

## **Projeto Neumann (Pronn)**

Documento de Requisitos

## PREFÁCIO

Versão	Data	Autor	Razão
1.0	03/08/2018	Gabriel, Juliene e Thales	Início do projeto
1.1	10/08/2018	Gabriel, Juliene e Thales	Acréscimo de requisitos
1.2	15/08/2018	Gabriel, Juliene e Thales	Acréscimo de requisitos e acréscimo de diagramas
1.3	17/08/2018	Gabriel, Juliene e Thales	Formatação antes do envio e acréscimo de diagramas

## GLOSSÁRIO

**Aplicação web:** plataforma web contendo páginas, campos textos e botões, sendo executado online na internet.

**Controller:** camada responsável pela interação entre a camada View e o SGBD.

**DAO:** camada que lida com a comunicação entre cliente e servidor.

**JavaScript:** é uma linguagem de programação que permite implementar funcionalidades mais complexas em páginas web.

**Model:** camada utilizada como parâmetro e retorno entre as chamadas das camadas View, Controller e DAO.

**MVC:** Modelo de desenvolvimento de *software* em camadas.

**MySQL:** SGBD gratuito desenvolvido pela Oracle Corporation.

**PHP:** é uma linguagem de programação utilizada para o desenvolvimento de aplicações presentes e atuantes no lado do servidor com o objetivo de gerar conteúdo na World Wide Web.

**SGBD:** coleção de softwares responsável pelo armazenamento e manutenção de informações persistentes do computador.

**View:** camada composta pelas interfaces visuais com o usuário.

## **SUMÁRIO**

PREFACIO .....	1
GLOSSÁRIO .....	2
SUMÁRIO .....	3
INTRODUÇÃO .....	4
ARQUITETURA DO SISTEMA .....	5
REQUISITOS DE USUÁRIO .....	8
REQUISITOS DE SISTEMA.....	12
CRONOGRAMA DE EXECUÇÃO.....	14
PLANEJAMENTO DE TESTE .....	15
APÊNDICE .....	16

## **INTRODUÇÃO**

O Projeto Neumann é um sistema tem que como objetivo ajudar na organização dos projetos de pesquisas científicas, auxiliando os líderes dos grupos na organização de suas pesquisas, publicações, equipamentos e membros. Além de cuidar da parte gerencial o sistema terá uma área pública para disponibilizar informações pertinentes sobre grupos e pesquisas.

## **Motivação**

Sabendo das dificuldades e necessidades que um líder de grupo de pesquisa tem para gerenciar os recursos da equipe, foi proposto um programa para auxiliar na administração daquilo que cabe aos líderes de grupo de pesquisas; principalmente na geração de relatórios que antes era feita de forma braçal, consumindo tempo dos líderes de grupo.

### **Resumo do Sistema**

Nome: Projeto Neumann (Pronn).

Tipo: Aplicação Web.

Linguagens: PHP, JavaScript.

Acesso a banco de dados: Sim (MySQL).

## ARQUITETURA DO SISTEMA

O sistema é desenvolvido no modelo MVC (Model View Controller) com a adição da camada DAO (Data Access Object) para fazer a comunicação com o servidor. O sistema segue a arquitetura de cliente-servidor, fazendo requisições para diminuir a quantidade de processamento no acesso. São as camadas:

- *Model* (Modelo)  
Camada responsável por armazenar os modelos das tabelas contidas no banco de dados. Com base nesses modelos são feitas as consultas no banco de dados.
- *View* (Aplicação)  
Camada responsável pelas interfaces gráficas de usuário. Única parte do sistema que é de fato visível para o usuário e por onde ele interage com o software. É a única camada que é exclusiva do lado do cliente.
- *Controller* (Controle)  
Camada responsável pela aplicação das regras de negócio do *software*. Também é a parte que garante a segurança dos dados, especialmente crucial num ambiente de arquitetura cliente-servidor.
- *DAO* (Objeto de Acesso a Dados)  
Camada responsável pela comunicação com um servidor (o *web service* no lado do cliente eo servidor de banco de dados no lado do servidor). É a camada que inicia, mantém, transmite e termina a conexão com o servidor.

