



Universidade de Brasília - UnB  
Faculdade UnB Gama - FGA  
Desenho de Software e Introdução à Computação Gráfica

## **Alke Bike**

Autor: Alke Games  
Orientador: Professora Milene Serrano e Professora Carla  
Rocha

Brasília, DF  
13 de abril de 2015





Alke Games

## **Alke Bike**

Relatório do jogo Alke Bike submetido na Faculdade UnB Gama da Universidade de Brasília.

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Orientador: Professora Milene Serrano e Professora Carla Rocha

Brasília, DF

13 de abril de 2015

# Lista de ilustrações

Figura 1 – Fluxograma das telas do jogo . . . . .	6
Figura 2 – Esquema de controle para o controle de XBox 360 . . . . .	7
Figura 3 – Diagrama de sequência, três primeiros cenários . . . . .	15
Figura 4 – Diagrama de sequência, quarto cenário . . . . .	16
Figura 5 – Diagrama de sequência, quinto cenário . . . . .	16

# Lista de tabelas

Tabela 1 – Tarefas do plano de iteração (1) . . . . .	19
Tabela 2 – Tarefas do plano de iteração (2) . . . . .	20

# Sumário

<b>1</b>	<b>GAME DESIGN DOCUMENT . . . . .</b>	<b>5</b>
<b>1.1</b>	<b>Objetivo . . . . .</b>	<b>5</b>
<b>1.2</b>	<b>Organização do Time . . . . .</b>	<b>5</b>
<b>1.3</b>	<b>Modos de Jogo . . . . .</b>	<b>5</b>
1.3.1	Endurance (Resistência) . . . . .	5
1.3.2	Dexterity (Destreza) . . . . .	6
<b>1.4</b>	<b>Telas . . . . .</b>	<b>6</b>
<b>1.5</b>	<b>Sistema de Pontuação . . . . .</b>	<b>6</b>
<b>1.6</b>	<b>Controle . . . . .</b>	<b>7</b>
<b>1.7</b>	<b>Tecnologias . . . . .</b>	<b>7</b>
<b>2</b>	<b>CENÁRIOS E LÉXICOS . . . . .</b>	<b>9</b>
<b>2.1</b>	<b>Cenários . . . . .</b>	<b>9</b>
2.1.1	Cenário 1 . . . . .	9
2.1.2	Cenário 2 . . . . .	9
2.1.3	Cenário 3 . . . . .	10
2.1.4	Cenário 4 . . . . .	10
2.1.5	Cenário 5 . . . . .	10
<b>2.2</b>	<b>Léxicos . . . . .</b>	<b>10</b>



<b>3</b>	<b>ESPECIFICAÇÕES SUPLEMENTARES . . . . .</b>	<b>13</b>
<b>3.1</b>	<b>Requisitos de Usabilidade . . . . .</b>	<b>13</b>
<b>3.2</b>	<b>Requisitos de Desempenho . . . . .</b>	<b>13</b>
<b>3.3</b>	<b>Requisitos de Suportabilidade . . . . .</b>	<b>13</b>
<b>4</b>	<b>DIAGRAMA DE SEQUÊNCIA . . . . .</b>	<b>15</b>
<b>5</b>	<b>PLANO DE ITERAÇÃO . . . . .</b>	<b>17</b>
<b>5.1</b>	<b>Introdução . . . . .</b>	<b>17</b>
5.1.1	Objetivo . . . . .	17
5.1.2	Escopo . . . . .	17
5.1.3	Visão Geral . . . . .	17
<b>5.2</b>	<b>Plano . . . . .</b>	<b>17</b>
5.2.1	Recursos . . . . .	17
5.2.2	Cenários . . . . .	17
5.2.3	Critérios de Avaliação . . . . .	18



# 1 Game Design Document

## 1.1 Objetivo

*Alke Bike* é um jogo de corrida de bicicleta 3D com elementos do estilo *arcade* que se passa nas Olimpíadas de 2016 no Rio de Janeiro. O atleta deve se esforçar na corrida para conseguir a medalha de ouro.

