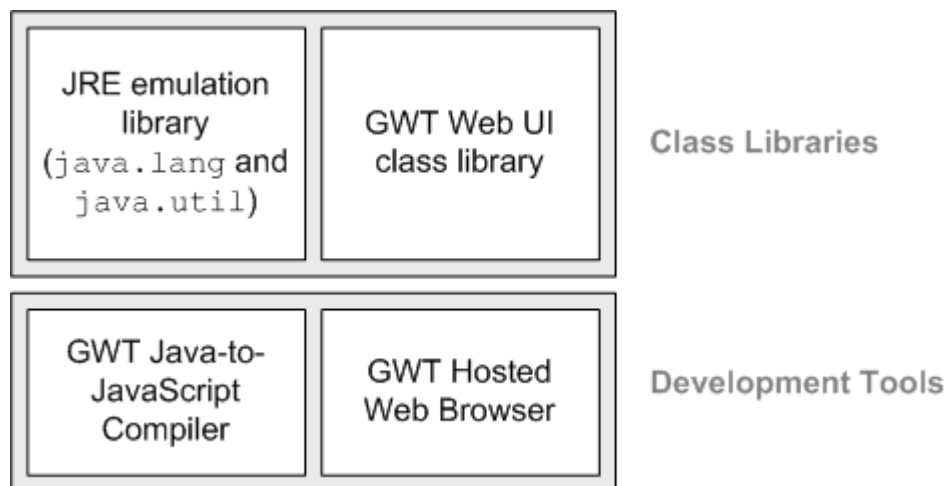


Figura 1 – Arquitetura GWT

O GWT contém implementações JavaScript das bibliotecas de classes Java mais utilizadas, como *java.lang* e um subconjunto de *java.util*. Outras classes Java não são suportadas pelo GWT, como a *java.io*, já que esta não se aplica a aplicações web.

É utilizada na elaboração de sua interface a uma biblioteca - GWT Web UI - que possui todos os componentes de interface com o usuário: botões, caixas de texto, imagens, texto e vários outros componentes, tal qual um Swing ou AWT.

## Biblioteca Comunitária

Ainda existem o seu compilador que é responsável pela tradução do código Java para o código JavaScript que vai executar na Internet - GWT Compiler – e um servidor interno para que permite executar aplicações GWT sem a necessidade de compilar para JavaScript, executando como Java na JVM. Isto possibilita verificar as alterações em modo de execução.

Desse modo, a arquitetura proposta será apresentada conforme o modelo de visões 4+1, proposta por Philippe Kruchten (Kruchten, 1995), utilizadas no RUP (Kruchten, 2000), que apresenta 5 perspectivas diferentes, sendo elas: Visão de Cenários, Visão Lógica, Visão de Processos, Visão de Componentes e Visão de Implantação. No entanto, conforme propósito deste documento apenas as visões de Casos de Uso, Lógica, Componentes e de Implantação serão descritas.

**Visão de Casos de Uso:** O propósito desta visão é apresentar os casos de uso arquiteturalmente significativos para o sistema, que direcionarão a arquitetura do sistema como um todo. Esta visão usará diagramas de casos de uso para expor as funcionalidades e diagramas de sequência para mostrar a interação entre objetos e o relacionamento entre os diversos elementos da arquitetura.

**Visão Lógica:** Esta visão irá apresentar as definições do sistema através de diagramas de classes, ou quaisquer outros diagramas que descrevam os serviços que o sistema fornecerá aos seus usuários. Esta visão será utilizada para apresentar a arquitetura num nível elevado de abstração.

**Visão de Processos:** Esta visão mostra a decomposição do sistema, bem como as formas de comunicação entre processos, passagem de mensagens, atividades entre componentes e sequência de mensagem. Essa representação é feita através dos diagramas de sequência, e opcionalmente pelos diagramas de colaboração e atividades.

**Visão de Componentes (Implementação):** Esta visão irá descrever a organização dos subsistemas e componentes do sistema. Esta visão irá mostrar as diversas camadas do software, bem como as fronteiras entre essas camadas.

**Visão de Implantação:** Esta visão descreverá como o sistema será distribuído e implantado nos nós físicos, nos quais será executado.

