

# Wiki UFC

## Apresentação da Aplicação

Álison Santos  
André Castro  
Adriano Freitas  
Luis Henrique  
Rafael Barbosa

28 de junho de 2007

1. Introdução
2. Metodologia
3. Tecnologias
4. Aplicação
5. Conclusão

## 1. Introdução

Problema atual

Solução proposta

Soluções alternativas

## 2. Metodologia

## 3. Tecnologias

## 4. Aplicação

## 5. Conclusão

# Problema atual

Hoje em dia, os alunos do nosso curso não possuem uma maneira eficiente de trocar informações sobre as disciplinas que estão cursando.

- Cada disciplina possui sua própria lista de emails
- Não existe um calendário oficial no qual os estudantes possam confiar
- É difícil compartilhar material como notas de aulas e listas de exercícios
- É difícil obter o material utilizado nos semestres anteriores

# Solução proposta

Uma aplicação WEB colaborativa onde os alunos poderão compartilhar informações sobre as disciplinas que estão cursando.

## Objetivos:

- Permitir a colaboração descentralizada de alunos para o curso
- Facilitar a divulgação do conhecimento
- Tornar a vida do estudante mais organizada

# Soluções alternativas

## WebAdmin

Sistema criado por Windson Viana para facilitar a manutenção de páginas pelos professores do curso de Computação. Em funcionamento no site <http://disciplinas.lia.ufc.br>

## Moodle

Moodle é um Software Livre de E-Learning, desenvolvido para facilitar a criação de cursos online não presenciais. Possui recursos avançados, como fóruns, blogs, wikis, chats, e quizzes.

1. Introdução
2. Metodologia
3. Tecnologias
4. Aplicação
5. Conclusão

Scrum é um método ágil, criado em 1993, para permitir que o desenvolvimento rápido de aplicações. Atualmente é utilizado em empresas como Google, Yahoo, Eletronic Arts, Philips e Nokia.

Como funciona?

- Manutenção de um [backlog](#), com as tarefas pendentes
- Desenvolvimento segmentado em ciclos, chamados de [sprints](#)
- Reuniões periódicas, ao final de cada sprint



## Active Tickets

Ticket	Summary	Component	Version	Milestone	Type	Owner	Created
#31	Geral: Documentação	geral	None	sprint8	task	somebody	06/11/07
#55	Geral: Apresentação	geral	None	sprint8	task	somebody	06/18/07
#29	Geral: Melhorar a segurança	administracao	None		task	bustamante	05/29/07
#39	Interface: Histórico do Wiki	interface	None	sprint8	task	andre	06/11/07
#40	User: Implementar preferências	geral	None	sprint8	task	rafael	06/11/07
#45	Wiki: Corrigir validação	geral	None		task	somebody	06/11/07
#46	Mural: Criar testes unitários e funcionais	geral	None		task	somebody	06/11/07
#47	Calendário: Criar testes unitários e funcionais	geral	None		task	somebody	06/11/07
#48	Wiki: Criar testes unitários e funcionais	geral	None		task	somebody	06/11/07
#49	User: Corrigir testes unitários	geral	None		task	somebody	06/11/07
#50	Shoutbox: Criar testes unitários e funcionais	geral	None		task	somebody	06/11/07
#51	Anexos: Criar testes unitários e funcionais	geral	None		task	somebody	06/11/07
#54	Mensagem: Mensagens pessoais	geral	None		task	rafael	06/11/07
#11	Wiki: Integração com o Latex	wiki	None	sprint8	task	adriano	05/09/07
#43	Interface: Página inicial do usuário	interface	None	sprint8	task	rafael	06/11/07
#44	Shoutbox: Excluir mensagens antigas	geral	None	sprint8	task	adriano	06/11/07
#37	Tuning: Criar indices no BD	geral	None		task	somebody	06/11/07
#41	Wiki: Utilizar o diff do Ruby	geral	None		task	somebody	06/11/07
#53	Geral: Implementar forum	geral	None		task	somebody	06/11/07
#52	Geral: Internacionalizar mensagens	geral	None		task	somebody	06/11/07