O foco do jogo será exclusivamente na corrida em si, tendo como inspiração jogos como *Don't Tap the White Tile*, *Heart Jump* e *Timberman*. O atleta tem que se esforçar para se manter equilibrado na bicicleta enquanto tenta vencer. O jogo possui dois modos, mas ambos no contexto de corrida.

## 1.2 Organização do Time

O time é composto por quatro integrantes sendo que cada um deles lidera as atividades de um papel diferente, porém todos atuam em todos os papéis. Os integrantes são:

- **Caio Nardelli (@CaioIcy):** Programador.
- **João Paulo Ribeiro (@JohnnysRibeiro):** Game Designer.
- **Matheus Godinho (@MatheusGodinho):** Gerente.
- **Simião Carvalho (@simiaosimis):** Programador/Game Designer.
- **Fagner Rodrigues (@fanig01):** Programador.

## 1.3 Modos de Jogo

### 1.3.1 Endurance (Resistência)

O atleta deve pedalar até onde conseguir, com a velocidade sendo gradativamente mais alta.

Os blocos de comando vêm em uma velocidade que vai aumentando conforme o jogador os acerta. A corrida só se encerra quando o atleta se desequilibra completamente na bicicleta.



### 1.3.2 Dexterity (Destreza)

O atleta deve pedalar uma distância fixa o mais rápido possível.

Os blocos de comando vêm a medida que o jogador acerta os anteriores. Sendo assim, o objetivo deste modo é completar a corrida o mais rápido possível, sem se desequilibrar. A corrida se encerra quando acabarem os blocos de comando.

## 1.4 Telas

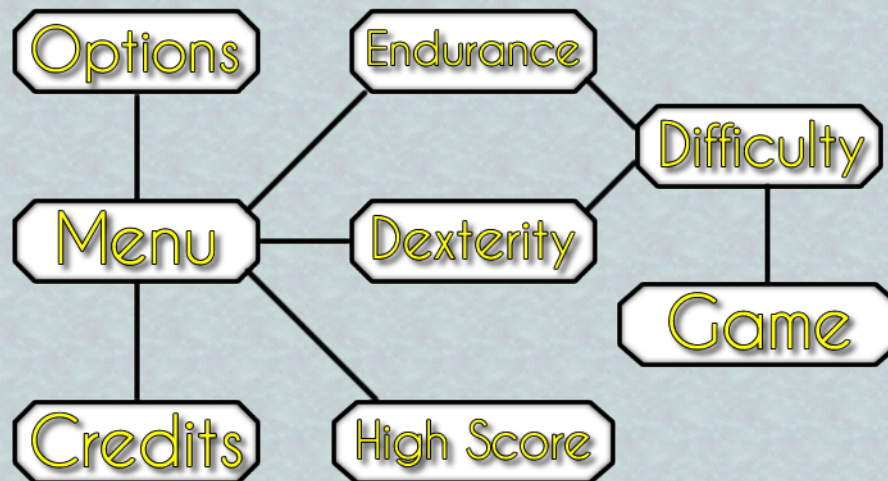


Figura 1 – Fluxograma das telas do jogo

## 1.5 Sistema de Pontuação

Para cada modo de jogo, a pontuação é feita de uma determinada forma, assim contemplando as diferentes características dos modos *Endurance* e *Dexterity*.

### Modo *Endurance*

No modo *Endurance*, a pontuação do jogador é feita de acordo com o tempo em que ele se manteve acertando os comandos.

### Modo *Dexterity*

No modo *Dexterity*, a pontuação é feita de acordo com a velocidade em que o jogador acertou os comandos. A quantidade de comandos é determinada de acordo com a dificuldade



## 1.6 Controle

O jogo tem controles muito simples, e o jogador poderá usar o teclado ou um controle de *XBox 360* ou de *Playstation 4*. Um exemplo do esquema dos controles está na figura a seguir.