### 3. OBJETIVOS ARQUITETURAIS E RESTRIÇÕES

A arquitetura da Biblioteca Comunitária deve obedecer às metas e restrições impostas pelos requisitos estipulados no Modelo de Casos de Uso e no Documento de Requisitos do sistema. Assim, itens são listados abaixo meramente para ressaltar alguns dos pontos que têm de ser considerados na escolha das alternativas arquitetônicas:

- ✓ O sistema deverá ser desenvolvido em Java, utilizando o Google Web Toolkit;
- ✓ O código fonte do sistema deve ser livre, certificado pela Apache;

**Biblioteca Comunitária**

- ✓ O sistema deverá ser hospedado na estrutura gratuita Google AppEngine e oferecer disponibilidade 24/7;
- ✓ A base de dados deverá ser em usa JDO e DataNucleos;
- ✓ O sistema deverá garantir a usabilidade do usuário;
- ✓ O administrador do sistema deverá possuir plenas condições técnicas de entender o Manual de Utilização do Sistema e interagir com o aplicativo online.

#### 4. VISÃO DE CASOS DE USO

Os itens a seguir identificam casos de uso e cenários escolhidos para promover a estabilização da arquitetura da Biblioteca Comunitária por meio da construção de casos de uso de teste durante a fase de Elaboração. Sua escolha foi determinada pela cobertura dada por cada um a aspectos centrais ou críticos da solução técnica.

Na tabela abaixo, o caso de uso e o requisito geral é referido pelo nome. Para uma descrição completa de cada um deles, veja o Modelo de Casos de Uso e a Especificação de Caso de Uso.

Caso de Uso/Cenário	Requisitos (foco de verificação)	Justificativa
Realizar Empréstimo	<ul style="list-style-type: none"><li>• Facilidade de utilização;</li><li>• Velocidade na utilização.</li></ul>	O protótipo servirá para: <ul style="list-style-type: none"><li>• Verificar a facilidade de instalação.</li></ul>
Devolver Mídia	<ul style="list-style-type: none"><li>• Facilidade de utilização;</li><li>• Velocidade na utilização.</li></ul>	
Reservar Mídia	<ul style="list-style-type: none"><li>• Facilidade de utilização;</li><li>• Velocidade na utilização.</li></ul>	

## **Atores do sistema**



**Figura 2 – Representação dos Atores.**

## **5. VISÃO LÓGICA**

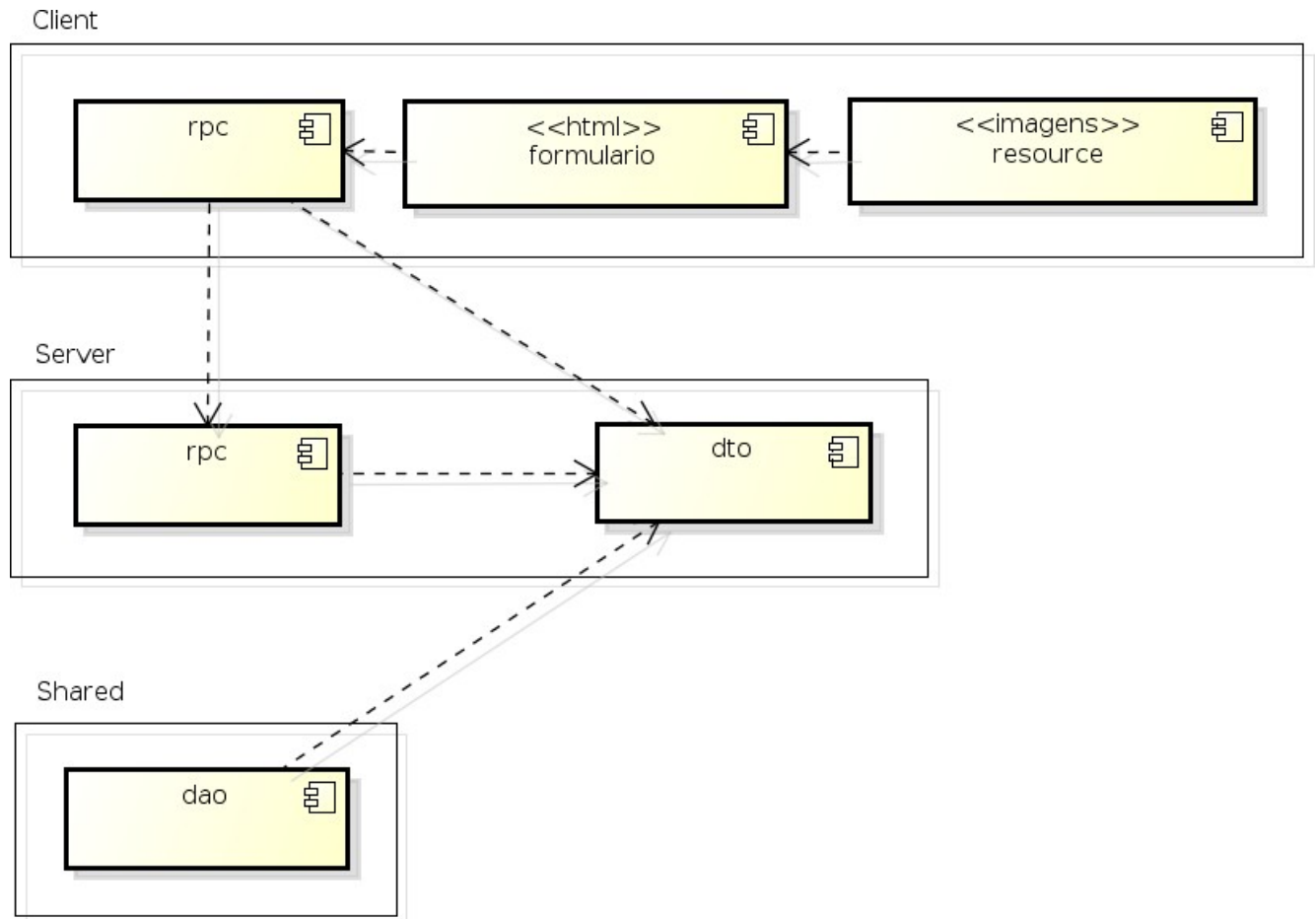
Essa visão apresenta o modelo arquitetural em camadas com seus respectivos padrões de projeto.

