Universidade Federal do Piauí
Campus Senador Helvídio Nunes de Barros
Curso Bacharelado em Sistemas de Informação

Mirielly Alves Marinho Sobral

Estudo e Implementação de Autenticação de Documentos Digitais para *sites* adaptáveis com *Design* Responsivo

Picos
2014

Mirielly Alves Marinho Sobral

Estudo e Implementação de Autenticação de Documentos Digitais para *sites* adaptáveis com *Design* Responsivo

Trabalho de Conclusão de Curso apresentado ao Curso de Sistemas de Informação Campus Senador Helvídio Nunes de Barros da Universidade Universidade Federal do Piauí como parte dos requisitos para obtenção do Grau de Bacharelado, sob orientação Professora Mestre Juliana Oliveira de Carvalho.

Picos

2014

Eu, **Mirielly Alves Marinho Sobral**, abaixo identificado(a) como autor(a), autorizo a biblioteca da Universidade Federal do Piauí a divulgar, gratuitamente, sem ressarcimento de direitos autorais, o texto integral da publicação abaixo discriminada, de minha autoria, em seu site, em formato PDF, para fins de leitura e/ou impressão, a partir da data de hoje.

Picos-PI 12 de março de 2014.

Assinatura

FICHA CATALOGRÁFICA
Serviço de Processamento Técnico da Universidade Federal do Piauí
Biblioteca José Albano de Macêdo

S677e Sobral, Mirielly Alves Marinho.
Estudo e implementação de autenticação de documentos digitais para sites adaptáveis com desgn responsivo / Mirielly Alves Marinho Sobral. – 2013.
CD-ROM : il. ; 4 ¾ pol. (82 p.)

Monografia(Bacharelado em Sistemas de Informação) – Universidade Federal do Piauí. Picos-PI, 2013.
Orientador(A): Profa. MSc. Juliana Oliveira de Carvalho

1. Atenticação de Documentos Digitais. 2. Ruby on Rails. 3. Usabilidade. 4. Dispositivos Móveis I. Título.

CDD 005.7

Mirielly Alves Marinho Sobral

Estudo e Implementação de Autenticação de Documentos Digitais para *sites* adaptáveis com *Design* Responsivo

Trabalho de Conclusão de Curso apresentado ao Curso de Sistemas de Informação Campus Senador Helvídio Nunes de Barros da Universidade Universidade Federal do Piauí como parte dos requisitos para obtenção do Grau de Bacharelado, sob orientação Professora Mestre Juliana Oliveira de Carvalho.

Data de Aprovação:

10/03/2014

Juliana Oliveira de Carvalho
UFPI

Patricia Medyna Lauritzen de Lucena Drumond
UFPI

Leonardo Pereira de Sousa
UFPI

Picos

2014

Dedico esse trabalho principalmente a Mainha, que apesar de não esta fisicamente aqui, sinto sua presença viva dentro do meu coração, e toda essa força, coragem e determinação que carrego dentro de mim aprendi com ela. Também dedico a Papai, pois sem ele não seria possível chegar até aqui. Ele é um exemplo de superação. Amo muito vocês, obrigada por tudo que me ensinaram. Dedico a minha titia Dorinha, minha irmã Netinha, minhas sobrinhas Evinny, Keke, Bianca e Beatriz, ao meu amor e aos meus verdadeiros amigos, por estarem sempre presentes na minha vida, ajudando nessa caminhada, seja com palavras, com abraços, com amor e carinho.

Agradecimentos

Bom, o que dizer nesse momento tão feliz. Simplesmente até agora escrevendo esses agradecimentos não estou acreditando que consegui terminar esse trabalho. Foi muito difícil,mas com muita força e determinação eu consegui. No entanto, isso não seria possível sem as pessoas que contribuíram para essa conquista, que encerra mais uma etapa na minha vida. Em especial agradeço... A Deus por toda força que me deu para realização desse trabalho. Por sempre encher meu coração de fé, e me fazer acreditar que tudo daria certo, mesmo diante de todos os problemas. "Para o homem é impossível, mas para Deus todas as coisas são possíveis."Mateus 19:26.

A minha família que sempre me apoiou em todos os momentos. Em especial a minha mainha, que infelizmente não pode estar presenciando fisicamente essa minha conquista, mas tenho certeza que está presente comigo em todos os momentos. Mainha te amo tanto, saudades e o que me resumo, obrigada por tudo. Ao meu pai por nunca desistir mesmo diante de tudo que enfrentamos e por sempre perseverar com força e fé, o senhor é um exemplo. Obrigado por nunca me deixar faltar nada. A minha titia Dorinha, que é minha segunda mãe. Obrigada pelo amor, cuidado, carinho, e por ser essa pessoa tão linda. Nem sei o que seria de mim aqui nesse Piauizão sem a senhora. A minha Irmãzona Netinha, a mais chata legal do mundo. Obrigada pelo amor dedicado, por sempre torcer pela minha vitória, me dando broncas quando necessário e por nunca desistir de mim. As minhas sobrinhas lindas (minhas leõesinhos), pois, o sorriso de cada uma delas me deu forças nos momentos mais tristes. Sem eles não seria a mesma coisa. Os dias de sol não seriam tão brilhantes e lindos sem vocês. Vocês são indispensáveis nessa caminhada chamada de vida. A uma pessoa muito importante Roselle Freitas, que me aguentou até quando eu mesmo não me aguentava (rsrsrs). Mesmo diante da distância você se fez presente todos os dias nessa caminhada. Obrigada pelo carinho, pela atenção, pelos sorrisos, por me dar seu ombro nos momentos de choro e por ser essa pessoa maravilhosa, prestativa que sempre me encorajou a seguir em frente independente dos obstáculos. Existem pessoas na vida que sempre queremos por perto, você é uma delas. Te amo!

A minha orientadora, Juliana Oliveira de Carvalho, aquela que nunca vi andar sem salto na Federal, ao não ser nos dias que arrumava as unhas, que enche um quadro de assuntos em menos de 1 minuto, que reprovei minha única disciplina, que nunca solta antes de 12:00 em ponto, que

anda sempre toda elegante e que tem o super poder de ser esposa, mãe, estudante, orientadora. Fico impressionada, como ela dar conta?. Enfim, com ela aprendi muito durante todo esse tempo e a quem tenho como exemplo a seguir. Obrigada pela disposição em dividir seu tempo e conhecimento, pelos conselhos, pelas ideias que sempre são boas, pela força quando o que eu mais queria era desistir, e acima de tudo obrigada pela sua amizade, até boliche joguei com a senhora (esse dia foi massa). Enfim, esse trabalho não seria possível sem a senhora.

Aos meus professores em especial a minha banca Prof[a] Patricia Medyna Lauritzen de Lucena Drumond (eita nome grande rsrs) que aceitou com muito carinho participar dessa banca, e que sempre foi muito simpática e dócil com todos, nos ouvindo e tentando resolver as coisas da melhor maneira possível e que esta revolucionando nosso curso, trazendo grandes melhorias para o mesmo. E ao Prof[o] Leornado Pereira de Sousa que aceitou com muito carinho participar desse trabalho contribuindo com melhorias para o mesmo.

Ao meu amigo irmão Pablo Moreira (Vulgo Pablito ou bochechão), que me ajudou na realização deste trabalho, que dedicou seu tempo me ensinando e me encorajando a não desistir e o fez com muito carinho e atenção. Sabe, nossa amizade era para ser, pois eu morando no Pará, próximo dele, vim o conhecer no Piauí. É por que era para ser mesmo, já estava escrito que seriamos bests nessa vida. Obrigada por tudo meu amigão você é uma das pessoas que levarei pra vida toda. Ainda vamos comemorar essa conquista. Me aguarde! Te adoro muito.

A minha mais chata legal amiga, Fátima dos Santos. Ela teve a maior prova de resistência da vida dela, que foi me aguentar durante 4 anos (rsrs), e ela passou. Obrigada pelas conversas nas madrugadas sem sono, pelos gestos de amizade, pelas chapinhas, por me compreender, por me ouvir, por me chatear, por tudo. Creio que tudo isso foi necessário para firmar nossa amizade na rocha e torná-la inabalável. Você é muito especial e te levarei pra toda vida. Te amo minha amiga. A gangue (do bem tá gente!), Andrei Maxwell, Bruno Fonseca e Matheus Duarte o novo integrante. Obrigado pela amizade de vocês, pela força, pelas gargalhadas, pelos Brunos'fests e por tudo nesses anos de amizade. Passamos por tantas coisas engraçadas, tantos apertos, as quais nunca esquecerei. Vocês são os melhores, meus irmãozinhos, adoro vocês, me empresta um real? (rsrsrsrs).