Figura 2 – Esquema de controle para o controle de XBox 360

## 1.7 Tecnologias

- Linguagem de programação: **C++**

Sua alta performance e qualidade, segue na liderança disparada em produção de jogos.

- Compilador: **GNU Compiler Collection (gcc)**

Confiável e já com bom suporte ao padrão C++14.

- Controle de versão de código e documentação: **Git (GitHub)**

Na liderança popular entre os forges de Git por sua qualidade e poder de socialização. O repositório pode ser acessado em [github.com/CaioIcy/ICG\\_OpenGL](https://github.com/CaioIcy/ICG_OpenGL).

- Editor de texto: **Sublime Text**

Preferido pelos desenvolvedores da equipe.

- Gerador de documentação: **Doxygen**

Um gerador de documentação excelente, e 100% compatível com C++. A documentação do código será hospedada *online* em uma *GitHub page*.

- Sistema operacional de desenvolvimento: **Linux Mint 64-bit**





## 2 Cenários e Léxicos

### 2.1 Cenários

Nessa seção, serão descritos os cenários do jogo Alke Bike.

#### 2.1.1 Cenário 1

**Título:** Dinâmica do Jogo no modo *Endurance*

**Objetivo:** Descrever as regras do Alke Bike no modo *Endurance*

**Contexto:** Início de Jogo

**Atores:** Jogador

**Recursos:** Personagem, *Bloco de Comando*

**Episódios:** Jogador inicia o jogo. Jogador seleciona o nível de dificuldade. Aparece um tempo regressivo de preparação para o Jogador. Jogador seleciona o Bloco de Comando para pedalar caso o bloco esteja na *Área de Comando*. Caso selecione o Bloco correto, o bloco muda de cor temporariamente e o jogador seleciona o próximo bloco quando este estiver na Área de Comando sendo que os Blocos vão chegando na Área de Comando mais rápido de acordo com o tempo, caso contrário, o jogo acaba e o *score* final dele é mostrado. Caso o score seja o maior do Jogador o *high score* é registrado.

#### 2.1.2 Cenário 2

**Título:** Dinâmica do Jogo no modo *Dexterity*

**Objetivo:** Descrever as regras do Alke Bike no modo *Dexterity*

**Contexto:** Início de Jogo

**Atores:** Jogador

**Recursos:** Personagem, Bloco de Comando, Quantidade de Blocos de Comando

**Episódios:** Jogador inicia o jogo. Jogador seleciona o nível de dificuldade. Aparece um tempo regressivo de preparação para o Jogador. Jogador seleciona o Bloco de Comando para pedalar caso o Bloco esteja na Área de Comando. Caso o jogador acerte o bloco, o bloco muda de cor temporariamente e o próximo Bloco entra na Área de Comando. Caso o jogador erre o comando ou acabe a quantidade de Blocos a serem pressionados, o jogo acaba e o *score* final dele é mostrado. Caso o score seja o maior do Jogador o *high score* é registrado.



### 2.1.3 Cenário 3

**Título:** Iniciação de Jogo

**Objetivo:** Descrever como o jogo é iniciado

**Contexto:** Aplicativo começa a ser executado

**Atores:** Jogador

**Recursos:** Menu, Personagem e Blocos de Comando

**Episódios:** Jogador inicia o aplicativo. Jogador seleciona o botão Jogar. Jogador escolhe o modo de jogo. Jogador escolhe nível de dificuldade. Jogador inicia o jogo.

### 2.1.4 Cenário 4

**Título:** Visualização dos *high scores*

**Objetivo:** Descrever como o jogador visualiza os *high scores* obtidas por ele

**Contexto:** Aplicativo começa a ser executado

**Atores:** Jogador

**Recursos:** Menu, Lista de *high scores*

**Episódios:** Jogador inicia o aplicativo. Jogador seleciona o botão *High Scores*. Jogador escolhe o modo de jogo.

### 2.1.5 Cenário 5

**Título:** Visualização dos créditos do jogo

**Objetivo:** Descrever como o jogador assiste aos créditos finais do jogo

**Contexto:** Aplicativo começa a ser executado ou após a derrota do jogador.

**Atores:** Jogador

**Recursos:** Menu, Créditos finais

**Episódios:** Jogador inicia o aplicativo. Jogador escolhe o botão Créditos.

## 2.2 Léxicos

### Blocos de Comando

- **Noção:** São os comandos dados ao jogador para que ele avance no jogo. No jogo existirão apenas dois tipos de blocos de comando que serão representados por direita ou esquerda