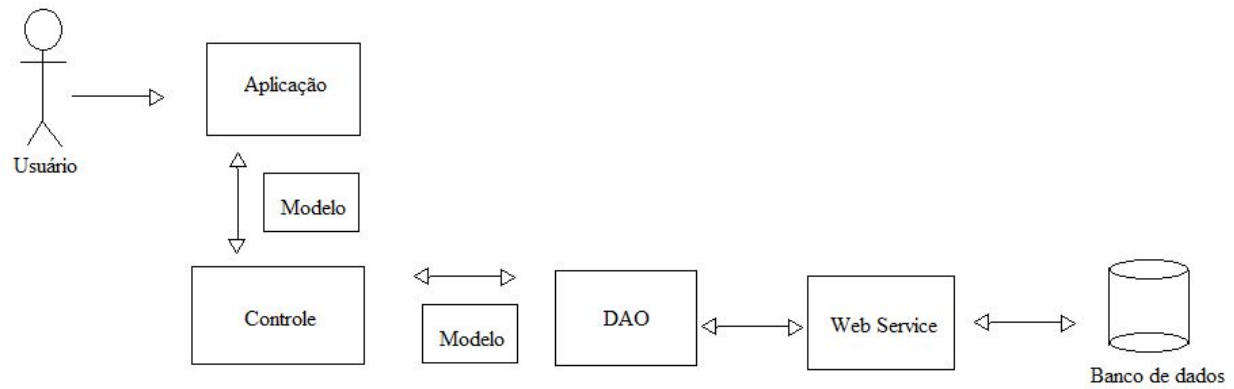


Figura 1 - Comunicação entre as camadas do *software*

O cliente é executado no Computador Local. O servidor de hospedagem, o Web Service e o SGBD são executados no Servidor.

A Figura 2 exibe o diagrama de implantação do sistema, onde é possível perceber a divisão do aplicativo cliente no Computador Local e dos servidores em outro computador.

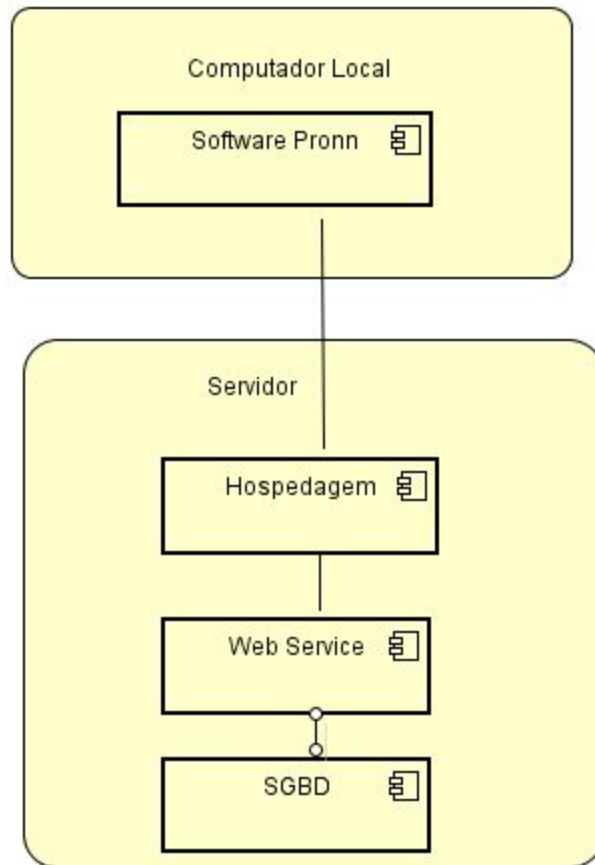


Figura 2 – Diagrama de implantação



## **REQUISITOS NÃO FUNCIONAIS DE USUÁRIO**

Descrevem as propriedades gerais do software de forma simples e concisa.

### **RNF01 – Utilização**

O sistema deve ser executado a partir de qualquer computador com navegador e acesso à internet.

### **RNF02 – Segurança das informações**

As informações devem ser armazenadas de forma segura.