Aos demais amigos em especial: Karine Nascimento (Essa também me aguentou demais, passamos por bons e ruins bocados, obrigada Karinete), Aislan Maia (esse eu enchi tanto o saco, obrigada cara), Ana Verônica Ferreira Carvalho (Dispostos no notebook e smartphone, RESPECTIVAMENTE. Ela vai entender. Obrigada!!!), Woshington Valdeci (Vulgo Oxito, meu gigante favorito), Janaina Moura (Naina, amiga lindíssima que sempre torceu por mim), Gabrielly Matos (Best, obrigada por sempre me incentivar), obrigada por todo o incentivo e contribuição no desenvolvimento desse trabalho, pela cumplicidade e amizade oferecidas em todos os mo-

mentos a mim. Obrigada a todos que, mesmo não tendo seus nomes citados aqui, contribuíram direta e indiretamente para conclusão dessa etapa. Obrigado meu povo.

“Tudo posso naquele que me fortalece.”

Filipenses 4:13

“O tempo é limitado, então não gastes seu tempo vivendo a vida de outro. Não fiques preso no dogma que é viver como os outros pensam que deverias viver. Não deixe que as opiniões dos outros calem sua voz interior. E o mais importante, tenha coragem para fazer aquilo que manda seu coração e intuição”

Steve Jobs

“A fé é a certeza das coisas que se esperam e a convicção dos fatos que não se vêem.”

Hebreus 11:1

Resumo

A Internet tem proporcionado uma grande mudança de hábitos, tanto das pessoas, como dos órgãos públicos e empresas. E uma dessas mudanças é a substituição dos documentos em papel pelos em meios eletrônicos. Essa prática proporciona benefícios em termos de agilidade, facilidade de acesso, desburocratização dos processos, preservação do meio ambiente, baixo custo, entre outros. Mas, além disso, faz-se necessário garantir a autenticidade e integridade desses documentos. Para isso se utilizam tecnologias de criptografia para torná-los seguros e confiáveis. O objetivo desse trabalho é apresentar o desenvolvimento de uma aplicação Web para o curso de Sistemas de Informação da Universidade Federal do Piauí, Campus Senador Helvídio Nunes de Barros, em Picos, no Estado do Piauí, que disponibilize e autentique os planos de curso das disciplinas que compõem o fluxograma do mesmo. A aplicação segue alguns critérios de usabilidade estabelecidos por Nielsen, e apresenta uma interface responsiva que se adapta aos diferentes tamanhos de tela dos dispositivos encontrados no mercado. O desenvolvimento do trabalho deve ser obtido através da utilização de tecnologias como: Framework Ruby on Rails, a linguagem de programação Ruby, Framework de front-end Bootstrap e técnicas de criptografia. Por fim, foram feitos testes com a aplicação, relacionados à autenticação dos planos de curso e a adaptabilidade em diferentes dispositivos, os mesmos são mostrados através de estudos de caso fictícios a fim de mostrar todas as possibilidades de resultados.

Palavras-chave: Autenticação de Documentos Digitais, Ruby On Rails, Usabilidade, Dispositivos Móveis, Responsividade.

Abstract

The internet has brought to us a great change of habits, not only for the people, but also for the public agencies and companies. And one of these changes is the replacement of paper documents for eletronic ones. This practice provides benefits in terms of agility, ease of access, less bureaucratic processes, environment preservation, low cost and more. But beyond that it's necessary to guarantee the authenticity and integrity of these documents. Encryption technologies are used to make them safe and trustworthy. The goal of this project is to develop a Web application for the course of Information Systems in the Federal University of Piaui (UFPI), Campus Senador Helvídio Nunes de Barros, in Picos, Piauí - Brazil that makes the course plans of the subjects that constitute the flowchart available and also authenticate them, and beyond that, this application will follow some usability criteria stablished by Nielsen and will present a responsive interface that adapts to the different screen sizes of the devices available in the market. The development of this project will be acquired through the utilization of technologies such as: Framework Ruby on Rails, Ruby's programming language, front-end Bootstrap Framework and encryption techniques. At last, tests related to the authencity of the course plans and to the adaptability indifferent devices were done on the application, the same are shown through a fictional study case in order to show all the results possibilities.

Keywords: Digital Documents Authentication, Ruby on Rails, Usability, Mobile Devices, Responsiveness

Lista de Figuras

Lista de abreviaturas e siglas

ACs	Autoridades Certificadoras
AR's	Autoridades de Registro
CP-BRASIL	Infraestrutura de Chaves Pública Brasileira
CSS	Cascading Style Sheets
DES	Data Encryption Standard
DETRAN	Departamento de Trânsito
DIPJ	Declaração do Imposto de Renda Pessoa Jurídica
DRY	Don't repeat yourself
e-CNPJ	Eletrônico- Cadastro Nacional da Pessoa Jurídica
e-CPF	Eletrônico -Cadastro de Pessoas Físicas
IDEA	International Data Encryption Algorithm
IHC	Interação Humano Computador
IRB	Interactive Ruby Shell
ITI	Instituto Nacional de Tecnologia da Informação
MD4	Message Digest Algoritmo 4
MD5	Message-Digest algorithm 5
MVC	Model-view-controller
NIST	National Institute of Standards and Technology
PGP	Pretty Good Privacy
RFC	Request for Comments
RIPE	Race Integrity Primitives Evaluation
RSA	Rivest,Shamir e Adleman
SHA-1	Secure Hash Algorithm
SIGAA	Sistema Integrado de Gestao de Atividades Academicas
SPED	Sistema Público de Escrituração Digital
SSL	Secure Sockets Layer
STJ	Superior Tribunal de Justiça
VPN	Redes privadas virtuais

Sumário

1 Introdução

A utilização de meios eletrônicos está inserido no cotidiano das pessoas. Com o advento da *Internet* o envio de documentos entre cidadãos, governos e empresas intensificaram-se. Essas práticas agilizam os processos e facilitam o acesso a informações, pois independente da localização, todos poderão acessar. Além disso, os documentos eletrônicos são fáceis de gerenciar, ocupam pouco espaço físico, e podem ser transmitido de forma rápida pela *Internet*.

Com a intensificação do envio de documentos por meio eletrônico, surgiu à necessidade de agregar confiança e segurança aos documentos digitais, e isso só foi possível através do uso da criptografia que possibilita a autenticação, a integridade e a tempestividade dos mesmos, tornando-os seguros. Com isso é possível criar um ambiente seguro que facilita a adesão das empresas a esse novo tipo de documentação. Esses três fatores acima citados: autenticação, integridade e tempestividade são de suma importância nesse contexto. É através deles que se pode saber quem é o autor do documento, se o documento foi alterado, e se ele foi apresentado dentro de um prazo legal, respectivamente.

As empresas têm migrado para o modelo de formato eletrônico e já estão dispostas a desativar o uso de papeis. Essa tendência é muito forte, e os documentos eletrônicos vieram para ficar (JIMENE, 2013). Essa reestruturação que está acontecendo, além de ser uma solução racional, preserva o meio ambiente, e substitui processos burocráticos e passíveis de fraudes. Segundo Giovanni de Melo (VIOTTI, 2011) alguns órgãos públicos tais como: DETRAN (Departamento de Tânsito), STJ (Superior Tribunal de Justiça), Ministério da Justiça, entre outras, já obtiveram sucesso na implantação de sistemas eletrônicos que substituem os documentos de papel pelos digitais.

Em relação a unidades escolares, sejam de ensino fundamental e médio, instituições superiores e cursos técnicos de qualquer nível, o plano de curso é de suma importância, já que, é a primeira impressão que se tem de uma disciplina. Além disso, é através do mesmo que os alunos tomam conhecimento dos objetivos da disciplina, o conteúdo programático, a metodologia de ensino, o sistema de avaliação empregado pelo professor e a relação da bibliografia utilizada. Portanto, este trabalho justifica-se devido à ausência desse recurso no SIGAA (Sistema de Gestão de Atividades Acadêmicas), e a burocratização encontrada no momento de assinar tais documentos, tornando o processo demorado. Com isso, a automatização deste serviço proporcionará benefícios em termos de agilidade, facilidade de acesso, confiabilidade e segurança desses planos de curso para os alunos. Além do que, os funcionários da coordenação do curso

ganharão tempo desenvolvendo outras atividades.

Interface é um fator importantíssimo na busca da qualidade dos *softwares*. Através da *interface* um *software* ou *site* pode ser rejeitado ou não, pois, o usuário qualifica-o de acordo com sua parte visível e com a qual ele interage. Por isso, torna-se importante utilizar técnicas de usabilidade, que pode ser definida como um estudo ou técnica que proporcionam a facilidade de uso de um dado objeto. Ela busca assegurar que qualquer pessoa use um objeto e que este funcione da forma esperada pela pessoa. Em resumo, usabilidade tem como objetivo a: facilidade de uso, facilidade de aprendizado, facilidade de memorização de tarefas, produtividade na execução de tarefas, prevenção, visando a redução de erros e satisfação do individuo (GOVERNOELETRÔNICO, 2010).

