

Documentação Software

Agendamento de Salas

Versão 1.0

Eduardo Pedroso
Pedro Paes
Marcio Maranhão
André Felipe

Prefácio

Este documento tem como objetivo fornecer informações relevantes sobre o sistema de agendamento de salas do IFSP campus Salto, referentes a versão 1.0 do software. A tecnologia escolhida para desenvolvimento foi Ruby com Rails e PostgreSQL como base de dados. O sistema conta também com o controle de versão e deploy automático com Git/GitHub e Heroku.

Todas as informações aqui contidas podem ser encontradas junto ao código fonte do sistema no link <https://github.com/drdpedroso/agendamento-sala-ifsp> .

1. Introdução ao Documento

1.1. Tema

O projeto tem como tema a automatização do agendamentos de salas, laboratórios e do auditório do IFSP campus Salto.

1.2. Objetivo do Projeto

Automatização no agendamento de salas, laboratórios e do auditório do IFSP Campus Salto, diminuindo margem de erro e conflitos nos mesmos.

1.3. Delimitação do Problema

O projeto se resume a um calendario baseado no Google Calendar para efetuar os agendamentos das salas. Utilizamos um sistema de login e cadastro com níveis de usuário e diferentes níveis de permissão para efetuar os agendamentos.

1.4. Justificativa da Escolha do Tema

A escolha do tema se dá justamente pela capacidade de se utilizar Orientação a objetos na aplicação do mesmo, além de possíveis padrões de projetos como MVC ou DAO. A POO se tornou versátil pois precisávamos de diversas classes para controlar usuários, salas e agendamentos além de classes para controlar interações com o Banco de Dados.

1.5. Método de Trabalho

Escolhemos a Orientação a Objetos e o padrão MVC(Model, View, Controller) para desenvolvermos o projeto simplesmente pelo fato de ser o recomendado pela comunidade e pelos desenvolvedores do framework de trabalho (Ruby On Rails). A convenção nos ajudou bastante nesse caso específico, onde nos adaptamos facilmente a tecnologia. No desenvolvimento escolhemos a metodologia do Git, onde cada desenvolvedor criava uma branch para trabalhar e um deles(com maior conhecimento da tecnologia e da plataforma) aprovava o código.

1.6. Organização do Trabalho

O documento se organiza em: Primeiro Checkout do sistema e como realizar funções básicas no mesmo via navegador.

2. Descrição Geral do Sistema

2.1. Descrição do Problema

Muitas vezes ocorriam agendamentos simultâneos feitos por professores diferentes na mesma sala/laboratório, o que prejudicava os alunos e tomava tempo de aula. Com a utilização do sistema, todos podem ver os agendamentos e escolher salas disponíveis que atendam as suas necessidades, sem perda de tempo.

2.2. Principais Envolvidos e suas Características

2.2.1 Usuario do Sistema

O sistema se destina principalmente a docentes e servidores que necessitem agendar salas ou laboratórios para quaisquer atividades.

2.2.2 Desenvolvedores do Sistema

O software foi desenvolvido por alunos do 5º semestre do curso de Análise e Desenvolvimento de Sistemas do IFSP campus Salto. O

responsável pelos deploys e pelo repositório do sistema foi o aluno Eduardo Pedroso.

2.3. Regras de Negócio

O sistema deve rodar em plataforma Web. O único limite atual da arquitetura é devido ao servidor, que não suporta mais de 12h de acesso concorrente. Como ferramenta de apoio temos o GitHub para possíveis resgates de versão caso a atual se desestabilize.

3. Descrição Geral do Sistema

3.1. Requisitos Funcionais

RF1	Criação de diferentes tipos de usuários (Administradores ¹ e Usuários Comuns ²)
RF2	Ter sistema de <i>login</i> com usuário e senha.
RF3	Disponibilizar a visualização do calendário de reserva do auditório e de todos os laboratórios e salas .
RF4	Possibilitar a reserva dos espaços supracitados.
RF5	Avisar, através de <i>e-mail</i> , Administradores e Usuários, sobre o pedido de reservas e confirmação de reservas

¹ - Usuário com permissão de **criar novos usuários** e por **aceitar, rejeitar ou revogar** os pedidos de reserva.

² - Usuário com permissão de visualizar o calendário/reservas e pedir a reserva de espaços.

3.2. Requisitos Não-Funcionais

Navegadores: Google Chrome ou Mozilla Firefox. Não compatível com Internet Explorer ou Edge. Sem testes efetuados no Opera.

Conexão com a internet ou um servidor local com Ruby e PostgreSQL instalado.