### **5.1 Visão Geral**

A arquitetura da Biblioteca Comunitária é dividida em duas camadas com uma camada intermediária que cuida da conversão das informações que trafegam entre a camada Client e Server.

### **5.2 Pacotes de Projeto Arquiteturalmente Significativos**

Os diagramas abaixo mostram as classes, os pacotes e as camadas mais significativas do ponto de vista arquitetural do framework de arquitetura.

**Biblioteca Comunitária****Figura 3 – Representação das camadas e dos respectivos padrões de projeto.****Client**

Utilização de Javascript, CSS e HTML gerados pelo compilador do GWT. Componentes e Estrutura definida conforme suas respectivas documentações.

**Server**

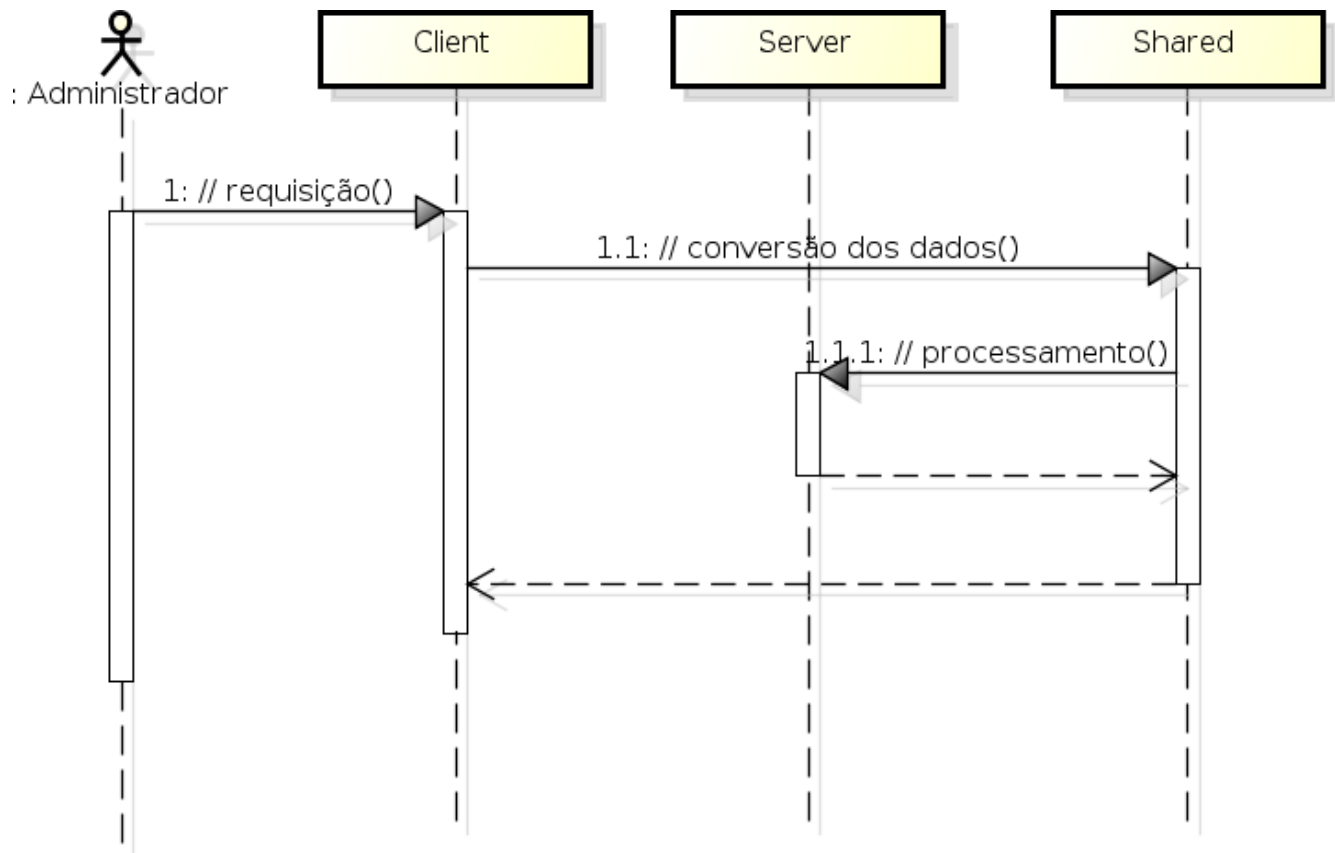
Composto pela RPC gerado pelo GWT, para transporte dos dados e ações em servidor da camada de apresentação.