Antes do uso do *Design* Responsivo, era comum para os desenvolvedores criarem dois *sites* separados, um para *mobile* e um para *desktop*. Devido a isso tornava-se o processo de desenvolvimento demorado. Hoje, faz-se apenas um *site* que vai se adaptar muito bem a qualquer tela em que ele for carregado. Além disso, ainda tem o problema dos diferentes tamanhos de tela dos dispositivos. Seria enlouquecedor desenhar múltiplas versões de um mesmo *site* que suprissem cada uma dessas variações de tamanho de tela e cada uma das resoluções de tela disponíveis no mercado. O *Responsive Web Design* é uma das soluções técnicas para esses problemas, através dele, cria-se um *site* de forma que os elementos que o compõem se adaptem automaticamente à largura de tela do dispositivo no qual ele está sendo visualizado, além de que, é necessário apenas um *site* , pois o que irá se adaptar é o *design* (TEIXEIRA, 2011).

Com isso, este trabalho visa o desenvolvimento de uma aplicação *Web* para o curso de Sistemas de Informação voltada para disponibilização de documentos digitais, como os planos de curso, focando na autenticação dos mesmos. Essa aplicação segue alguns critérios de usabilidade estabelecidos por Nielsen (OLIVEIRA, 2013), com uma *interface* responsiva que se adapta aos diferentes tamanhos de tela dos dispositivos encontrados no mercado.

O trabalho está organizado da seguinte maneira:

- Capítulo 2 – Autenticação de documentos digitais: neste capítulo são apresentados o conceito e a importância da autenticação de documentos digitais, além de abordar sobre a regulamentação do uso desses documentos e as técnicas mais utilizadas para autenticação.

- Capítulo 3 - Desenvolvimento e resultados: esse capítulo descreve o modelo proposto neste trabalho, além de detalhar as tecnologias utilizadas, a implementação do Serviço de Autenticação de Documentos Digitais e do *site* Adaptável com *Design* Responsivo, e mostrar os resultados obtidos.

- Capítulo 4 – Conclusão: neste capítulo são apresentadas as conclusões finais alcançadas e possíveis trabalhos futuros.

2 Autenticação de Documentos Digitais

A produção de documentos eletrônicos cresce em ritmo acelerado, hoje os documentos impresso já não mantêm o monopólio na produção documentária. Um exemplo para isso são os jornais, as revistas, pois todos esses publicam seus conteúdos na *Internet*. Para as empresas, ter os documentos em formato eletrônico é um benefício, devido à diminuição dos custos, tanto com papel, quanto com a armazenagem com maior segurança e privacidade das informações constantes nesses documentos. Portanto, a importância dos documentos eletrônicos se da na facilidade do acesso, na diminuição de custos e na diminuição do tempo necessário para se encontrar a informação.

Este capítulo aborda o conceito e a importância da Autenticação de Documentos Digitais, retrata sobre a regulamentação do uso desses documentos e mostra as técnicas mais utilizadas para autenticação.

2.1 Documentos Digitais

Diante da evolução da sociedade e da revolução da informação por meio da *Internet* o conceito de documento vem sofrendo mudanças. Por isso, pode-se entender como documento qualquer meio capaz de representar um significado compreensível, independente do meio, podendo ser físico ou digital (GANDINI et al., 2002).

O documento digital pode ser definido como eletrônico ou informático. Porém, ambos abrangem um mesmo sentido, visto que são produzidos através do uso do computador. Conceituá-lo não é uma tarefa fácil, uma vez que envolve dados ligados diretamente à informática e à tecnologia, que evoluem a todo dia. Assim, pode ser conceituado como aquele que se encontra memorizado em forma digital, sendo percebido pelo homem somente com o auxílio de um programa de computador. Desta forma não é nada mais do que uma sequência de *bits* que, traduzida, nos representará um fato (GANDINI et al., 2002).

Webb (WEBB, 2000) também conceitua o documento eletrônico ou documento digital como sendo todo registro gerado ou recebido por uma entidade pública ou privada, no desempenho de suas atividades, armazenado e disponibilizado ou não, através de sistemas de computação.

Além disso, costuma-se atribuir aos documentos eletrônicos as seguintes características: volatilidade, alterabilidade e fácil falsificação. Os documentos digitais, mesmo com todas estas implicações, podem ter validade jurídica, desde que preencham determinados requisitos,

que são os mesmos exigidos para os documentos tradicionais; contudo, os digitais continuarão diferenciando-se dos tradicionais pela forma prática de seu suprimento e verificação. Os requisitos acima mencionados são a integridade, a autenticidade e a tempestividade (GANDINI et al., 2002) .

Entende-se que a autenticidade de um documento eletrônico é a garantia da identificação e a associação do autor ao conteúdo. Já a integridade é a possibilidade de verificar a qualquer momento se o conteúdo assinado esta íntegro, ou seja, que não sofreu adulterações. Por fim, a tempestividade, que é a possibilidade de verificar se determinado documento foi ou não produzido em determinada ocasião. Em relação aos documentos tradicionais, as formas de autenticação baseiam-se em características materializadas no suporte (não no próprio conteúdo, como é o caso dos documentos eletrônicos), tais como a aposição de assinaturas e marcas, o uso de produtos de segurança (papéis e tintas especiais, por exemplo), a feitura de perícias grafológicas e técnicas, etc. No caso dos documentos eletrônicos, a autenticação será efetuada somente com base no conteúdo (desprezando-se o suporte). Apesar dessa diferença de meios, a autenticação, em ambos os casos, busca atingir os mesmos resultados, ou seja, aferir o cumprimento de requisitos essenciais para a obtenção da eficácia probatória (TADAMO, 2002).

Além do mais, os recursos eletrônicos suprem as reais limitações verificadas com o uso da documentação tradicional, que geralmente é feita através do papel, tornando o documento mais seguro, confiável e sua transmissão se torna rápida e eficiente. Além disso, o trabalho com esse tipo de documento também se torna mais fácil, se comparado com o que se utiliza de papel, visto que é simples a reestruturação de seu conteúdo.

Alguns podem não compreender o uso do documento eletrônico, pois estão acostumados com o clássico, onde é utilizado o papel. Porém, a diferença básica entre o documento tradicional e o documento eletrônico é simplesmente sua forma de materialização. Não devemos permanecer estáticos frente às inovações da tecnologia, pois o mercado avança em direção ao futuro, visando à comodidade e facilidade das pessoas(GANDINI et al., 2002).

Portanto, para a garantia da segurança dos documentos digitais podem ser utilizadas algumas técnicas tais como: criptografia, algoritmos de resumo, assinatura digital, certificação digital etc. O uso dessas técnicas torna o uso dos documentos digitais mais ágeis e menos burocráticos e com total garantia de autenticidade.

2.2 Técnicas que visam garantir a segurança do Documento Digital

2.2.1 Criptografia

A palavra criptografia deriva dos termos gregos Kryptós, que quer dizer oculto, e graph, escrever. A criptografia é uma informação distorcida, que busca ocultar ou dificultar a leitura de uma mensagem pelos interceptadores, fazendo com que somente o destinatário correto consiga decifrar a mensagem. Em dicionários de língua portuguesa, pode ser encontrada a seguinte definição para a palavra criptografia: escrita secreta por meio de abreviaturas ou de sinais convencionados de modo a preservar a confidencialidade da informação (PINHEIRO; NETO, 2012).

Segundo Silva e Luiz (SILVA et al., 2011)

> Dentre as diversas tentativas de definir criptografia de maneira precisa, pode-se dizer de um modo simples, que criptografia é a "ciência"de fazer com que o custo de adquirir uma informação de maneira imprópria seja maior do que o custo obtido com a informação.

Ângela Bittencourt Brasil (BRASIL, 2000) deixa claro que a técnica de assinatura feita através da criptografia e da criptoanálise "consiste numa mistura de dados ininteligíveis onde é necessário o uso de duas chaves, a pública e a privada, para que ele possa se tornar legível". Compara a criptografia como sendo semelhante ao segredo de um cofre forte. Esclarece, ainda, que essa assinatura é formada por uma série de letras, números e símbolos e é feita em duas etapas, sendo que na primeira o autor, através de um *software* que contém um algoritmo próprio, realiza uma operação e faz um tipo de resumo dos dados do documento que quer enviar, também chamado de função *hash*. Em um segundo momento, ele utiliza a chave privada, a qual irá encriptar esse resumo e o resultado desse processo, que é a assinatura digital. Em conclusão, aponta a mesma autora que a assinatura eletrônica, diferentemente da assinatura real, se modifica a cada arquivo transformado em documento, fazendo com que seu autor não a repita como faz com as assinaturas apostas nos documentos reais .

