Bate

**ESII 2017**

Fábrica Experimental de Desenvolvimento de Software

ESII 2017

***Projeto: Bate Papo***

***Bate Papo***

Documento de Requisitos

**Baseado na 5ª edição do Guia PMBOK®**

Sumário

[Histórico de alterações do documento 1](#_Toc515887183)

[1. Lista de Stakeholders 3](#_Toc515887184)

[2. Requisitos Funcionais 3](#_Toc515887185)

[3. Requisitos Funcionais 3](#_Toc515887186)

[[RF001] Aplicação Cliente 3](#_Toc515887187)

[[RF002] Aplicação Servidor 4](#_Toc515887188)

[[RF003] Web Service SOAP 4](#_Toc515887189)

[[RF004] Web Service REST 4](#_Toc515887190)

[4. Requisitos Não-Funcionais 5](#_Toc515887191)

[[NF001] Usabilidade 5](#_Toc515887192)

# Lista de Stakeholders

* *João Pedro A. Campos – Gerente do projeto / Programador*
* Felipe Taparello *– Programador*
* Eduardo Golin Zannela *– Programador*
* Gabriel Consalter *– Programador*
* Rafael Vissotto *– Programador*
* Jônatas Casiraghi Vaz *– Programador*
* Igor Scheuermann *– Programador*

# Requisitos Funcionais

*[RF001] Aplicação Cliente – Telas* responsáveis pelas interações com o cliente.

*[RF002] Aplicação Servidor –* Responsável pelas interações entre o sistema e a integração das funções de acesso aos Web Services.

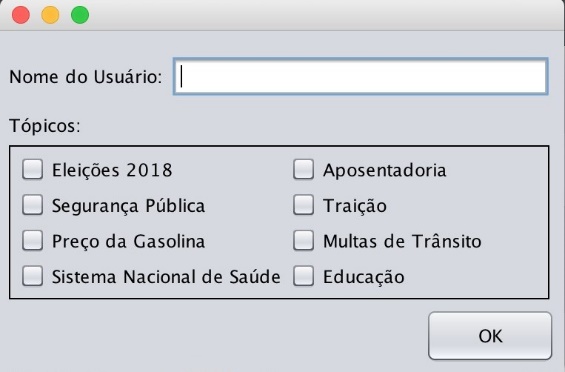
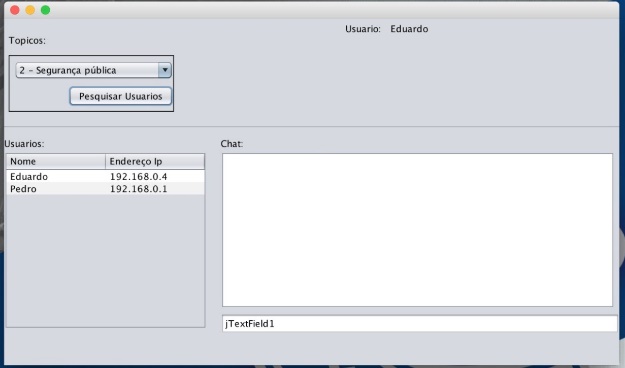
*[RF003] Web Service SOAP –* Responsável pela manutenção das tabelas de dados do sistema.