**Shared**

Utilização do DAO gerado para a conversa entre os dados do cliente/servidor, para a sua persistência e recuperação.

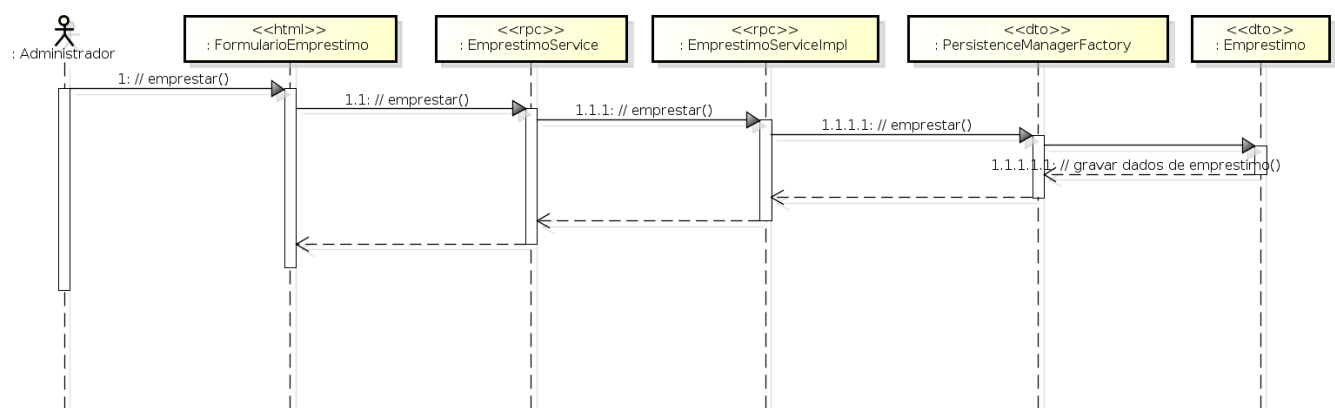


**Biblioteca Comunitária**



**Figura 4 – Representação genérica de utilização da infra-estrutura.**

### 5.3 Realizações de Casos de Uso



**Figura 5 – Diagrama de Sequência do Fluxo Principal: Realizar Empréstimo.**

## 6. VISÃO DE PROCESSO

Não se aplica.

## 7. VISÃO DE IMPLANTAÇÃO

Os itens a seguir descrevem a configuração da topologia de rede na qual o sistema deve ser implantado e executado.

O diagrama abaixo representa uma configuração de rede e os protocolos utilizados na comunicação entre os nós a partir da interpretação dos requisitos do sistema. Deverá ser detalhado o quanto antes, as demais configurações do hardware do servidor de aplicação para o qual o sistema será instalado, devido à sua considerável influência sobre as decisões arquiteturais. É necessário esclarecer ainda a localização dos sistemas integrados com a Biblioteca Comunitária via HTTP.