Existem essencialmente duas grandes técnicas de criptografia, denominadas simétrica e assimétrica, que são baseadas na troca de pares de chaves. Este tipo de criptografia possibilita recursos necessários que garante a autenticação de documentos e de pessoas.

Criptografia simétrica

A criptografia de chaves secreta esta baseada na cumplicidade de quem envia e de quem recebe a informação, e somente com o código correto trocado entre eles as mensagens podem ser decodificadas. Segundo Silva e Luiz (SILVA et al., 2011):

> Criptografia de chave secreta (também chamada de criptografia simétrica) usa uma chave secreta para criptografar uma mensagem de texto cifrado e a mesma chave para decifrar o texto cifrado em texto pleno.

Geralmente são utilizados para criptografar dados e fluxos de dados, pois são rápidos e possuem um nível de segurança quase perfeita em algumas implementações. Os algoritmos de chave simétrica podem ser divididos em duas categorias: de bloco e de fluxo. Algoritmos de bloco criptografam os dados um bloco de cada vez, enquanto os algoritmos de fluxo criptografam *byte* por *byte*. (GARFINKEL, 1999).

A criptografia de chaves simétricas se dá quando uma mesma chave é utilizada na cifragem e decifragem da informação como mostra a figura 1.

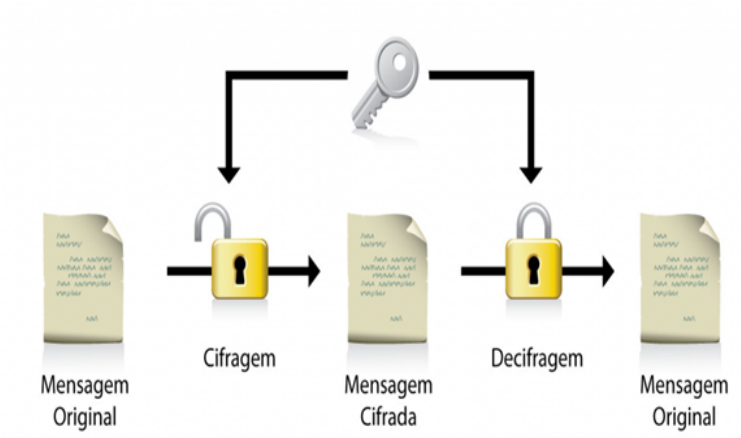


Figura 1 *– Encriptamento e desencriptamento usando chave secreta.*

Várias técnicas de viabilização da criptografia simétrica geram grande confiabilidade deste método, inclusive melhorias contínuas nestas técnicas, fazem que técnicas de criptoanálise se tornem inviáveis num período de tempo. Diferentes algoritmos oferecem diferentes níveis de segurança, este nível depende da dificuldade de quebra do mesmo (SCHNEIER, 1996). Atualmente existem vários algoritmos de chaves simétricas em uso. Os mais comuns no campo de segurança na *Web* são sucintamente descrito a seguir (GARFINKEL, 1999):

- **DES *(Data Encryption Standard)***: é um algoritmo de bloco que usa uma chave de 56 *bits*. Sua segurança de quebra está fragilizada nos últimos tempos vítima da evolução do poder computacional.
- ***Triple*-DES**: Utilizando-se o algoritmo de criptografia DES convencional três vezes com três chaves diferentes obtém-se pelo menos duas vezes mais segurança em relação ao uso singelo do DES.
- ***Blowfish***: Algoritmo de criptografia em bloco, rápido compacto e simples, inventado por Bruce Schneier. Permite uso de chave de tamanho variável até 448 *bits*.

- **IDEA *(International Data Encryption Algorithm)***: Usado pelo programa PGP *(Pretty Good Privacy)* para criptografar arquivos e correio eletrônico usa uma chave de 128 *bits* e é bastante poderoso.
- **RC5**: Desenvolvido por Ronald Rivest, permite que o tamanho da chave, o tamanho dos blocos e o número de vezes que a criptografia será realizada sejam definidos pelo usuário.

A criptografia de chave simétrica garante a confidencialidade da informação, mas não a sua autenticidade ou integridade, visto que a chave utilizada na cifragem deve ser conhecida não só pelo autor da mensagem, mas também por todos os seus leitores. Um leitor de posse da chave poderia substituir o conteúdo da mensagem original, ou ainda publicar outras mensagens em nome do autor (SOBRAL, 2012).

Criptografia assimétrica

A criptografia assimétrica utiliza duas chaves distintas: uma chave privada, conhecida apenas pelo seu criador, e uma chave pública que o criador da chave pode distribuir livremente como mostra a figura 2. Ela pode ser utilizada para garantir a confidencialidade da informação trafegada, para isso, as partes envolvidas trocam as suas chaves públicas e passam a enviar dados cifrados com a chave pública do destinatário. Uma vez cifrada pela chave pública, à mensagem só poderá ser decifrada pela chave privada correspondente, que só é conhecida pelo destinatário (SOBRAL, 2012). Segundo Silva e Luis (SILVA et al., 2011).

> A criptografia de chave pública (também chamada de criptografia assimétrica) envolve duas chaves distintas, uma pública e uma privada. A chave privada é mantida em segredo e nunca deve ser divulgada. Por outro lado, a chave pública não é secreta e pode ser livremente distribuída e compartilhada com qualquer pessoa.

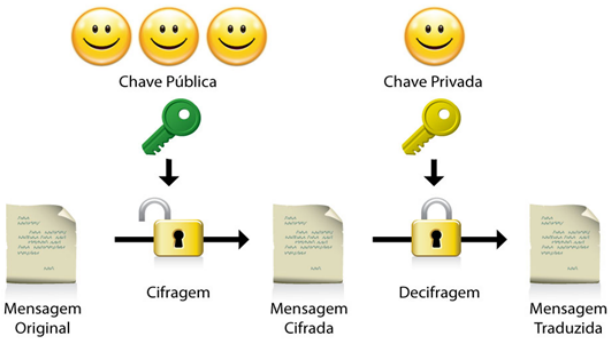


***Figura 2** – Encriptamento e desencriptamento usando chaves públicas e privadas.*

Além disso, a criptografia por chaves assimétricas também podem ser utilizadas para garantir a autenticidade da informação, basta usar as chaves de forma inversa, ou seja, cifra-se a mensagem utilizando a chave privada, com isso ela poderá ser decifrada por qualquer um que possua a chave pública correspondente como mostra a figura 3.

Figura 3 *– Encriptamento e desencriptamento usando chaves de forma inversa.*

2.2.2 Função Hash

Além de garantir autenticidade, também pode-se conferir integridade as informações. Para isso, utiliza-se na computação um conceito de *hash* para gerar uma forma de assinatura única e um arquivo como mostra a figura 4. Uma função *hash* é um programa que utiliza funções matemáticas para transformar um arquivo de qualquer tamanho em uma assinatura única de tamanho fixo. Então, independente do arquivo em questão ser um soneto ou todo o conteúdo da bíblia, a assinatura deles possuirá o mesmo tamanho e será única. Qualquer alteração mínima no conteúdo original pode ser identificada através desse processo (SOBRAL, 2012).

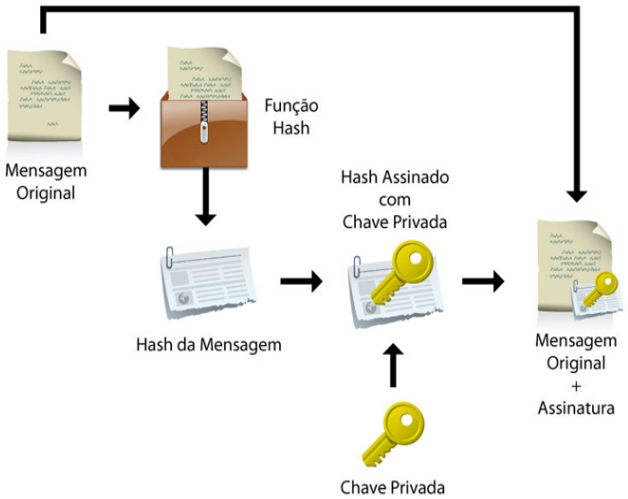


Figura 4 *– Função hash, sendo aplicada para gerar uma forma de assinatura única e um arquivo.*

Por fim, alguns dos algoritmos *hash* mais utilizados são:

- **MD4** ***(Message Digest)***: Criado por Ron Rivest da empresa RSA (RSA *Security*, Inc.). Produz um valor *hash* de 128-*bits*. Efetua uma manipulação de *bits* para obter o valor do *hash*, de forma rápida. É um padrão da *Internet* (RFC-1320). (vários ataques foram detectados, o que fez com que o algoritmo fosse considerado frágil).