- **Sinônimos:** Comando de Ação, Botão de Ação.
- **Impacto:** Os blocos de comando tem impacto direto no jogo pois se não forem pressionados na hora certa, o jogador perde.

### **Área de comando**

- **Noção:** É a área onde o jogador deve acionar o bloco de comando para prosseguir com o jogo.
- **Sinônimos:** Área de reação.
- **Impacto:** O impacto da Área de comando no jogo é direto pois é onde os blocos de comando tem que estar na hora que o jogador tiver que pressioná-los.

### **Score**

- **Noção:** É a pontuação final do jogador após o mesmo perder.
- **Sinônimos:** Pontuação.
- **Impacto:** A pontuação impacta na lista de *high scores* pois é lá onde ela é registrada caso ela seja uma das 10 melhores pontuações.

### **Dexterity**

- **Noção:** É um dos modos de jogo.
- **Sinônimos:** Não se aplica.
- **Impacto:** O impacto é nas regras do jogo que mudam de acordo com o modo.

### **Endurance**

- **Noção:** É um dos modos de jogo.
- **Sinônimos:** Não se aplica.
- **Impacto:** O impacto é nas regras do jogo que mudam de acordo com o modo.





## 3 Especificações Suplementares

As especificações suplementares serão divididas no modelo de requisitos FURPS+ sendo que os requisitos funcionais já foram contemplados no GDD e nos cenários.

### 3.1 Requisitos de Usabilidade

- O jogo não possuirá uma seção para ensinar o jogador como jogar, sendo que o próprio jogo ensinará o jogador conforme ele jogar;

### 3.2 Requisitos de Desempenho

- Os comandos tem que ter menos de 1 segundo de delay;
- O jogo deverá carregar em menos de 1 minuto;

### 3.3 Requisitos de Suportabilidade

- O jogo deverá ter suporte para controles de Xbox 360, PlayStation 3 e PlayStation 4;
- O jogo deverá rodar em máquinas com OpenGL 3.3 ou superior;



## 4 Diagrama de Sequência

Neste capítulo, serão apresentados os diagramas de sequência do jogo relacionados com os cenários previamente descritos.

O primeiro diagrama contempla os três primeiros cenários uma vez que os dois primeiros são muito parecidos pois explicam como funciona o jogo e o terceiro é uma generalização do início dos dois primeiros.

