



UNIVERSIDADE FEDERAL DO CEARÁ  
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA  
CURSO DE GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

**Tiago Augusto da Silva Bencardino**

# **Estudo comparativo das plataformas Android e iOS para aplicações RESTful**

FORTALEZA – CEARÁ  
FEVEREIRO 2013

**Autor:**

Tiago Augusto da Silva Bencardino

**Orientador:**

Prof. Dr. José Marques Soares

Estudo comparativo das plataformas Android e iOS para aplicações  
RESTful

Monografia de Conclusão de Curso apresentada à Coordenação do Curso de Pós-Graduação em Engenharia de Teleinformática da Universidade Federal do Ceará como parte dos requisitos para obtenção do grau de **Engenheiro de Teleinformática**.

FORTALEZA – CEARÁ

FEVEREIRO 2013

TIAGO AUGUSTO DA SILVA BENCARDINO

**Estudo comparativo das plataformas Android e iOS para aplicações  
RESTful**

Esta Monografia foi julgada adequada para a obtenção do diploma de Engenharia do Curso de Graduação em Engenharia de Teleinformática da Universidade Federal do Ceará.

---

Tiago Augusto da Silva Bencardino

Banca Examinadora:

---

Prof. Dr. José Marques Soares  
Orientador

---

Prof. p1

---

Prof. p2

---

Prof. p3

Fortaleza, 3 de janeiro de 2013

# Resumo

Um grande número de aplicações *Web* têm, para as plataformas móveis *iOS* e *Android*, uma versão para *mobiles*, onde as mesmas informações são compartilhadas na *núvem*. Algumas redes sociais populares, como *Twitter* e *Foursquare*, possuem interfaces de comunicação REST como um serviço para outras aplicações. Este trabalho tem como objetivo fazer um estudo comparativo entre aplicações *iOS* e *Android* que utilizam um aplicativo web RESTful. Para realizar o estudo, é criada uma pequena rede social de criação de *quizzes*, utilizando *Ruby on Rails* e um serviço de *SaaS*, como o *Heroku*.

**Palavras-chave:** *iOS*, *Android*, *Mobile*, *Ruby on Rails*, REST, RESTful

# Abstract

A great number of web applications has, for mobile platforms *iOS* and *Android*, some version for mobile systems, which same information is shared on cloud. Some popular social networks like *Twitter* and *Foursquare* has communication interfaces REST as a service to other applications. This paper aims to make a comparative study between iOS and Android applications that use a RESTful web application. To conduct the study, it created a small social network to create quizzes using *Ruby on Rails* and a SaaS service like *Heroku*.

**Keywords:** iOS, Android, Mobile, Ruby on Rails, REST, RESTful.

Dedico este trabalho a uma galera ai.

*... Cada sonho que você deixa pra trás, é um pedaço do seu futuro que deixa de  
existir.*

Steve Jobs

# Sumário

<b>Lista de Figuras</b>	<b>vii</b>
<b>Lista de Tabelas</b>	<b>viii</b>
<b>Lista de Siglas</b>	<b>ix</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	1
1.2 Objetivos . . . . .	1
1.2.1 Objetivos Gerais . . . . .	1
1.2.2 Objetivos Específicos . . . . .	1
1.3 Organização do Texto . . . . .	1
<b>2 Fundamentação teórica</b>	<b>2</b>
2.1 Introdução . . . . .	2
2.2 SaaS . . . . .	2
2.3 REST . . . . .	2
2.3.1 RESTful . . . . .	2
2.4 JSON . . . . .	3
2.5 Rails . . . . .	3
2.5.1 Visão Geral . . . . .	3
2.6 iOS . . . . .	3
2.6.1 Visão Geral . . . . .	3
2.6.2 Ciclo de vida . . . . .	3
2.7 Android . . . . .	4
2.7.1 Visão Geral . . . . .	4
2.7.2 Ciclo de vida . . . . .	4
2.8 Resumo do Capítulo . . . . .	4
<b>3 Aplicação</b>	<b>5</b>
3.1 Visão geral . . . . .	5
3.2 Requisitos . . . . .	6
3.2.1 Requisitos funcionais . . . . .	6
3.2.2 Requisitos não-funcionais . . . . .	6
3.3 Casos de uso . . . . .	6
3.4 Diagrama de classes . . . . .	7
3.5 Diagrama de entidades . . . . .	7