- **MD5** ***(Message-Digest algorithm 5)***: É uma extensão do MD4. Produz como saída um valor *hash* de tamanho de 128-*bits*. A obtenção do valor de *hash* é mais lenta, porém é mais seguro. Está definido como um padrão da *Internet*. (RFC-1321). É usado pelo PGP (*Pretty Good Privacy*).

- **SHA-1 (*Secure Hash Algorithm*)**: Desenvolvido pelo NIST (*National Institute of Standards and Technology*), produz um valor *hash* de 160-*bits*. Seu desenvolvimento tem muita relação com o MD5, mas com certas diferenças (Ex: saída 160-*bits*). É considerado mais seguro que o MD4 e MD5 pelo seu tamanho.

- **RIPEMD-160**: É uma função *hash* criptográfica desenhada por Hans Dobbertin, AntoonBosselaers, e Bart Preneel em um projeto chamado RIPE (*Race Integrity Primiti- ves Evaluation*,(1988-1992)). Produz uma saída de 160 *bits*.

2.2.3 Assinatura digital

Nos documentos de arquivo são as assinaturas manuscritas e os carimbos que validam sua autoria, autenticidade e integridade. Já para os documentos eletrônicos essas assinaturas que antes eram feitas a mão, precisavam ganhar também sua forma ou um formato digital, para garantir que os mesmos não fossem violados (ARAUJO; VIEIRA, 2012).

Ao assinar um documento de papel firma- se que o mesmo é íntegro e autêntico. Para Custódio Dias e Rolt (CUSTÓDIO et al., 2009), “o ato de assinar um documento estabelece um vinculo entre quem assina e o documento em si”. Essa ligação acontece tanto na forma manuscrita como na forma digital, porém no caso da assinatura digital essa ligação entre o documento e o autor é feita por um algoritmo de autenticação. Tanto as assinaturas manuscritas quanto as assinaturas digitais estabelecem os mesmos objetivos e finalidades, a de possibilitar ao criador que o documento criado não seja alterado ou violado (ARAUJO; VIEIRA, 2012).

Uma assinatura digital é um algoritmo de autenticação, que possibilita ao criador de um objeto unir ao objeto criado, um código que irá agir como uma assinatura. Esta assinatura confirma que o objeto não foi alterado, desde o ato de sua assinatura e permite identificar o

assinante (MONTEIRO; MIGNONI, 2007).

No processo de assinatura digital o destinatário utiliza uma chave de verificação para averiguar a origem da mensagem recebida e ter certeza de que a mesma não foi alterada enquanto estava sendo enviada. Ainda de acordo com Monteiro e Mignoni (MONTEIRO; MIGNONI, 2007)

> As chaves de assinatura e verificação são distintas, garantindo que o destinatário possa somente verificar a assinatura, mas não será capaz de forjá-la. Devido ao fato de não ser computacionalmente viável forjar uma assinatura sem a posse da chave de assinatura, o autor não pode repudiar o fato que assinou uma mensagem.

A assinatura digital visa garantir que um determinado documento não seja alterado após assinado. A assinatura digital é realizada em duas etapas. Primeiramente o autor, através de um *software* próprio, realiza uma operação e faz um tipo de resumo dos dados do documento que quer enviar, também chamado de "função *hash*". Após essa operação, ele usa a chave privada de seu certificado digital para encriptar este resumo. O resultado deste processo é a assinatura digital como mostra a figura 5. (STSC, 2013)

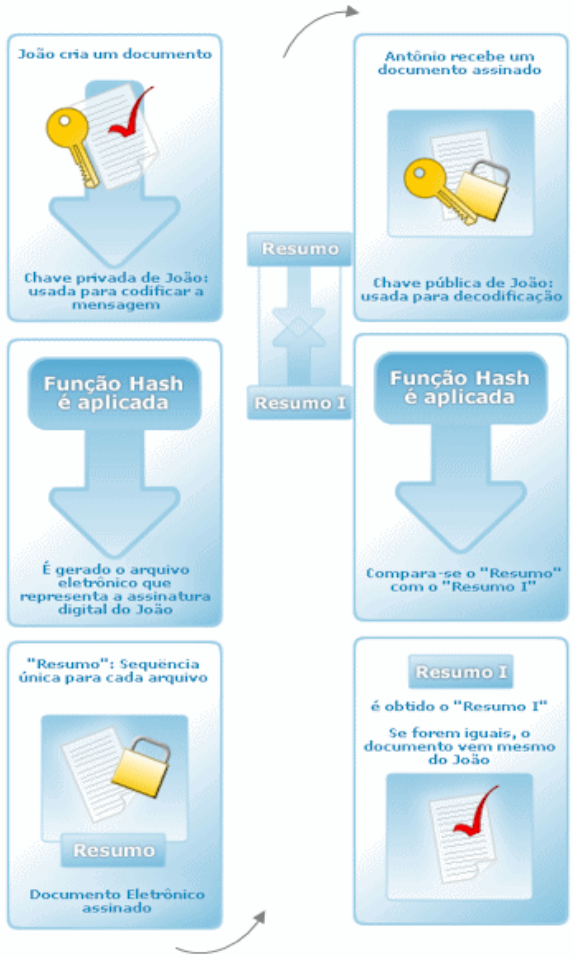


Figura 5 *– Processo de assinatura digital.*

2.2.4 Certificado digital

Os certificados digitais, também chamados de identidade digital, é um arquivo de computador capaz de identificar dados de um indivíduo ou entidade, possuindo chaves para fazer a certificação. Eles são compostos por um par de chaves (Chave Pública e Privada). Segundo Silva e Luiz (SILVA et al., 2011):

> De uma forma genérica, um certificado digital é a versão digital de um documento de identidade. Quando é necessário comprovar sua identidade, o certificado é utilizado como forma de presença, por mostrar a chave privada que se relaciona com uma chave pública.

Qualquer pessoa tem a possibilidade de criar seu próprio certificado digital e usá-lo, sabendo disso foram criadas as Autoridades Certificadoras (ACs). As ACs são entidades ou empresas com alto nível de confiança e reputação, elas visam garantir a segurança da parte que necessita de comprovação de identidade do proprietário do certificado. Elas emitem certificados digitais para outras entidades, empresas e indivíduos, que precisam se identificar e garantir as suas operações no mundo digital. As ACs trabalham junto com uma autoridade de registro, que é uma empresa ou uma entidade responsável pela verificação das informações fornecidas pelos requisitantes dos certificados (ARAUJO; VIEIRA, 2012).

As ACs são responsáveis por emitir, suspender, renovar ou revogar certificados, vinculando pares de chaves criptográficas ao respectivo titular. Essas entidades devem ser supervisionadas e submeter-se à regulamentação e fiscalização de organismos técnicos. No meio físico, para que uma credencial de identificação seja aceita em qualquer estabelecimento, a mesma deverá ser emitida por um órgão habilitado pelo governo. No meio digital ocorre o mesmo - devemos apenas aceitar certificados digitais que foram emitidos por autoridades certificadoras de confiança (CERTISIGN, 2012).

O objetivo dos certificados digitais é confirmar a identidade do usuário na *Web*, no correio eletrônico, transação *on-line*, transação eletrônica, informação eletrônica, cifrar chaves de sessão (utilizadas para cifrar grandes volumes de dados) e assinatura de documento eletrônico, conferindo validade jurídica e garantindo a segurança de suas informações (VOLPI, 2001).

A certificação digital permitiu uma nova interpretação na maneira como as pessoas se interagiam com os negócios e transações *on-line*, pois instituiu validade jurídica nas ações realizadas pelo meio *on-line*, trazendo para os usuários maior segurança, credibilidade, comodidade e agilidade (PINHEIRO; NETO, 2012).

Com validade jurídica vigente na legislação brasileira, a assinatura eletrônica com certificado digital substitui o papel com total garantia de autenticidade da autoria, integridade do conteúdo do documento (se uma vírgula for alterada, sabe-se que houve alteração do seu con-

teúdo) e privacidade (CERTISIGN, 2012).