### **RNF03 – Compatibilidade**

O sistema pode ser acessado por estacionários e mobile, como a plataforma utilizada é Web, pode ser executado independente de sistema operacional, apenas necessitando de um navegador.

### **RNF04 – Disponibilidade dos dados**

As informações devem ser disponibilizadas de forma rápida e concisa.

### **RNF05 - Privacidade**

Nenhum dado do usuário é disponibilizado sem consentimento prévio do usuário.

### **RNF06 - Facilidade de uso**

O sistema possui interface simples e que tenta exigir o mínimo de esforço tanto físico quanto de memória do usuário.

### **RNF07 - Desempenho**

O sistema deve permitir o acesso de páginas e a geração de relatórios em tempo útil para o usuário.

### **RNF08 - Fácil manutenção**

O código segue padrões de boas práticas, com variáveis nomeadas para garantir a fácil compreensão de sua função.

## REQUISITOS FUNCIONAIS DE USUÁRIO

Descrevem as funcionalidades do software de maneira simples e objetiva.

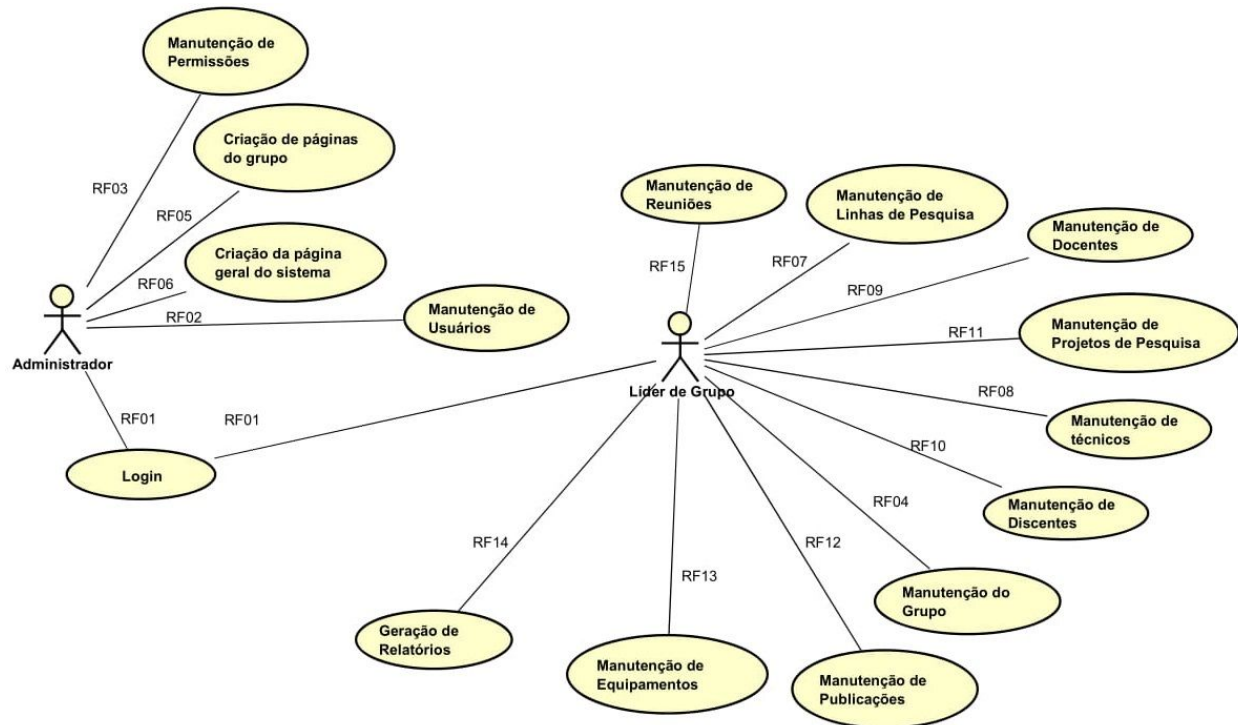


Figura 3 - Diagrama de Caso de Uso

O diagrama de caso de uso possui dois atores, Administrador e Líder do Grupo, e cada um tem seus casos de uso individuais com exceção do Login (RF01), comum a ambos.