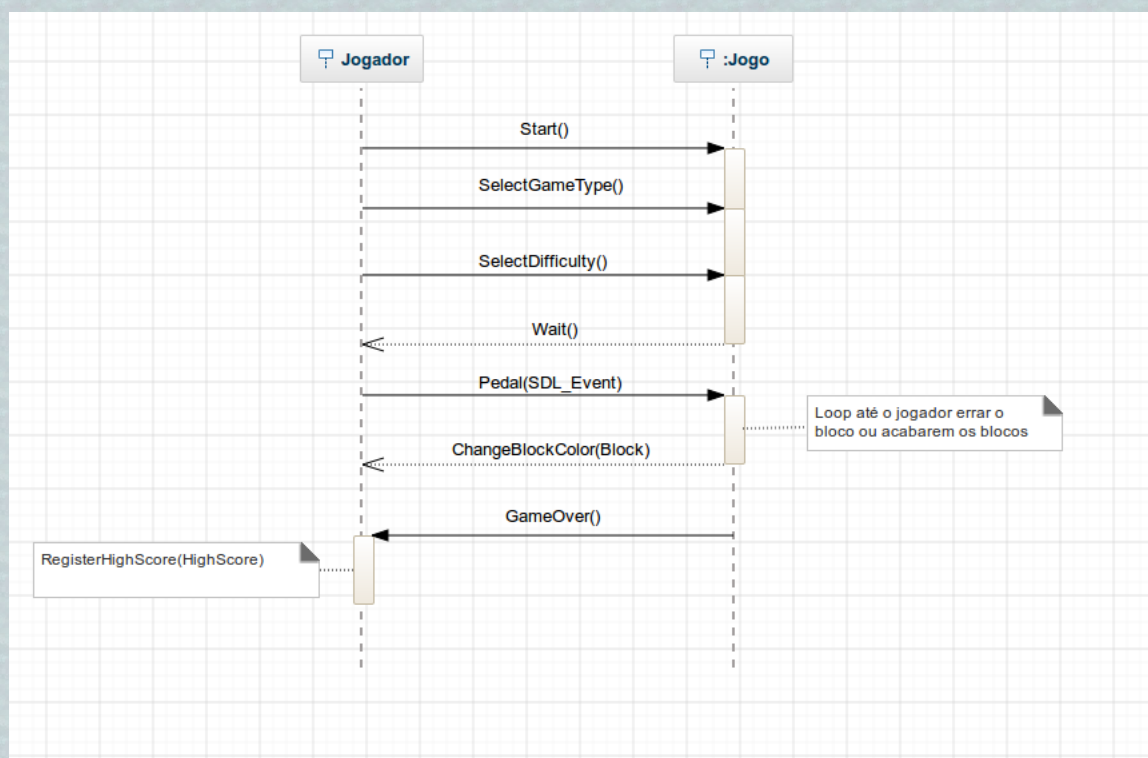


Figura 3 – Diagrama de sequência, três primeiros cenários

O segundo diagrama contempla o quarto cenário que explica como o jogador visualiza o *high score*.



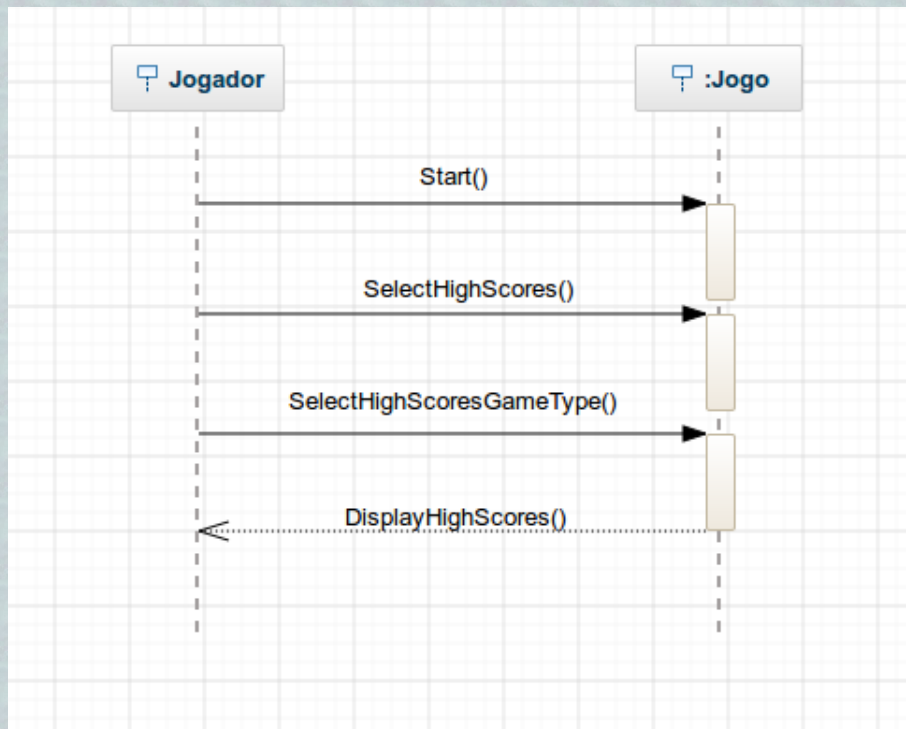


Figura 4 – Diagrama de sequência, quarto cenário

O terceiro diagrama contempla o quinto cenário que explica como o jogador assiste aos créditos finais.