Segundo Certising (CERTISIGN, 2012), podemos destacar algumas vantagens que a certificação digital trouxe para nossa realidade, como:

- **Os contribuintes podem renegociar dívidas com o governo federal**: Os certificados digitais ou códigos de acesso são necessários para a validação do parcelamento da dívida;
- **Na era do TI Verde**: Para fugir dos inconvenientes de percas de documentos e, além disso, alcançar o lema "ecologicamente correto", muitas empresas têm adotado uma ferramenta fundamental para migração das cópias impressas para o formato eletrônico: a Certificação Digital;
- **Entrega da DIPJ (Declaração do Imposto de Renda Pessoa Jurídica)**: Companhias tributadas pelo lucro real ou arbitrado deverão entregar a declaração com certificação digital;
- **Mais produtividade**: Menos papel e mais tempo garantem mais produtividade e competitividade a seus usuários;
- **Evite fraudes digitais**: A Certificação Digital garante sigilo, autenticidade e integridade para você executar transações eletrônicas com mais segurança;
- **Segurança na *Web***: Nos dias de hoje, em que a *Internet* é um dos principais canais de comunicação entre pessoas, à segurança das informações trocadas na rede, assim como a integridade dos *websites*, são pontos essenciais para uma interação positiva e bem-sucedida;
- **O cidadão no centro das atenções**: Órgãos públicos investem em melhorias na gestão de processos para diminuir burocracia e beneficiar o contribuinte;
- **Ferramenta para agilizar processos jurídicos**: Justiça do Trabalho investe em tecnologia para acabar com a papelada que lota tribunais e atrasa julgamentos;
- **SPEED e NE-e: Sistemas Tributário e Fiscal em evolução**: O sistema tributário e fiscal brasileiro vem mostrando sinais de evolução nos últimos anos para agilizar a dinâmica de processos e proteger a comunicação com os contribuintes. O primeiro passo importante nesse sentido foi a criação do Sistema Público de Escrituração Digital (SPED), implementado com o objetivo de modernizar o sistema atual e substituir o repasse em papel de informações aos fiscos por arquivos digitais.

Tipos de certificados

Os tipos básicos de certificados oferecidos no Brasil pelas autoridades certificadoras, são basicamente e-CPF e e-CNPJ do tipo: A1, A2, A3, A4, para assinatura e S1, S2, S3 e S4 para sigilo. Quanto mais alto o número mais complexo é o nível de criptografia do certificado. Existem outros tipos de certificados, como o SSL *(Secure Sockets layer)*, que basicamente é para sites (PINHEIRO; NETO, 2012).

A série que reúne os certificados de assinatura digital, utilizados na confirmação de identidade na *Web*, em *e-mail*, em redes privadas virtuais (VPN) e em documentos eletrônicos com verificação da integridade de suas informações é a A (A1, A2, A3 e A4). Já a séria que reúne certificados de sigilo, que geralmente são utilizados na codificação de documentos, de bases de dados, de mensagens e de outras informações eletrônicas sigilosas é a série S (S1, S2, S3 e S4). Esses oitos tipos de certificados se diferem pelo uso, pelo nível de segurança e pela validade.

O armazenamento das chaves privadas desses certificados difere uns dos outros. Por exemplo, nos certificados do tipo A1 e S1 as chaves privadas ficam armazenadas no próprio computador do usuário. No entanto, nos tipos A2, A3, A4, S2, S3 e S4 as chaves e informações referente ao certificado ficam armazenadas em um *hardware* criptográfico – cartão inteligente (*Smartcard* –figura 6) ou cartão de memória (*Token* USB ou *Pen Drive* – figura 6). Para acessar as informações é necessário utilizar a senha pessoal determinada no momento da compra.

Figura 6 *– Tipos de Certificados Digitais.*

2.3 Regulamentações do uso de documentos e assinaturas digitais no Brasil

Atualmente, no Brasil, a norma que disciplina o uso dos documentos e assinaturas digitais tema é a Medida Provisória nº. 2.200-02, de 24 de agosto de 2001. É importante destacar que esta medida provisória, apesar de ter sido publicada há alguns anos, ainda está em vigor, em razão do que expressa o artigo 2º da Emenda Constitucional no. 32, de 11/09/2001 (LACORTE, 2006).

O governo brasileiro apresentou essa medida provisória de regulamentar o uso de certificados digitais no país com objetivo de usá-los nas transações *online* entre os vários órgãos públicos e seus fornecedores. A ideia era dar valor legal, permitindo maior agilidade no processo de compras e a diminuição de custos com uso, gerenciamento e armazenamento de documentos oficiais sigilosos ou não sigilosos (ARAUJO; VIEIRA, 2012).

A primeira Infraestrutura de Chaves Públicas Brasileiras chamava-se ICP-Gov, posteriormente transformou-se na ICP-BRASIL. A Infraestrutura de Chaves Pública Brasileira (ICP-Brasil) é uma cadeia hierárquica e de confiança que viabiliza a emissão de certificados digitais para identificação virtual do cidadão. Observa-se que o modelo adotado pelo Brasil foi o de certificação com raiz única, sendo que o Instituto Nacional de Tecnologia da Informação (ITI), além de desempenhar o papel de Autoridade Certificadora Raiz (AC-Raiz), também tem o papel de credenciar e descredenciar os demais participantes da cadeia, supervisionar e fazer auditoria dos processos (ITI, 2013).

ICP-Brasil tem por finalidade a garantia de autenticidade, integridade e validade jurídica dos documentos produzidos de forma eletrônica. A infraestrutura da ICP-BRASIl como mostra a figura 7, é formada pela Autoridade Certificadora Raiz (AC Raiz), pelas Autoridades Certificadoras (AC's) e pelas Autoridades de Registro (AR's). De acordo com Araújo e Vieira (ARAUJO; VIEIRA, 2012) existem alguns casos em que somente as transações realizadas com certificados emitidos por autoridades credenciadas na ICP-BRASIL tem validade reconhecida, como exemplo as transações com a Secretaria da Receita Federal.

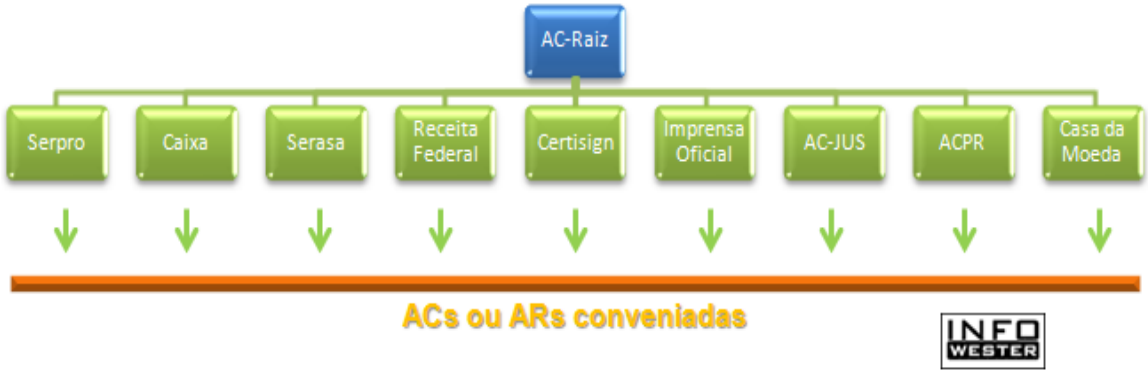


Figura 7 *– Estrutura da ICP-Brasil.*

São essas instituições que devem ser procuradas por quem deseja obter certificado digital legalmente reconhecido no Brasil. Note que cada uma dessas entidades pode ter critérios distintos para a emissão de certificados, o que inclusive resulta em preços diferentes, portanto, é conveniente ao interessado saber qual AC é mais adequada às suas atividades. Repare também que essas entidades podem ter ACs "secundárias"ou ARs ligadas a elas (ALECRIM, 2011).

3 Desenvolvimento e resultados

Como mostrado nos capítulos anteriores este trabalho se propõe ao desenvolvimento de um serviço de autenticação de documentos digitais, mais especificamente planos de cursos para o curso de Sistemas de Informação da Universidade Federal do Piauí campus de Picos, o qual será ofertado pelo *site* do curso, e o site desenvolvido utiliza técnicas de adaptabilidade e responsividade. Assim este capítulo descreve o modelo proposto neste trabalho, além de detalhar as tecnologias utilizadas, a implementação do Serviço de Autenticação de Documentos Digitais e do *site* Adaptável com *Design* Responsivo, e mostrar os resultados obtidos.

3.1 Tecnologias

O sistema operacional usado para fornecer a gerência e a interação das tarefas no decorrer do desenvolvimento da aplicação foi o Linux Ubuntu 12.04 LTS 32-*bits, software* livre e que não precisa de programas anti-vírus. Também, foram utilizadas algumas tecnologias ao decorrer do desenvolvimento, tais como: a linguagem *Ruby, Framework Ruby on Rails*, Editor de texto *Gedit* e algumas *Gems* para auxiliar em algumas funcionalidades do sistema como: *upload* de arquivos, manipulaçao de pdfs, geração de *hash*, etc.

3.1.1 Linguagem *Ruby*

A linguagem *Ruby* foi desenvolvida no Japão em 1993 por Yukihiro Matz Matsumoto, inspirada, principalmente, por linguagens como *Python, Perl, Lisp, Eiffel, Ada e Smalltalk*, tendo grande similaridade, principalmente com *Python* (RUBY-LANG.ORG, 2013).