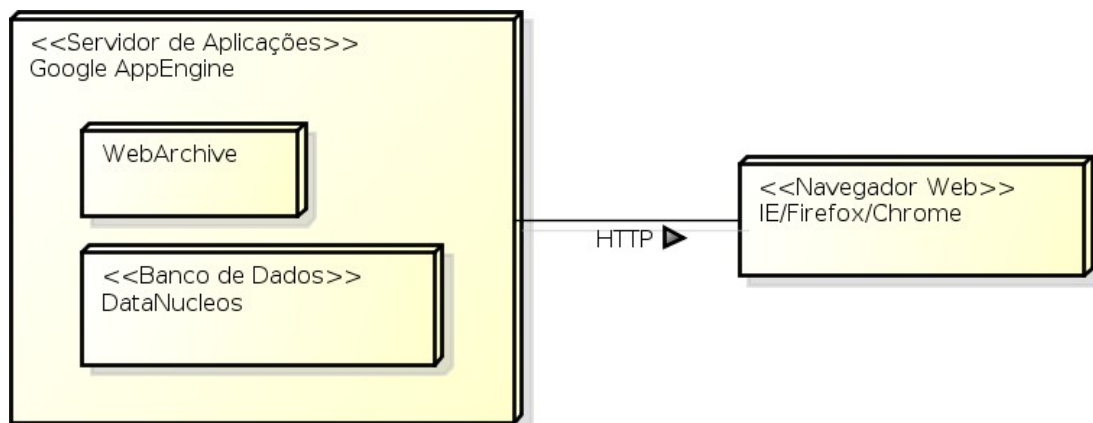


Figura 6 – Diagrama de Implantação: Topologia de Rede.

## 8. VISÃO DE IMPLEMENTAÇÃO

A visão de implementação tem como objetivo mostrar como está organizado o sistema e seus componentes. Os itens abaixo mostram a divisão em camadas e o reuso dos ativos de software.

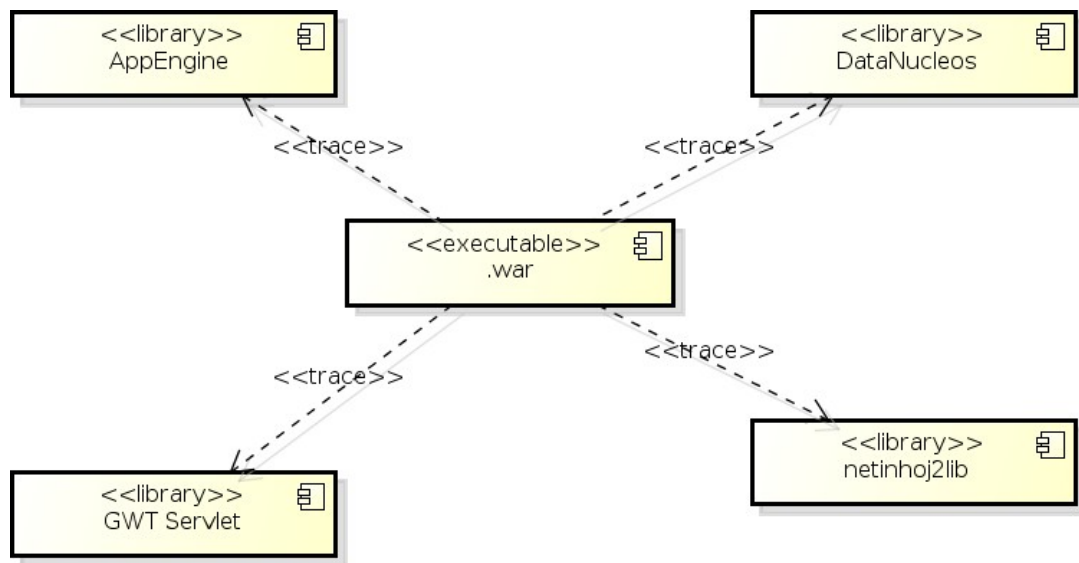
### 8.1 Visão Geral

Vide seção [5.2 Pacotes de Projeto Arquiteturalmente Significativos](#)

### 8.2 Camadas

Vide seção [5.2 Pacotes de Projeto Arquiteturalmente Significativos](#)

O diagrama de componentes abaixo representa a dependência de componentes do sistema.

**Biblioteca Comunitária**

**Figura 7 – Diagrama de Componentes: Reuso dos ativos de software.**

## 9. TAMANHO E PERFORMANCE

Para informações sobre as características de tamanho e desempenho a serem suportadas pela arquitetura da Biblioteca Comunitária, veja o documento de Requisitos.

## 10. QUALIDADE

Para informações sobre os requisitos de qualidade a serem suportadas pela arquitetura da Biblioteca Comunitária, veja o documento de Requisitos.

## 11. ASSINATURAS

Concordam com conteúdo deste documento:

Data: \_\_\_\_/\_\_\_\_/\_\_\_\_

Data: \_\_\_\_/\_\_\_\_/\_\_\_\_