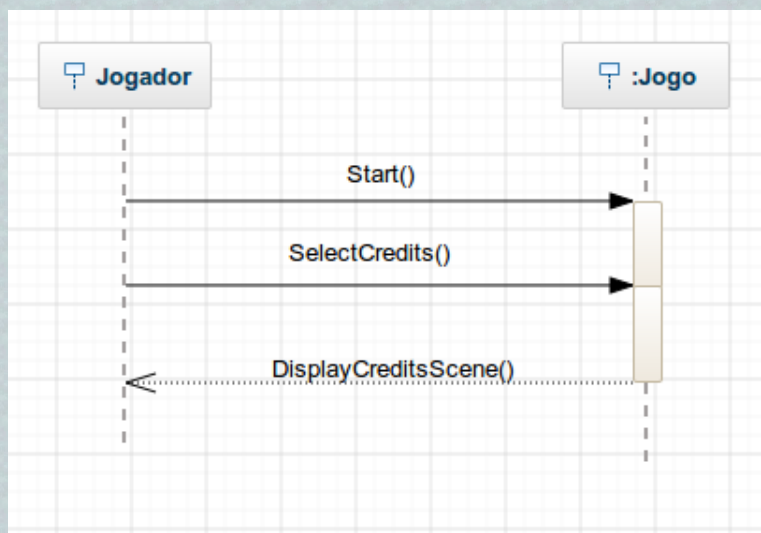


Figura 5 – Diagrama de sequência, quinto cenário

## 5 Plano de Iteração

### 5.1 Introdução

#### 5.1.1 Objetivo

Esse Plano de Iteração descreve os planos do desenvolvimento do jogo Alke Bike.

#### 5.1.2 Escopo

Esse Plano de Iteração especifica o planejamento do tempo e recursos nas fases de iniciação, elaboração, construção e transição do desenvolvimento do jogo Alke Bike pela empresa Alke Games.

#### 5.1.3 Visão Geral

O plano de Iteração explicita a distribuição dos recursos nas tarefas ao longo do tempo útil do projeto. Especificando as quatro fases do projeto (Concepção, Elaboração, Construção e Transição) e situando os envolvidos no andamento do projeto.

### 5.2 Plano

#### 5.2.1 Recursos

A distribuição dos recursos no projeto se dá conforme mostra a [Tarefas do plano de iteração](#).

#### 5.2.2 Cenários

Durante a Iteração de Concepção, todos os cenários serão identificados. Os objetivos, contexto, atores, recursos e episódios serão determinados e documentados nas Especificações dos cenários e Lexicos. A implementação dos cenários será iniciado na próxima iteração.

Na fase de Elaboração sera desenvolvido parte do cenário 1, Dinâmica do Jogo no modo *Endurance*, visando o desenvolvimento da arquitetura do jogo.

Todos os outros cenários serao desenvolvidos na fase de Construção, ver seção [Cenários](#).

Na Iteração de Transição todos os cenários já devem estar implementados e em fase final de teste de aceitação, para que o jogo seja distribuído para os possíveis jogadores.

### 5.2.3 Critérios de Avaliação

O principal objetivo da Iteração de Concepção é definir o sistema para o nível de detalhes requerido para uma boa compreensão do projeto a partir de uma perspectiva de viabilidade de desenvolvimento. Quando a iteração for concluída, uma revisão da Concepção com foco na qualidade, tempo e escopo chegará a uma decisão Aprovado / Não Aprovado para o projeto.

O objetivo principal da Iteração de Elaboração é concluir a análise e o design dos cenários selecionados e desenvolver parte majoritária da arquitetura de trabalho. O risco associado ao design de arquitetura ou o desempenho do sistema será percebido no final da iteração.