<b>4</b>	<b>Implementação web</b>	<b>8</b>
4.1	Criação de aplicação . . . . .	8
4.2	Rotas . . . . .	8
4.3	Autenticação . . . . .	8
4.4	Front-End com bootstrap . . . . .	8
4.5	Deploy . . . . .	8
<b>5</b>	<b>Comparativo Android x iOS</b>	<b>9</b>
5.1	Ambientação . . . . .	10
5.1.1	Linguagens . . . . .	10
5.1.2	IDEs . . . . .	10
5.2	Arquitetura . . . . .	10
5.2.1	Arquivos gerados . . . . .	10
5.3	Controles básicos . . . . .	10
5.3.1	Mostrando textos . . . . .	10
5.3.2	Inserindo textos . . . . .	10
5.3.3	Capturando eventos de botões . . . . .	10
5.4	Criando listas . . . . .	10
5.4.1	Listando arrays . . . . .	10
5.4.2	Personalizando linhas . . . . .	10
5.4.3	Capturando eventos de seleção . . . . .	10
5.5	Acesso a dados . . . . .	10
5.5.1	SQLite . . . . .	10
5.5.2	Preferências . . . . .	10
5.6	Parser JSON . . . . .	10
5.7	Conexão HTTP . . . . .	10
<b>6</b>	<b>Resultados</b>	<b>11</b>
<b>7</b>	<b>Conclusão</b>	<b>12</b>
7.1	Perspectivas Futuras . . . . .	12
	<b>Referências Bibliográficas</b>	<b>17</b>

# Lista de Figuras

3.1	Mapa Mental . . . . .	5
3.2	Diagrama de casos de uso para Criador e Jogador . . . . .	6
3.3	Diagrama de entidades . . . . .	7

# Lista de Tabelas

# Lista de Siglas

# Capítulo 1

## Introdução

### 1.1 Motivação

---

### 1.2 Objetivos

---

Os objetivos gerais e específicos desta monografia, são apresentados a seguir.

#### 1.2.1 Objetivos Gerais

#### 1.2.2 Objetivos Específicos

### 1.3 Organização do Texto

---

# Capítulo 2

## Fundamentação teórica

Neste capítulo, a evolução dos discos rígidos é descrita de maneira breve para contextualizar o leitor quanto às tecnologias utilizadas atualmente e ao seu funcionamento.

### 2.1 Introdução

---

### 2.2 SaaS

---

### 2.3 REST

---

[colocar um bla bla bla inicial, historico, explicando o acronimo, etc]

#### 2.3.1 RESTful

Em uma arquitetura julgada como RESTful, o método desejado é informado dentro do método HTTP, contido no header do mesmo. Além disso, o escopo da informação é colocado na URL, o que, de acordo com 1, torna uma "combinação poderosa". De acordo com 1, por definição, uma aplicação deixa de ser RESTful caso o método HTTP não combine com o método da informação, ou seja, com a funcionalidade esperada para aquela estrutura de dados. [aqui cabe um exemplo de algo não-RESTful, mas que parece ser]

## 2.4 JSON

---

### 2.4.1 Introdução

[aqui cabe acronimo, historico, etc]

### 2.4.2 Definição

De acordo com 1, "é muito mais facil para um browser lidar com uma estrutura javascript oriunda de uma estrutura JSON do que a partir de um documento XML". Ainda de acordo com 1, cada web browser oferece uma interface JavaScript diferente para seus parsers XML, enquanto um objeto JSON, que por definição é um objeto JavaScript, será interpretado da mesma maneira em qualquer interpretador JavaScript. De acordo com 1, o JSON é uma alternativa mais leve para serialização de dados do que o XML, definido pelo XML Schema. [apresentar alguma pesquisa comparativa, vinda de algum artigo, pode ser uma boa].