### RF01 - Login

O usuário, para acessar a parte privada do sistema, deve se autenticar utilizando uma identificação de usuário e uma senha. Caso seja o primeiro acesso, o usuário deve, ao invés de se autenticar, cadastrar os dados sobre o Administrador do sistema, que é o usuário que tem acesso às configurações de manutenção geral do sistema.

### RF02 - Manutenção de usuários

O administrador pode inserir, ver, editar e excluir as informações sobre os usuários do sistema. Caso o usuário esqueça sua senha, existe a opção de recuperar a senha por meio de um e-mail enviado pelo sistema.

#### RF03 - Manutenção de permissões

Há uma interface que permite a manutenção (inserir, ver, editar e excluir) dos privilégios dos usuários Administrador e Líderes de Grupo, onde podem ser configurados os acessos que estes usuários possuem.

#### RF04 - Manutenção do grupo

O líder do grupo pode inserir, ver, editar e excluir as informações sobre seu grupo.

#### RF05 - Criação de páginas do grupo

Após cadastrar as informações sobre o grupo, o sistema deve gerar a página pública e a página privada do grupo, utilizando um leiaute padrão.

#### RF06 - Criação da página geral do sistema

Depois de cadastrar grupos, o Administrador pode gerar a página inicial do sistema, onde estarão listados todos os grupos existentes com seus nomes, descrições e links.

#### RF07 - Manutenção de linha de pesquisa

O líder de grupo pode inserir, ver, editar e excluir as informações sobre as linhas de pesquisa que seu grupo segue.

#### RF08 - Manutenção de técnicos

O líder de grupo pode inserir, ver, editar e excluir as informações sobre os técnicos que participam de seu grupo de pesquisa.

#### RF09 - Manutenção de docentes

O líder de grupo pode inserir, ver, editar e excluir as informações sobre os docentes que participam de seu grupo de pesquisa e das linhas de pesquisa que eles seguem.

#### RF10 - Manutenção de discentes

O líder de grupo pode inserir, ver, editar e excluir as informações sobre os alunos que participam de seu grupo de pesquisa.

#### RF11 - Manutenção de projetos de pesquisa

O líder de grupo pode inserir, ver, editar e excluir as informações sobre os projetos de pesquisa de seu grupo de pesquisa.

#### RF12 - Manutenção de publicações

O líder de grupo pode inserir, ver, editar e excluir as informações sobre as publicações referentes aos projetos de pesquisa produzidos por seu grupo de pesquisa.

#### RF13 - Manutenção de equipamentos

O líder de grupo pode inserir, ver, editar e excluir as informações sobre os equipamentos pertencentes a seu grupo de pesquisa.

#### RF14 - Geração de relatórios

O líder de grupo pode selecionar um ano de escopo para gerar relatórios de formato pré-definido sobre os dados relativos ao ano escolhido.

#### RF15 - Manutenção de reuniões

O líder de grupo pode inserir, ver, editar e excluir as informações sobre as reuniões que irão ocorrer ou já ocorreram em seu grupo de pesquisa.

## REQUISITOS NÃO FUNCIONAIS DE SISTEMA

Descrevem as propriedades gerais do sistema de forma completa, indicando características e métricas utilizadas, quando necessário.

### RNF01 – Utilização

O sistema é baseado em *Web*, o que significa que qualquer dispositivo com um navegador e acesso à Internet poderá utilizá-lo. Como grande parte do processamento irá ocorrer no lado do servidor, também não demanda grande poder computacional da parte do dispositivo de acesso.

### RNF02 – Segurança das informações

O sistema utiliza um Sistema Gerenciador de Banco de Dados (SGBD) para guardar as informações persistentes da aplicação. O SGBD em questão será o MySQL, gratuito e livre para utilização. Este sistema gerenciador possui configurações de segurança padrão. O tráfego de dados será criptografado usando o padrão OpenPGP.

### RNF03 – Compatibilidade

O sistema poderá ser acessado por estacionários e mobile, como a linguagem utilizada foi web é executado independente de sistema operacional, apenas necessitando de um navegador.