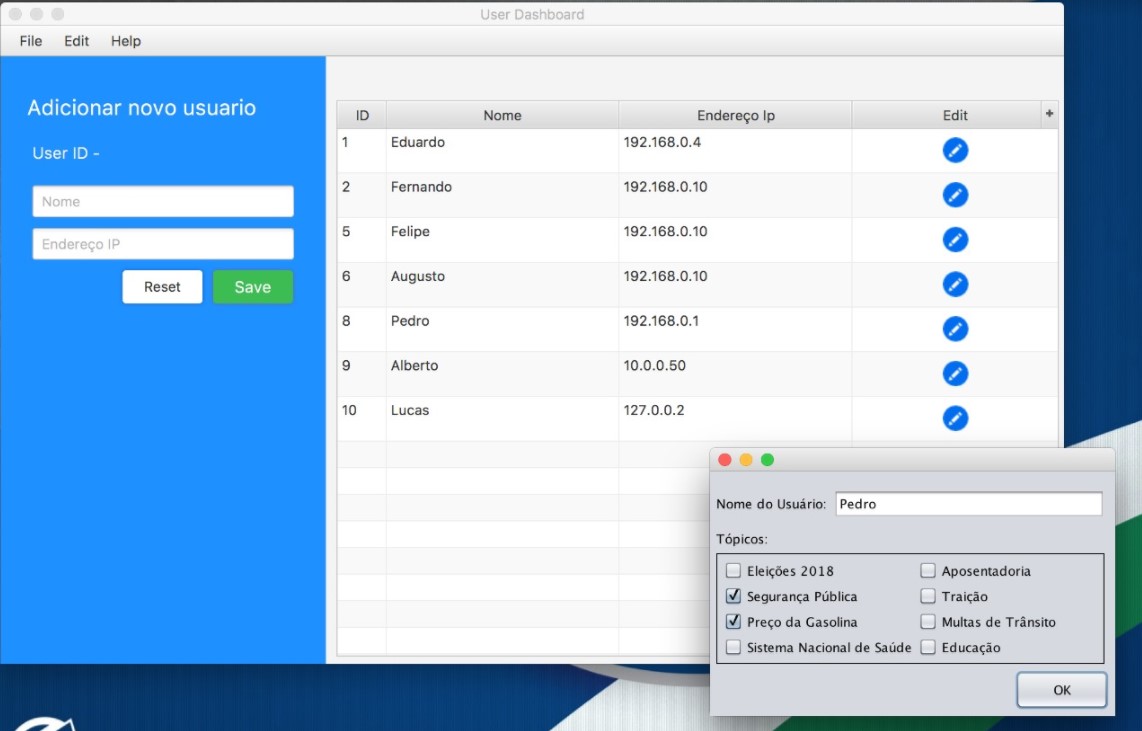
*[RF004] Web Service REST –* Responsável por funcionalidades de análise dos dados gerais do sistema.

# Requisitos Funcionais

[RF001] Aplicação Cliente

* **Descrição do caso de uso:**  
  O usuário deverá ter acesso ao sistema para que possa selecionar as pessoas com quem gostaria de iniciar um chat e o Servidor deverá auxiliá-lo nisso. Para isso será feita uma interface para o acesso ao sistema. A aplicação ainda deve ser capaz de mostrar ao usuário o chat com outros usuários, com uma conexão direta entre eles, através do protocolo TCP.
* **Entrada e pré-condições**  
  O usuário deverá ter um nome de usuário cadastrado no banco de dados para ter acesso ao sistema, além de selecionar os tópicos quais está disposto a conversar.  
  Os dados de conexão no chat devem ser recebidos do servidor e uma conexão estabelecida entre os clientes envolvido no chat.
* **Processamento**  
  O usuário deve informar seu nome de Login, o qual deve estar cadastrado no sistema. Em seguida deve forneces os tópicos que está disposto a conversar. Os tópicos devem ser listados em uma aba com Check Boxes. Após entrar no sistema, ele deverá escolher um tópico para conversar, os quais estarão em uma Combo Box, com o tópico selecionado, o servidor irá fornecer os usuários dispostos a conversar sobre aquele tópico. Desse modo o cliente poderá selecionar um usuário, onde iniciará um chat com a pessoa escolhida. A conexão do chat será feita através do protocolo TCP e os dados de conexão serão recebidos pelo servidor.



|  |  |  |  |
| --- | --- | --- | --- |
| Prioridade: | * Essencial | * Importante | * Desejável |

[RF002] Aplicação Servidor

* **Descrição do caso de uso:**  
  A Aplicação do servidor será feita para facilitar a conversa entre os usuários. O servidor irá receber os usuários dentro do sistema e redirecioná-los para os tópicos, usuários e chats que os mesmos escolherem.
* **Entrada e pré-condições**  
  Aplicação servidor deve estar em execução.