### 2.4.3 Estrutura

[aqui cabe a descrição de como montar um json]

## 2.5 Rails

---

### 2.5.1 Visão Geral

## 2.6 iOS

---

### 2.6.1 Visão Geral

O iOS é um sistema operacional para dispositivos móveis, lançado pela Apple em 2007. Inicialmente, foi desenvolvido para o iPhone, sendo posteriormente aproveitado nos dispositivos iPod Touch, iPad e Apple TV. Ele é um sistema operacional licenciado para rodar apenas em hardware produzido pela Apple, otimizado para a arquitetura de processadores ARM. Sua entrada de dados é feita de forma direta, através de multi- toques. Esses toques podem ser desde encostar o dedo, similar a um clique do mouse, até balançar o aparelho, de modo a utilizar seu acelerômetro. Todos os controles de entrada de dados são controlados pela GUI Cocoa Touch.

### 2.6.2 Ciclo de vida

O ciclo de vida constitui uma sequência de eventos entre o início e a finalização da aplicação. Um aplicativo iOS começa quando o usuário toca o ícone da mesma na home do dispositivo. Feito isso, o sistema operacional inicia alguns procedimentos de renderização e chama a função principal (`main.m`) do aplicativo. Uma vez iniciado, o comando da execução passa a ser do UIKit, framework de controle do iOS, que carrega a interface gráfica e lê o loop de eventos. Durante o loop, o UIKit delega cada evento a seu respectivo objeto e responde aos comandos emitidos pelo aplicativo. Quando o usuário realiza uma ação que causa um evento de saída, o UIKit notifica a aplicação e inicia o processo de saída.