### RNF04 – Disponibilidade dos dados

O sistema garante a disponibilidade de seus dados de maneira ininterrupta supondo que o usuário tenha acesso à internet.

### RNF05 - Privacidade

Nenhuma informação privada será publicada sem o consentimento prévio do usuário, o que é garantido pelo RNF02.

### RNF06 - Facilidade de uso

O sistema possui uma interface simples e fácil de usar. Todas as informações estão a no máximo quatro cliques do usuário e o layout é padronizado sobre o local de botões, campos e outros tipos de entrada.

#### RNF07 - Desempenho

O sistema deve gerar e disponibilizar relatórios em menos de vinte segundos. Em um computador com processador Quad-Core de 2.8GHz, 4 GB de Memória RAM, conexão à internet banda larga de 10 Mbps constantes e usando o navegador Google Chrome deve ser capaz de acessar uma página do sistema em menos de cinco segundos.

#### RNF08 - Fácil manutenção

O código segue boas práticas de programação e o padrão Camel Case, além de ser bem comentado para auxiliar futuras manutenções.

## CRONOGRAMA DE EXECUÇÃO

O cronograma de execução das tarefas está distribuído na Tabela 1, onde se discrimina o requisito funcional a ser entregue na semana, o profissional responsável pela entrega do requisito, os pré-requisitos para a entrega deste requisito na semana informada, e o total de horas prevista para o desenvolvimento do requisito.

<b>Incremento</b>	<b>Requisito</b>	<b>Responsável</b>
1	RF01	Gabriel
1	RF02	Juliane
1	RF03	Thales
2	RF04	Gabriel
2	RF05	Juliane
2	RF06	Thales
3	RF07	Gabriel
3	RF08	Juliane
3	RF09	Thales
4	RF10	Gabriel
4	RF11	Juliane
4	RF12	Thales
4	RF13	Gabriel
5	RF14	Juliane
5	RF15	Thales

Tabela 1 – Planejamento semanal de entregas

## **PLANEJAMENTO DE TESTES E MANUTENÇÃO**

Durante o desenvolvimento do sistema, serão realizados testes de integração para encontrar erros em sua execução.

Ao fim do desenvolvimento de cada requisito funcional, serão realizados testes de unidades, para verificar o seu funcionamento.

Após o término do desenvolvimento, serão realizados testes de releases para encontrar erros no sistema. Tais testes incluem:

- Operações com campos que ultrapassem o limite de caracteres permitido no banco de dados;
- Operações com dados obrigatórios faltando;
- Operações com o servidor do banco de dados desligado;
- Operações com data de término prévia à data de início;
- Operações em itens inativos ou finalizados (como inserir aluno em projeto já finalizado).



## APÊNDICE

Inicialmente, os requisitos principais foram levantados através de entrevistas com o cliente. A primeira entrevista tratou de explicar de maneira generalizada tudo o que ele gostaria que o sistema fizesse. Nas entrevistas seguintes, foram esclarecidas dúvidas e especificados requisitos de maneira mais concreta. Como requisitado, neste primeiro momento não foi feito o levantamento dos Requisitos Funcionais de Sistema, que serão discutidos nos incrementos futuros.

Para o levantamento dos requisitos não-funcionais foi utilizada a Tabela 2, apresentada abaixo:

<b>Característica</b>	<b>Essencial</b>	<b>Muito importante</b>	<b>Importante</b>	<b>Indiferente</b>	<b>Sem necessidade</b>
<b>Velocidade</b>			<b>x</b>		
<b>Compatibilidade</b>	<b>x</b>				
<b>Segurança</b>	<b>x</b>				
<b>Facilidade de uso</b>		<b>x</b>			
<b>Fácil manutenção</b>				<b>x</b>	
<b>Facilidade de instalação</b>					<b>x</b>

Tabela 2 - Questionário de avaliação de requisitos

Esta tabela teve como intuito precisar exatamente quais requisitos não-funcionais o cliente mais necessita em seu sistema, sem permitir que a todos fosse dada a mesma importância, o que acarretaria em um maior tempo necessário para o desenvolvimento. O cliente teve a oportunidade de escolher dois requisitos como “Essencial” e um para cada outro grau de importância.