3.3. Protótipo

Agendamento de sala IFSP Nova agendamento Cadastrar [Logout](#)

< June 2016 >

DOM	SEG	TUE	WED	THU	FRY	SAB
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2

Novo agendamento

Evento

Descrição

Data início

2016 July 1 00:05

Data final

2016 July 1 00:05

Sala

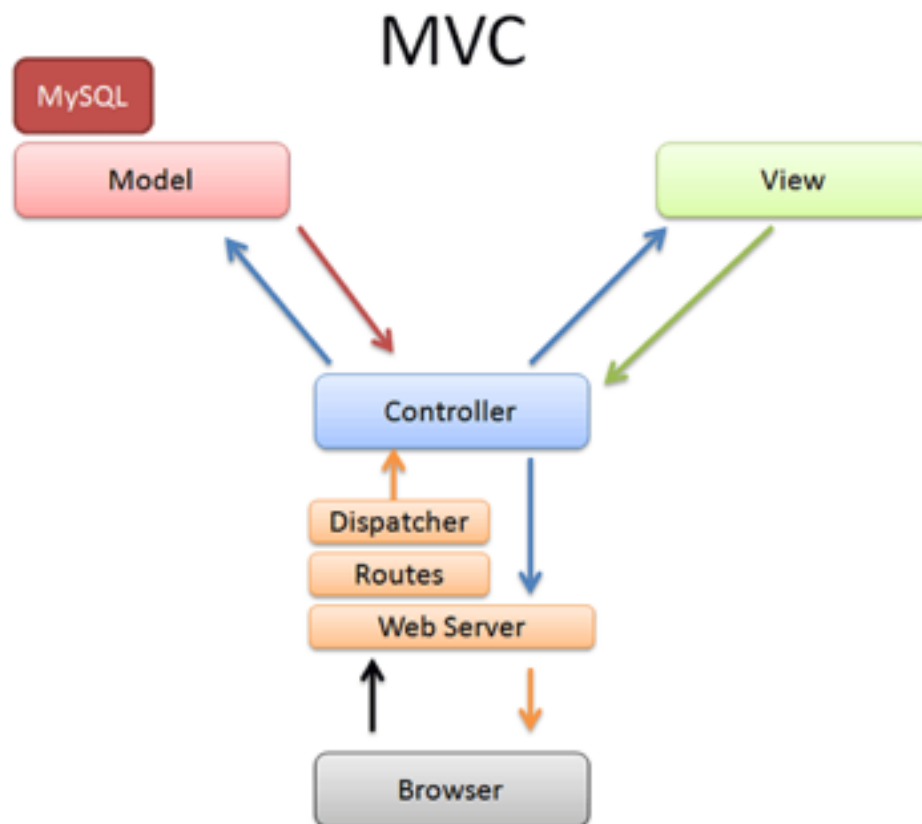
Sala 310

[Criar agendamento](#)

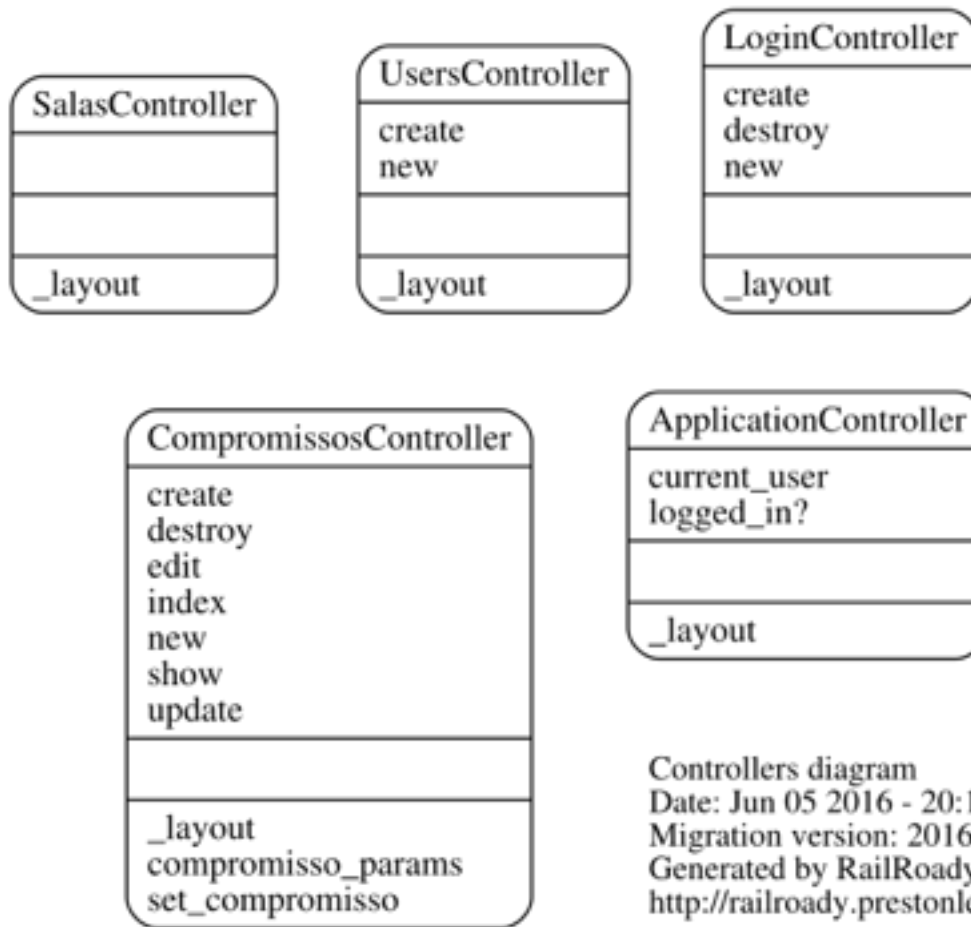
[Voltar](#)