1. Cliente -> Envio do dados para login: “username” e tópicos. Mensagem UDP.
2. Servidor -> Resposta: OK, acesso ao sistema. Mensagem TCP.
3. Cliente -> Envio do tópico selecionado: “id\_topico”. Mensagem TCP.
4. Servidor -> Resposta: Lista de usuários interessados no tópico (Busca na tabela usuário-tópicos) Mensagem TCP.
5. Cliente -> Seleção de usuário requerido e abertura de novo Chat. Mensagem TCP.
6. Servidor -> Resposta: Envio dos dados para conexão entre os usuários, como porta e endereço IP. Mensagem TCP.

* **Processamento**Após o login do usuário, o cliente irá requisitar o tópico no qual está disposto a conversar, onde o Sistema deverá encontrar, na base de dados, os demais usuários que estão dispostos a falar sobre o mesmo tópico selecionado. Em seguida o cliente deve receber como resposta do servidor uma lista de usuários. Ao selecionar uma pessoa para iniciar a conversa, o cliente deve receber os dados para a conexão com o a pessoa escolhida.

|  |  |  |  |
| --- | --- | --- | --- |
| Prioridade: | * Essencial | * Importante | * Desejável |

[RF003] Web Service SOAP

* **Descrição do caso de uso:**  
  O Web Service SOAP será responsável por fazer a manutenção da tabela de Usuários, permitindo que se possa incluir, excluir e listar os usuários. Também deve ser capaz de registrar acessos ao sistema, registrar pesquisas sobre tópicos e registrar chats.
* **Entrada e pré-condições**  
  Usuário cadastrado no sistema, através da interface de cadastro.
* **Processamento**  
  Cliente informará seu *Username* e os tópicos qual está disposto a conversar através de uma mensagem UDP para o servidor, que contatará o Web Service SOAP para requisitar acesso ao sistema. O mesmo irá cadastrar o usuário e devolver acesso ao sistema. A descrição do funcionamento pode ser encontrada em <https://bitbucket.org/trabalho_sd_2018/bate_papo/wiki/Web%20Service%20Soap>

|  |  |  |  |
| --- | --- | --- | --- |
| Prioridade: | * Essencial | * Importante | * Desejável |

[RF004] Web Service REST

* **Descrição do caso de uso:**  
  O Web Service REST deverá ser implementado como uma complementação às funções de análise dos dados do sistema. Ele deve ser capaz de filtrar as pesquisas da rede por data, além de fornecer dados sobre:

1. Quantos usuários acessaram a rede.
2. Quais os tópicos mais acessados.
3. Quantos chats foram realizados.
4. Quantos chats por usuário foram realizados.

* **Entrada e pré-condições**  
  A mensagem de requisição deve ser enviada através de um JSON, requisição POST, com data inicila e data final, no formato “DD/MM/AAAA hh:mm:ss”. Para o acesso dos dados em geral, apenas uma requisição GET é necessária. O servidor do Web Service Rest deve estar rodando.
* **Processamento**  
  O Servidor do Web Service Rest deve estar em funcionamento, para que receba as requisições. Assim que a requisição esperada for recebida nos endpoints: “host/topicos-mais-acessados”, “host/chats-por-usuario”, “host/usuarios-acessos-rede”, “/chats-realizados”. Cada requisição retornará um JSON com a informação requerida. O funcionamento do Web Service Rest pode ser encontrado em <https://bitbucket.org/trabalho_sd_2018/bate_papo/wiki/Web%20Service%20Rest>

|  |  |  |  |
| --- | --- | --- | --- |
| Prioridade: | * Essencial | * Importante | * Desejável |

# Requisitos Não-Funcionais

[NF001]Usabilidade

A interface com o usuário é de vital importância para o sucesso do sistema.

O sistema terá uma interface amigável ao usuário e condizente ao tema proposto pela disciplina.

|  |  |  |  |
| --- | --- | --- | --- |
| Prioridade: | * Essencial | * Importante | * Desejável |