Atualmente, *Ruby* ocupa o 11º lugar no *ranking* das linguagens de programação mais populares do mundo, segundo o Índice Tiobe (TIOBE, 2013). Existem diversas implementações alternativas de *Ruby*, alguns exemplos são *JRuby* (KENAI, 2013), e *IronRuby* (FRIEDMAN, 2010), cada uma com suas particularidades referente a interpretação e compilação.

Segundo Matz: " *Ruby* é uma linguagem de *scripting* interpretada cujo objetivo é tornar a programação orientada a objeto simples e rápida (...). É simples, direta, extensível e portável" (STEWART, 2001). Além disso, uma de suas principais características é a expressividade que

possui. Pois, desde o começo teve-se como objetivo que *Ruby* fosse uma linguagem muito simples de ler e ser entendida, para facilitar o desenvolvimento e manutenção de sistemas escritos com ela. Entretanto, *Ruby* é interpretada e, como tal, necessita da instalação de um interpretador antes de executar algum programa (CAELUM, 2013).

O IRB *(Interactive Ruby Shell)* é um dos principais recursos disponíveis aos programadores *Ruby*. Funciona como um *console*/terminal, e os comandos vão sendo interpretados ao mesmo tempo em que vão sendo inseridos, de forma interativa. O IRB avalia cada linha inserida e já mostra o resultado imediatamente (CAELUM, 2013). Além disso, *Ruby* apresenta alguns recursos interessantes que foram inspirados e aproveitados de outras linguagens de sucesso, tornando-a assim uma linguagem bastante flexível. Oliveira (OLIVEIRA, 2006) lista alguns recursos apresentados pela linguagem:

- Possui sintaxe simples, parcialmente inspiradas por Eiffel e Ada.
- Possui recursos para tratamento de exceções, assim como na linguagem Java e *Python*, facilitando o tratamento de erros;
- Operadores podem ser redefinidos facilmente por métodos;
- Orientada a objetos, isso significa que todo dado em *Ruby* é um objeto;
- Desenvolvido para ser aberto a melhorias. Com isso é possível que uma instância de uma classe possa se comportar diferentemente de outras instancias da mesma classe;
- Possui herança única, com a possibilidade de estender módulos. Estes módulos são coleções de métodos, sendo assim, toda classe pode importar um modulo e pegar todos os métodos;
- Possui blocos em sua sintaxe que podem ser passados para os métodos;
- Com a utilização do *garbage coletor*, que atua em todos os objetos, não há necessidade de manter uma contagem de referencias em bibliotecas externas;
- Inteiros são usados sem contar sua representação interna. Há inteiros pequenos e grandes, porém não é preciso definir qual será utilizado devido à ocorrência automática de conversão;
- Se o sistema operacional permitir, há a possibilidade de carregar bibliotecas de extensão dinamicamente;
- Suporte a *threads*;

- Possui grande portabilidade entre os sistemas operacionais.

Similar ao que acontece em Java, em *Ruby* existe uma classe "mãe-de-todas" chamada *Object*, mas as similaridades ficam por aí. Em *Ruby* não existem tipos primitivos. Todas as variáveis são objetos e todos os tipos são classes (RUBY et al., 2009).

Com o surgimento do *framework Ruby on Rails*, o interesse por *Ruby* cresceu ainda mais, sendo dado ao *Rails*, frequentemente, o crédito de ter tornado a linguagem famosa. A ligação entre linguagem e o *Framework* é tão forte que, muitas vezes, os iniciantes em *Ruby* confundem a linguagem com o *Framework* (CARLSON; RICHARDSON, 2006).

Por fim, neste trabalho utilizou-se a versão da linguagem *Ruby* 1.9.3. Para instala-la faz-se necessário ter um SO compatível, nesse caso o Ubuntu 12.04, abrir o terminal e digitar os comandos: `sudo apt-get install ruby1.9.3` e então inicia-se o *download* dos pacotes. Após isso, verifica-se digitando o comando `ruby -v` no terminal, se a instalação foi concluída com sucesso.

Gems

Em diversas linguagens modernas é comum utilizar-se bibliotecas para auxiliar e acelerar o desenvolvimento. No universo *Ruby* essas bibliotecas são distribuídas através de *gems*, que são arquivos que terminam com a extensão .gem, onde ficam localizados o condigo fonte da biblioteca e também informações e metadados, como versão e nome. E existem diversos tipo de *gems*, para todos os tipos de funções tais como: *upload* de arquivos, criação de pdfs, autenticação de usuários, *layouts*, etc. Para instalação dessas *gems*, e necessário abrir o arquivo *Gemfile*, que existe em todo projeto *Rails* e especificar a *gem* e a versão requerida, por exemplo: `gem 'bootstrap-sass', ' > 3.1.0'`. Nele encontra-se a relação de todas as *gems* que estão sendo utilizadas no projeto. Após isso digita-se o comando "`bundle install`" no terminal e inicia-se o *download* dos pacotes.

Nessa aplicação foram utilizadas algumas dessas *gems*, tais como:

- `Gem paperclip`: essa *gem* está sendo utilizada para fazer *uploads* de arquivos, ela reduz a complexidade do *upload* e do processamento. Além de que, ela utiliza a biblioteca *ImageMagick* para manipular imagens, com isso, deve-se garantir que ela esteja instalada. Para utilizar o *paperclip* o único requisito e possuir a versão do *Ruby e Rails* a partir da 1.9.2 e 3.0 respectivamente.

- `Gem "prawn", " > 0.14.0"`: é uma *gem* que trabalha com pdfs, desde da criação ate a manipulação dos mesmos. É extremamente simples, muito rápida e completa.

- `Gem 'bcrypt-ruby', ' > 3.0.0'`: essa *gem* é um algoritmo *hash* de senha, que

permite o armazenamento de um *hash* seguro de senhas dos usuários. Garantindo que se alguém acessar indevidamente o banco de dados da aplicação, não descobrirá as senhas armazenadas.

- `Gem 'bootstrap-sass', ' > 3.1.0'`: é uma versão sass do *bootstrap*, que e integrada com *Rails*. Através dele consegue-se deixar o *designer* do sistema responsivo.

- `Gem 'jquery-rails'`: é uma *gem* que automatiza o uso de *JQuery* com *Rails*. Ela fornece arquivos *JQuery, JQuery UI*, adaptador *JQuery UJS*, etc.

3.1.2 *Framework Ruby on Rails*

O *Rails* foi criado em 2004 por David Heinemeier Hansson e desde então foi expandido pelo time central do *Rails*, com mais de 3.000 contribuidores. Além disso, trata-se de um *Framework* que inclui tudo o que é necessário para criar aplicações *Web*. O *Framework* é escrito sobre a linguagem interpretada *Ruby*, utilizando o paradigma de orientação a objetos. Ele é desenvolvido sobre a arquitetura MVC (*Model-View-Controller*), que é amplamente utilizada em desenvolvimento *Web* por manter separada a camada de visualização das camadas de lógica e regras de negócios da aplicação. Essa separação se torna atraente para a estrutura de uma aplicação baseada na *Web*, pois essas aplicações costumam ser desenvolvidas para diferentes navegadores de *Internet* que possuem diferentes padrões de exibição e estilização de suas paginas. Com isso, a separação da camada de visão permite que o problema de incompatibilidade dos navegadores seja tratado em uma camada apenas, e não na aplicação com um todo (BURBECK, 1992).

Segundo Vinícius Baggio Fuentes (FUENTES, 2013) a arquitetura MCV – *Model- View-Controller*, ou Modelo, Apresentação e Controle adotada pelo *Ruby on Rails* pode ser definida da seguinte maneira:

- **Modelos**: possuem duas responsabilidades: eles são os dados que normalmente ficam persistidos em um ou mais banco de dados (o perfil de usuário, por exemplo). Eles também fazem parte da regra de negocio, ou seja, cálculos e outros procedimentos, como verificar se uma senha é valida.

- **Controle**: é a camada intermediaria entre a *Web* e o seu sistema. Ele pega os dados que vem de parâmetros na *URL* e/ou de um formulário e repassa para os modelos, que vão fazer o trabalho pesado. Em seguida, pega o resultado e transforma da maneira adequada para a Apresentação.

- **Apresentação**: é como o aplicativo mostra o resultado das operações e os dados. Normalmente podem ser uma bela página usando as novas tecnologias de CSS3 e HTML 5 e até pequenas representações de objetos em JSON (*JavaScript Object Notation*).