O objetivo principal da Iteração de Construção e desenvolver os cenários do jogo para sua release. Além disso deve ser observado a qualidade do jogo com base nos testes e na avaliação do código implementado.

O principal objetivo da Iteração de Transição é possibilitar ao jogador a instalação do jogo e obter a aceitação do jogo por parte dos jogadores.



Nome da tarefa	Início	Conclusão	Nomes dos Recursos
<b>Marcos</b>			
Início	09/03/2015	09/03/2015	
Fase de Concepção	16/03/2015	30/03/2015	
Fase de Elaboração	06/04/2015	06/04/2015	
Fase de Construção	09/03/2015	11/06/2015	
Fase de Transição	12/06/2015	19/06/2015	
<b>Fase de Concepção</b>			
<b>Modelagem do Negócio</b>			
Capturar vocabulário comum	16/03/2015	30/03/2015	João
Definir cenários	16/03/2015	23/03/2015	João
Especificar cenários e léxicos	24/03/2015	30/03/2015	João
<b>Requisitos</b>			
Desenvolver o GDD	16/03/2015	30/03/2015	Caio, João, Matheus, Simião
Priorizar cenários	24/03/2015	30/03/2015	Simião
Definir restrições do sistema	16/03/2015	30/03/2015	Matheus
<b>Gerenciamento de Configuração</b>			
Estabelecer práticas de GC	16/03/2015	30/03/2015	Caio
Estabelecer ambiente de GC	16/03/2015	30/03/2015	Caio
<b>Fase de Elaboração</b>			
<b>Modelagem do Negócio</b>			
Elaborar modelo de domínio	06/04/2015	10/04/2015	Matheus
Desenvolver diagrama de sequência	06/04/2015	13/04/2015	João
<b>Análise e Design</b>			
Análise priorizada dos cenários	06/04/2015	13/04/2015	Caio, João, Matheus, Simião
Análise de arquitetura	06/04/2015	13/04/2015	Caio, Simião
<b>Implementação de arquitetura</b>			
Estruturar implementação de arquitetura	06/04/2015	13/04/2015	Caio, Simião
Testar implementação de arquitetura	06/04/2015	13/04/2015	Caio, Simião
<b>Gerenciamento</b>			
Atualizar plano de iteração	06/04/2015	13/04/2015	João, Matheus
Avaliar iteração	13/04/2015	13/04/2015	Caio, João, Matheus, Simião

Tabela 1 – Tarefas do plano de iteração (1)

Nome da tarefa	Início	Conclusão	Nomes dos Recursos
<b>Fase de Construção</b>			
<b>Análise e Design</b>			
Refinar arquitetura	14/04/2015	11/06/2015	Caio, Simião, Fagner
Design priorizado dos cenários	14/04/2015	11/06/2015	João, Matheus
<b>Implementação</b>			
Implementar cenários priorizados	14/04/2015	11/06/2015	Caio, João, Matheus, Simião, Fagner
Revisar e refatorar código	14/04/2015	11/06/2015	Caio, João, Matheus, Simião, Fagner
<b>Testar</b>			
Planejar testes	14/04/2015	01/06/2015	Caio, João, Matheus, Simião
Teste de jogabilidade	01/06/2015	11/06/2015	João, Matheus, Fagner
Teste de integração	14/04/2015	11/06/2015	Caio, Simião
<b>Gerenciamento</b>			
Atualizar plano de iteração	14/04/2015	11/06/2015	Caio, João, Matheus, Simião, Fagner
Avaliar iteração	01/06/2015	11/06/2015	Caio, João, Matheus, Simião, Fagner
<b>Fase de Transição</b>			
<b>Implantação</b>			
Scripts de instalação	12/06/2015	19/06/2015	Caio, João, Matheus, Simião, Fagner
Empacotar software	12/06/2015	19/06/2015	Caio, João, Matheus, Simião, Fagner
Distribuir software	12/06/2015	19/06/2015	Caio, João, Matheus, Simião, Fagner
Testes de aceitação	12/06/2015	19/06/2015	

Tabela 2 – Tarefas do plano de iteração (2)