## 2.7 Android

---

### 2.7.1 Visão Geral

### 2.7.2 Ciclo de vida

## 2.8 Resumo do Capítulo

---



# Capítulo 3

## Aplicação

### 3.1 Visão geral

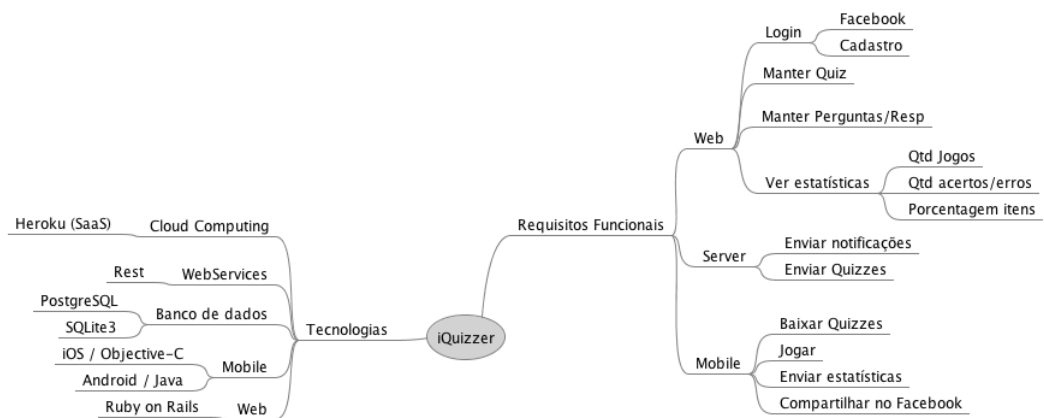


Figura 3.1: Mapa Mental

## 3.2 Requisitos

### 3.2.1 Requisitos funcionais

### 3.2.2 Requisitos não-funcionais

## 3.3 Casos de uso

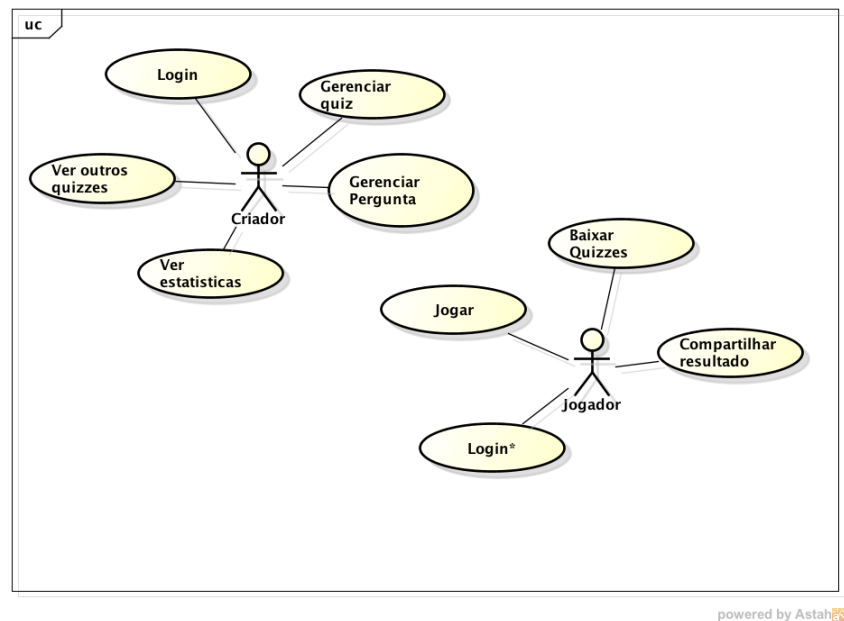


Figura 3.2: Diagrama de casos de uso para Criador e Jogador

## 3.4 Diagrama de classes

## 3.5 Diagrama de entidades

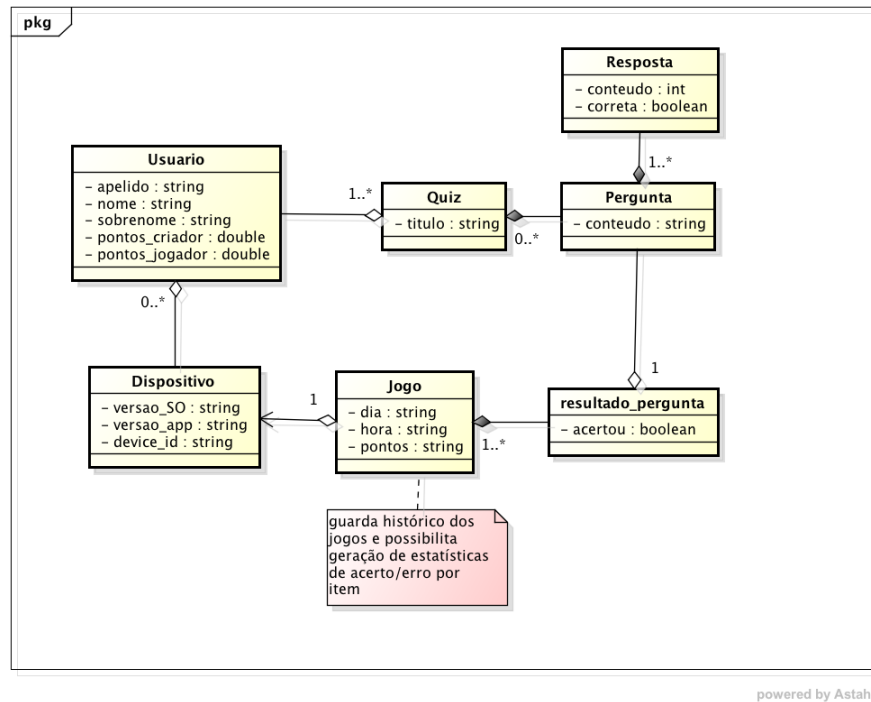


Figura 3.3: Diagrama de entidades

# Capítulo 4

## Implementação web

### 4.1 Criação de aplicação

### 4.2 Rotas

### 4.3 Autenticação

### 4.4 Front-End com bootstrap

### 4.5 Deploy

# Capítulo 5

# Comparativo Android x iOS

## 5.1 Ambientação

---

### 5.1.1 Linguagens

### 5.1.2 IDEs

## 5.2 Arquitetura

---

### 5.2.1 Arquivos gerados

## 5.3 Controles básicos

---

### 5.3.1 Mostrando textos

### 5.3.2 Inserindo textos

### 5.3.3 Capturando eventos de botões

## 5.4 Criando listas

---

### 5.4.1 Listando arrays

### 5.4.2 Personalizando linhas

### 5.4.3 Capturando eventos de seleção

## 5.5 Acesso a dados

---

### 5.5.1 SQLite

### 5.5.2 Preferências

## 5.6 Parser JSON

---

## 5.7 Conexão HTTP

---

# Capítulo 6

## Resultados

# Capítulo 7

## Conclusão

### 7.1 Perspectivas Futuras

---



# Referências Bibliográficas

ADAMS, R. *High performance memory testing : design principles, fault modeling, and self-test*. Boston: Kluwer Academic, 2003. ISBN 1402072554.