4. Análise e *Design*

4.1. Arquitetura do Sistema



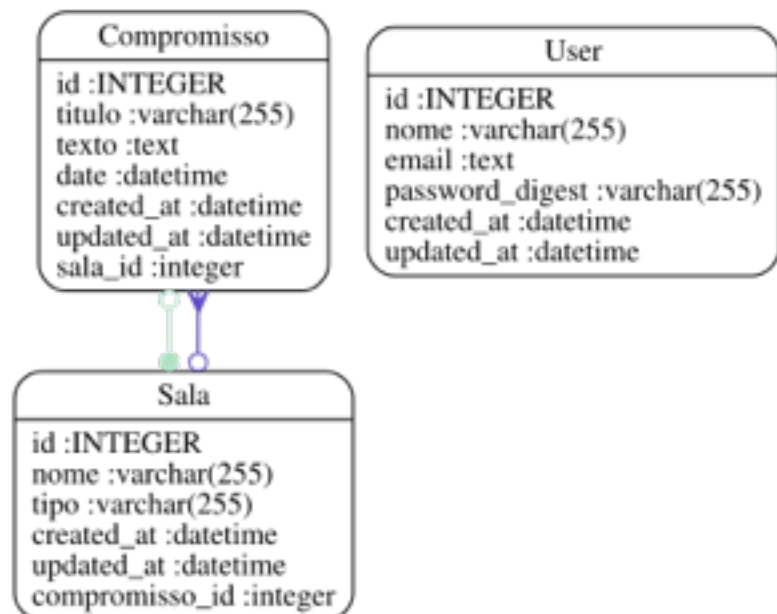
4.2. Diagrama de *Controllers*



4.3. Diagrama de *Models*

Models diagram

Date: Jun 05 2016 - 20:17
Migration version: 20160526010928
Generated by RailRoady 1.4.2
<http://railroady.prestonlee.com>



4.4. Ambiente de Desenvolvimento

Para desenvolvimento utilizamos o Ruby Mine como IDE para trabalhar com Ruby (linguagem de programação) e Rails (framework). Em ambiente de desenvolvimento, o banco utilizado foi o SQLite3. A maior parte do sistema foi desenvolvida em sistemas Mac OSX ou Ubuntu. O Git foi amplamente usado para controlar versão em conjunto com o GitHub e o Heroku para fazer deploy automático da nossa aplicação para um servidor de homologação onde os testes eram realizados com o PostgreSQL como Base de dados.

4.4. Sistemas e componentes externos utilizados

Utilizamos Git e GitHub para controle de versão e Heroku para deploy da aplicação em um servidor de homologação. Além disso, utilizamos o RailRoady para geração dos diagramas de classes.

5. Testes

5.1. Plano de Testes

Realizamos os seguintes testes:

- Criação de agendamento com e sem super-usuário para verificar a visão de cada um.
- Deletamos e editamos os agendamentos com e sem super-usuário, onde apenas o dono(ou o usuário com permissão) poderia realizar essas alterações
- Efetuamos o login com um email/senha valido e com email/senha invalidos.
- Cadastramos um email unico e depois tentamos cadastra-lo mais uma vez
- Tentamos acessar rotas sem estar logado

Todos os testes ocorreram como esperado com resultados positivos.

6. Manual do Usuario

1 - Logar com superuser.

root@root.com
senha123

2 - Na parte superior (header), clique em Cadastrar para criar um novo usuário e em Novo Agendamento para criar um novo agendamento.

3 - Criando um agendamento: Preencha os campos pedidos no formulário e clique em criar agendamento. Uma tela de confirmação deverá ser exibida.

4 - Cadastrando novo usuário: Preencha os campos do formulário. O email deve ser único no sistema e as senhas devem ser iguais nos dois campos(criação e confirmação). Caso tudo ocorra bem, voce será redirecionado para a tela de calendário.

5 - Você pode editar/cancelar eventos que você criou e que outros tenham criado(caso você seja superusuário) apenas clicando no menu que fica no dia em que o evento foi agendado.

O tutorial em video e a documentação completa podem ser vistos no link:
<https://github.com/drdpedroso/agendamento-sala-ifsp>