No nosso projeto,

- Cada sprint durava uma semana
- As reuniões eram realizadas segundas feiras, às 16h

Em cada reunião:

- Discutíamos o que havia sido implementado no sprint anterior
- Discutíamos o que não havia sido implementado, por algum motivo..
- Colocávamos novos itens no backlog
- Definíamos que tarefas cada um realizaria durante o próximo sprint

## 1. Introdução

## 2. Metodologia

## 3. Tecnologias

Linguagem Ruby

Ruby on Rails

Subversion e Trac

## 4. Aplicação

## 5. Conclusão

# Linguagem Ruby

Ruby é uma linguagem de programação orientada a objetos, criada por Yukihiro Matsumoto em 1995. Implementação oficial licenciada sob a GPL.

- Sintaxe simples e legível
- Completamente orientada a objetos
- Herança múltipla avançada, através de Mixins
- Introspecção, reflexão e meta-programação
- Garbage collector

# Ruby on Rails

Ruby on Rails é um framework para o desenvolvimento aplicações web, que segue os princípios *Don't Repeat Yourself (DRY)* e *Convention over Configuration*.



- Arquitetura Model-View-Controller
- Mapeamento objeto-relacional
- Scaffolds, Generators e Migrations
- Suporte nativo a testes automatizados
- Integração com Ajax e Webservices
- Servidor web incluso!

# Ruby on Rails

## A arquitetura MVC

Arquitetura para desenvolvimento que trabalha com três partes:

**Model** Dados armazenados da aplicação

**View** Seção visível ao usuário, e com a qual ele interage

**Controller** Interface entre Model e View, que estabelece as regras de negócio.

# Subversion e Trac

Subversion é um software open source para controle de versões.

- O código fonte é armazenado em um servidor web
- Cada programador pode ler e modificar os arquivos remotamente
- As versões antigas dos arquivos estão sempre disponíveis para consulta.

Trac é uma aplicação web para gerenciamento de projetos e bugtracking, com integração a Subversion.

1. Introdução

2. Metodologia

3. Tecnologias

4. Aplicação

Funcionalidades

Demonstração

5. Conclusão



Entre as funcionalidades implementadas estão:

- Um WIKI para conter as notas de aula
- Um calendário, para receber datas de provas e entregas de trabalho
- Um repositório de arquivos, onde serão armazenadas listas de exercícios, e provas antigas
- Um mural, para postagem de notícias sobre a disciplina

E, além das funcionalidades básicas:

- Wiki com suporte a sintaxe Markdown e controle de versões,
- Calendário com exportação nos formatos iCalendar e RSS,
- Shoutbox, que permite a comunicação rápida entre os usuários,
- Página pessoal, que integra as informações sobre as disciplinas cursadas pelo aluno,
- Site personalizável.

Demonstração...

1. Introdução

2. Metodologia

3. Tecnologias

4. Aplicação

5. Conclusão

Dificuldades Encontradas

Pontos Positivos

Perspectivas

# Dificuldades encontradas

As maiores dificuldades para realizar o projeto foram:

- Estudar a linguagem Ruby
- Aprender a utilizar os recursos do framework Ruby on Rails
- Conseguir tempo para programar, já que nós estamos cursando, pelo menos, quatro outras disciplinas
- Sincronizar o nosso método ágil com relatórios tradicionais da disciplina

# Pontos positivos

Sobre o projeto em geral:

- A equipe aprendeu a utilizar muitas tecnologias novas
- Aplicamos alguns conceitos da disciplina na prática

Sobre o método Scrum:

- Reuniões semanais: não ficou tudo pro fim!
- Backlog: Sabíamos exatamente o que ainda faltava ser feito

Para o futuro, planejamos:

- Implementar alguns itens que, por falta de tempo, não puderam ser inclusos no projeto original.
- Colocar em funcionamento, para os alunos puderem usar de verdade.
- Abrir como um projeto Open Source.

## Perguntas?