ARCHIVES, I. *Magnetic Disk*. 1970. Disponível via Internet, <[http://www-03.ibm.com/ibm/history/exhibits/storage/storage\\_PH4-15.html](http://www-03.ibm.com/ibm/history/exhibits/storage/storage_PH4-15.html)>. Acessado em 15 de Outubro de 2011.

BEZ, R. *et al.* Introduction to flash memory. *Proceedings of the IEEE*, v. 91, n. 4, p. 489 – 502, abril 2003. ISSN 0018-9219.

CARRIER, B. *File system forensic analysis*. Boston, Mass. London: Addison-Wesley, 2005. ISBN 0321268172.

COMMITTEE, I. C. for I. T. S. I. T. T. *AT Attachment 8 - ATA/ATAPI Architecture Model(ATA8-AAM)*. Projeto t13/1700-d. [S.l.], Maio 2006.

COMMITTEE, I. C. for I. T. S. I. T. T. *AT Attachment 8 - ATA/ATAPI Command Set (ATA8-ACS)*. Projeto t13/1699-d. [S.l.], Maio 2007.

COMMITTEE, I. C. for I. T. S. I. T. T. *SCSI / ATA Translation - 2 (SAT-2)*. Projeto t10/1826-d. [S.l.], Junho 2008.

COMMITTEE, I. C. for I. T. S. I. T. T. *SCSI Architecture Model - 5 (SAM-5)*. Projeto t10/2104-d. [S.l.], Setembro 2008.

COMMITTEE, I. C. for I. T. S. I. T. T. *T10 Working Drafts*. 2011. Acessado em 15 de Agosto de 2011. Disponível em: <<http://www.t10.org/drafts.htm>>.

COMPONENTS, I. T. A. E. *Solid State Drives, Separating myths from facts*. Junho 2009. Disponível via Internet, <[http://www.toshiba.com/taec/components/Generic/SSD\\_Myths.pdf](http://www.toshiba.com/taec/components/Generic/SSD_Myths.pdf)>. Acessado em 04 de Novembro de 2011.

CORPORATION, D. A. *Hard Disk Sector Structures*. Dezembro 2001. Disponível via Internet, <[http://www.dewassoc.com/kbase/hard\\_drives/hard\\_disk\\_sector\\_structures.htm](http://www.dewassoc.com/kbase/hard_drives/hard_disk_sector_structures.htm)>. Acessado em 02 de Novembro de 2011.

EIWFELDT, H. *The Linux SCSI programming HOWTO*. Maio 1996. Disponível via Internet, <<http://tldp.org/HOWTO/archived/SCSI-Programming-HOWTO/>>. Acessado em 20 de Maio de 2011.

HARKER, J. M. *et al.* A quarter century of disk file innovation. *IBM Journal of Research and Development*, v. 25, n. 5, p. 677 –690, sep. 1981. ISSN 0018-8646.

HUGHES, G. *et al.* Improved disk-drive failure warnings. *Reliability, IEEE Transactions on*, v. 51, n. 3, p. 350 – 357, sep 2002. ISSN 0018-9529.

INTEL. *What are the advantages of TRIM and how can I use it with my SSD?* Setembro 2010. Disponível via Internet, <[http://www.intel.com/support/ssdc/hpssd/sb/CS-031846.htm?wapkw=\(TRIM\)](http://www.intel.com/support/ssdc/hpssd/sb/CS-031846.htm?wapkw=(TRIM))>. Acessado em 04 de Novembro de 2011.

