# Documento de Visão do Projeto - P2P (pirate to pirate) com WebRTC e WebSocket

## Histórico de Versões

| **Data** | **Versão** | **Descrição** | **Autor** | **Revisor** | **Aprovado por** |
| --- | --- | --- | --- | --- | --- |
| 04/02/2025 | 1.1 | DVP do projeto | Iderlan Matheus | Rennan | Rennan |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## 1. Objetivo

O objetivo deste projeto é fornecer um sistema de comunicação peer-to-peer (P2P) utilizando WebRTC e WebSocket. O sistema permitirá que dois usuários se conectem diretamente para troca de arquivos sem necessidade de um servidor intermediário, reduzindo latência e aumentando a privacidade. Um servidor de sinalização baseado em WebSocket será utilizado para facilitar a descoberta dos pares e a troca de credenciais necessárias para a conexão WebRTC.

## 2. Descrição do Projeto

O sistema consiste em um servidor WebSocket responsável pela troca inicial de arquivos de sinalização entre os clientes. Após estabelecida a conexão WebRTC, a comunicação ocorre diretamente entre os clientes. A interface do usuário será baseada em HTML e JavaScript, permitindo uma interação simples e intuitiva para envio e recebimento de arquivos.

## 3. Envolvimento

### 3.1 Abrangência

O sistema será acessível via navegador web e poderá ser utilizado tanto em redes internas quanto na internet. Ele é projetado para ser um sistema leve, podendo ser utilizado para comunicação direta entre usuários sem a necessidade de instalar aplicativos adicionais.

### 3.2 Papel da Equipe de Desenvolvimento

* Responsável pela identificação dos requisitos e desenvolvimento do sistema.
* Implementa as funcionalidades necessárias para o funcionamento correto da aplicação.
* Garante a correta implementação do fluxo P2P utilizando WebRTC.

## 4. Necessidades e Funcionalidades

| **Necessidade** | **Benefício** |
| --- | --- |
| Troca de arquivos em tempo real sem servidor intermediário | Menor latência e mais privacidade |
| Interface intuitiva para conexão entre usuários (avaliada por questionário SUS) | Facilidade de uso |
| Geração de ID único para cada cliente | Facilita o pareamento |
| Notificação para que aceite de conexão | Controle sobre quem pode se conectar |

As funcionalidades implementadas incluem:

* **F1.1**: Geração automática de um ID para cada cliente ao acessar o sistema.
* **F1.2**: Interface para entrada do ID do outro cliente para solicitação de conexão.
* **F1.3**: Notificação de conexão para o cliente receptor.
* **F1.4**: Aceitação manual de conexão antes de estabelecer comunicação.
* **F1.5**: Troca de arquivos via WebRTC sem passar pelo servidor.

## 5. Restrições Tecnológicas

* O sistema deve ser acessível via navegadores modernos (Google Chrome, Firefox, Edge).
* O servidor de sinalização deve rodar em PHP com Ratchet e estar acessível publicamente.
* O WebRTC exige que pelo menos um cliente tenha um STUN server para obter IPs válidos.

## 6. Critérios de Aceitação do Sistema

* O sistema deve permitir que dois clientes se conectem via ID e troquem arquivos diretamente.
* A interface deve ser responsiva e de fácil uso (testada via questionário SUS).
* O sistema deve garantir que somente clientes autorizados possam estabelecer conexão.
* A conexão P2P deve ser mantida estável durante a troca de arquivos.

## 7. Escopo Não Incluído no Sistema

* O sistema não oferecerá chamadas de áudio ou vídeo, apenas troca de arquivos.
* Não haverá armazenamento de arquivos no servidor.
* O sistema não implementará autenticação de usuários.

## 8. Premissas e Restrições

| **Premissas** | **Restrições** |
| --- | --- |
| Todos os usuários possuem acesso à internet. | O servidor de sinalização não deve armazenar logs de conversa. |
| O navegador do usuário suporta WebRTC e WebSocket. | Apenas conexões com aceitação explícita serão estabelecidas. |