ISERMANN, R. *Fault-diagnosis applications : model-based condition monitoring : actuators, drives, machinery, plants, sensors, and fault-tolerant systems*. Heidelberg New York: Springer, 2011. ISBN 9783642127663.

KARI, H. *Latent sector faults and reliability of disk arrays*. Helsinki University of Technology, 1997. ISBN 9789512235483. Disponível em: <<http://books.google.com/books?id=pspEAAAACAAJ>>.

KINNEY, M. H. A. *Generic SCSI Command Generator*. Dissertação — Worcester Polytechnic Institute, Março 2004. Disponível em: <<http://web.cs.wpi.edu/~claypool/mqp/gd-scsi/final.pdf>>.

KUNETT, V. *et al.* An in-system reprogrammable 256k cmos flash memory. In: *Solid-State Circuits Conference, 1988. Digest of Technical Papers. ISSCC. 1988 IEEE International*. [S.l.: s.n.], 1988. p. 132.

LEE, P.; ANDERSON, T. *Fault tolerance, principles and practice*. Wien New York: Springer-Verlag, 1990. ISBN 0387820779.

LOUGHE, P. *An Overview of the SquashFS filesystem*. 2008. Disponível via Internet, <<http://tree.celinuxforum.org/CelfPubWiki/ELCEurope2008Presentations?action=AttachFile&do=get&target=squashfs-elce.pdf>>. Acessado em Novembro de 2011.

MAMUN, A. *Hard disk drive : mechatronics and control*. Boca Raton, FL: CRC Press, 2007. ISBN 9780849372537.

MASON, H. Scsi, the industry workhorse, is still working hard. *Computer*, v. 33, n. 12, p. 152 – 153, dec 2000. ISSN 0018-9162.

MASUOKA, F. *et al.* A new flash e2prom cell using triple polysilicon technology. In: *Electron Devices Meeting, 1984 International*. [S.l.: s.n.], 1984. v. 30, p. 464 – 467.

MCLEAN, P. *Proposal for a Selective Self-test*. Outubro 2001. Disponível via Internet, <<http://www.t10.org/t13/technical/e01139r0.pdf>>. Acessado em 15 de setembro de 2011.

MORIMOTO, C. *Hardware - O Guia Definitivo*. SULINA, 2007. ISBN 9788599593103. Disponível em: <<http://books.google.com/books?id=P0ZDbwAACAAJ>>.

MORIMOTO, C. *Hardware, V.2 - O Guia Definitivo*. SULINA, 2010. ISBN 9788599593165. Disponível em: <<http://books.google.com/books?id=vPpBbwAACAAJ>>.

MORIMOTO, C. E. *Ciclos de Gravação e a questão da logenvidade*. Outubro 2010. Disponível via Internet, <<http://www.hardware.com.br/tutoriais/entendendo-ssd/ciclos-gravacao-longevidade.html>>. Acessado em 20 de Outubro de 2011.

PARRISH, K. *Seagate Launching "Industry's First" 4TB HDD*. Setembro 2011. Disponível via Internet, <<http://www.tomshardware.com/news/GoFlex-Desk-4TB-3.5-inch-industrial-design-USB-3.0,13371.html>>. Acessado em 02 de Novembro de 2011.

PC-DOCTOR. *Case Study, PC-Doctor Hard Drive Testing*. 2011. Disponível via Internet, <[http://www.pc-doctor.com/files/english/case\\_study\\_hdd\\_testing.pdf](http://www.pc-doctor.com/files/english/case_study_hdd_testing.pdf)>. Acessado em Julho de 2011.

POTTS, P. R. *SixHardDriveFormFactors.jpg*. Março 2008. Disponível via Internet, <<http://en.wikipedia.org/wiki/File:SixHardDriveFormFactors.jpg>>. Six hard disk drives with cases opened showing platters and heads; 8, 5.25, 3.5, 2.5, 1.8 and 1 inch disk diameters are represented.

RENT, T. M. *Origins of Solid State Drives*. Março 2010. Disponível via Internet, <<http://www.storagereview.com/origin-solid-state-drives>>. Acessado em 12 de Outubro de 2011.

RENT, T. M. *SSD Architecture*. Abril 2010. Disponível via Internet, <[http://www.storagereview.com/ssd\\_architecture](http://www.storagereview.com/ssd_architecture)>. Acessado em 20 de Agosto de 2011.

RENT, T. M. *SSD Controller*. Abril 2010. Disponível via Internet, <[http://www.storagereview.com/ssd\\_controller](http://www.storagereview.com/ssd_controller)>. Acessado em 12 de Outubro de 2011.

RICHARDS, M. *Card Capacitor Read Only Store (CCROS) for microprogram*. 1965. Disponível via Internet, <<http://www.computerhistory.org/revolution/memory-storage/8/264/1103>>. Acessado em 22 de Outubro de 2011.

SEAGATE. *The Case for Solid-State Hybrid Drives*. 2011. White Paper. Disponível via Internet, <[http://www.seagate.com/docs/pdf/corporate/seagate\\_view\\_solid\\_state\\_hybrid\\_drive\\_gen.pdf](http://www.seagate.com/docs/pdf/corporate/seagate_view_solid_state_hybrid_drive_gen.pdf)>. Acessado em 14 de Setembro de 2011.

SHIMPI, A. L. *OCZ Z-Drive R4 CM88 (1.6TB PCIe SSD) Review*. Setembro 2011. Disponível via Internet, <<http://www.anandtech.com/show/4879/ocz-zdrive-r4-cm88-16tb-pcie-ssd-review>>. Acessado em 02 de novembro de 2011.

STEVENS, C. E. *ATA Command Pass-Through*. 04-262r8a. ed. [S.l.], Março 2005.

STEVENS, L. D. The evolution of magnetic storage. *IBM Journal of Research and Development*, v. 25, n. 5, p. 663 –676, sep. 1981. ISSN 0018-8646.

- STROM, B. *et al.* Hard disk drive reliability modeling and failure prediction. *Magnetics, IEEE Transactions on*, v. 43, n. 9, p. 3676 –3684, sept. 2007. ISSN 0018-9464.
- SYSADM, H. *Solid State Drives, some theory and a selection of videos*. Julho 2011. Disponível via Internet, <<http://www.happysysadm.com/search?q=ssd>>. Acessado em 20 de Agosto de 2011.
- TANENBAUM, A. *Modern operating systems*. Upper Saddle River, N.J: Prentice Hall, 2001. ISBN 0130313580.
- TAO, M. Y. *Tour the Linux generic SCSI driver*. Fevereiro 2009. Disponível via Internet, <<http://www.ibm.com/developerworks/linux/library/l-scsi-api/index.html>>. Acessado em 8 de maio de 2011.
- TOOLBOX, L. S. *SCSI TOOLBOX, LLC - Using SAT to Access SATA drives*. Agosto 2008. Disponível via Internet, <<http://www.scsitoolbox.com/pdfs/IssuingATACommands.pdf>>. Acessado em 28 de Julho de 2011.
- TROJANOWSKI, B. *ATA messages via SCSI Layer*. Janeiro 2010. Disponível via Internet, <<http://www.jukie.net/bart/blog/ata-via-scsi>>. Acessado em 02 de Novembro de 2011.
- VASCONCELOS, L. *Hardware Total*. [S.l.]: MAKRON BOOKS, 2002.
- WESZ, R. *Falha, erro ou defeito?* Junho 2007. Disponível via Internet, <<http://testsnews.wordpress.com/2007/06/24/falha-erro-ou-defeito/>>. Acessado em 20 de Outubro de 2011.
- WIKI, U. *BootToRam*. Julho 2011. Disponível via Internet, <<https://wiki.ubuntu.com/BootToRAM>>. Acessado em Julho de 2011.
- ZHAO, X. *et al.* A novel method to restrain the radial error propagation in self-servowriting in hard disk. In: *Automation and Logistics, 2007 IEEE International Conference on*. [S.l.: s.n.], 2007. p. 2736 –2739.
- ZIELSKI, S.; SOSNOWSKI, J. The scsi interface conformance tests generation. In: *Dependability of Computer Systems, 2007. DepCoS-RELCOMEX '07. 2nd International Conference on*. [S.l.: s.n.], 2007. p. 360 –367.