O *Rails* possui dois princípios básicos de existência. O DRY (*Don't repeat yourself*) e a "Convenção sobre configuração". O DRY estimula a diminuição do retrabalho visando à melhora de produtividade, ou seja, se uma característica da aplicação já foi declarada uma vez, ela não deve ser redeclarada em outro local. O *Framework* deve ser inteligente o suficiente para buscar o local da primeira declaração. O principio da "Convenção sobre configuração" reflete que o *Framework* não deve ser configurável para cada tipo de aplicação que ele irá manter, mas sim que cada aplicação deve tomar certas convenções para que ela funcione corretamente sobre o ambiente *Rails*. Essas convenções são basicamente os locais onde cada tipo de arquivo será armazenado dentro da estrutura do *Framework* e a nomenclatura desses arquivos e classes. Com essas simples convenções o *Rails* consegue carregar e manipular as estruturas das quais necessita ao executar uma aplicação (AKITA, 2006).

O *Ruby on Rails* é um *meta-framework*, ou seja, um *Framework* de *Framework*, composto por alguns *Frameworks* como: *Active Record, Action Pack, Action Mailer, Action Support*. Cada um desempenhando seu papel de forma a tornar o *Rails* um diferencial no mercado (CAELUM, 2013). Segundo Evan Williams, criador do *Blogger* e do ODEO: "Depois de pesquisar o mercado, *Ruby on Rails* se demonstrou a melhor opção. Nós estamos felizes com esta decisão, e vamos continuar a utilizar o *Rails* e a considerar ele como uma vantagem competitiva." (RUBYONRAILS, 2013).

Por fim, nesta aplicação utilizou-se a versão 4.0.0 do *Framewrok Ruby on Rails*. Para instala-lo é necessário ter um SO compatível, nesse caso o Ubuntu 12.04, abrir o terminal e digitar os comandos: `sudo gem install rails -version 4.0.0` e então inicia-se o *download* dos pacotes. Após isso,verifica-se digitando o comando `rails-v` no terminal, se a instalação foi concluída com sucesso.

3.1.3 Editor de texto Gedit

O Editor de texto *gedit* é o editor de texto padrão GUI (*graphical user interface*) no sistema operacional Ubuntu. Ele é compatível com a codificação UTF-8 e suporta a maioria dos recursos de um editor de texto padrão, assim como muitas características avançadas. Dentre estas características incluem multilinguagem, verificação ortográfica, suporte extensivo ao destaque de sintaxe e um grande número de *plugins* oficiais de terceiros. O procedimento de instalação e utilização foi através de linha de comando, digitando no terminal o seguinte comando:`sudo apt-get install gedit`.

3.2 Usabilidades e Responsividade

Este sistema foi desenvolvimento utilizando técnicas de usabilidade e responsividade. A fim de proporcionar uma *interface* fácil de usar e adaptável a qualquer dispositivos móveis existentes no mercado. Resultando em maior comodidade para o usuário.

3.2.1 Usabilidade

Interface é um fator importantíssimo na busca da qualidade dos *softwares*. Através da *interface* um *software* ou site pode ser rejeitado ou não. Pois, o usuário qualifica-o de acordo com sua parte visível e com a qual ele interage. A área que estuda a relação Humano-Computador é chamada de Interação Humano Computador ou apenas IHC. Ela foca na interação do usuário com o sistema e nos resultados práticos para o projeto da *interface* (PRATES;BARBOSA,2007). Com isso, buscam-se *interfaces* de qualidade, bem como a avaliação da usabilidade das mesmas.

A usabilidade é a qualidade que caracteriza o uso de um sistema interativo. Ela se refere a relação que se estabelece entre usuário, tarefa, interface , equipamento e demais aspectos do ambiente no qual o usuário utiliza o sistema (CYBIS, 2010). Assim uma *interface* eficiente é a principal motivação que leva as pessoas a utilizarem um sistema. Em 1998, cerca de três bilhões de dólares deixaram de ser ganhos na *Web* norte-americana por causa de *design* mal projetado, as páginas que os usuários acessavam dificultavam a compra de produtos em vez de facilitar.

A utilização de técnicas de usabilidade facilitam aos *web designers* o desenvolvimento de *interfaces* interativas, fornecendo aos mesmos conhecimento para utilizarem determinadas metodologias e padrões para obter eficiência e eficácia na utilização de *sites*, bem como a facilidade de acesso à informações e funcionalidades, sempre objetivando a satisfação do usuário (OLIVEIRA, 2013).

Nessa seção serão detalhados os procedimentos utilizados para realização do estudo de usabilidade proposto para o *site*. Assim, serão mostradas as heurísticas de usabilidade propostas por Nielsen, Em seguida, faremos uma descrição dos critérios ergonômicos propostos pelos pesquisadores Dominique Scapin e Christian Bastien (BASTIEN; SCAPIN, 1993), que são utilizados para analisar os diversos componentes de uma *interface* e a interação do usuário com a mesma.

Heurísticas de usabilidade

As heurísticas utilizadas no estudo da usabilidade desse *site* foram as definidas por Nielsen (NIELSEN, 1993), no total são 10 heurísticas. Elas são empregadas para desenvolvimento

ou avaliação de *interfaces* de sistemas *web*. As mesmas tem como foco facilitar o desenvolvimento de *layouts* interativos agradáveis, eficientes e eficazes na execução de tarefas feitas pelo usuário.

1. **Visibilidade do estado do sistema**: a *interface* sempre deve informar ao usuário o que esta acontecendo, ou seja, todas as ações precisam de *feedbacks* instantâneos para orientá-los.

2. **Mapeamento entre o sistema e o mundo real**: o sistema deve manter um linguajar simples, que faça sentido para o usuário. Evitar utilizar palavras de sistema, jargões de programadores. Pois, toda a comunicação precisa ser contextualizada ao usuário, e ser coerente com o chamado modelo mental do mesmo.

3. **Liberdade e controle ao usuário**: o sistema deve facilitar as "saídas de emergências" para os usuários, permitindo-os desfazer ou refazer ações no sistema e retornar ao ponto anterior, quando o mesmo se sentir perdido ou em situações inesperadas.

4. **Consistência e padrões**: é recomendável que o sistema mantenha o padrão, em relação a ícones, cores, ou seja, que ele fale a mesma língua o tempo todo e nunca identifique a mesma ação com ícones ou palavras diferentes. Trate coisas similares, da mesma maneira, facilitando a identificação do usuário.

5. **Prevenção de erros**: de acordo com Nielsen "Ainda melhor que uma boa mensagem de erro é um design cuidadoso que possa prevenir esses erros". Por exemplo, ações definitivas, como deleções ou solicitações podem vir acompanhadas de um *checkbox* ou uma mensagem de confirmação.

6. **Reconhecer em vez de relembrar**: Evite acionar a memória do usuário o tempo inteiro, fazendo com que cada ação precise ser revista mentalmente antes de ser executada. Permita que a *interface* ofereça ajuda contextual, e informações capazes de orientar as ações do usuário – ou seja – que o sistema dialogue com o usuário.

7. **Flexibilidade e eficiência de uso**: O sistema precisa ser fácil para usuários leigos, mas flexível o bastante para se tornar ágil à usuários avançados. Essa flexibilidade pode ser conseguida com a permissão de teclas de atalhos, por exemplo. No caso de *websites*, uso de máscaras e navegação com tab em formulários são outros exemplos.

8. ***Design* estético e minimalista**: Evite que os textos e o *design* fale mais do que o usuário necessita saber. Os "diálogos" do sistema precisam ser simples, diretos e naturais, presentes nos momentos em que são necessários.

9. **Suporte para o usuário reconhecer, diagnosticar e recuperar erros**: As mensagens de erro do sistema devem possuir uma redação simples e clara que ao invés de intimidar o usuário com o erro, indique uma saída construtiva ou possível solução.

10. **Ajuda e documentação**: Um bom *design* deveria evitar ao máximo à necessidade de ajuda na utilização do sistema. Ainda assim, um bom conjunto de documentação e ajuda deve ser utilizado para orientar o usuário em caso de dúvida. Deve ser visível, facilmente acessada, e oferecer uma ferramenta de busca na ajuda.

Critérios Ergonômicos

O princípio da ergonomia é estudar o conforto e a adaptação aos objetos e *interfaces* visando aumentar a produtividade e a satisfação das pessoas. Na informática, o objetivo da ergonomia é a adequação dos sistemas aos usuários e às tarefas que eles executam, de modo a proporcionar aumento de produtividade, (CHRISTOL, 1987) sem desrespeitar a inteligência dos mesmos e permitindo (MONTMOLLIN, 1986) também, a melhoria de suas competências.

Os pesquisadores Scapin e Christian Bastien (BASTIEN; SCAPIN, 1993) propuseram um conjunto de critérios ergonômicos, com o objetivo de proporcionar o aumento da sistematização dos resultados das avaliações de uma dada *interface*. Isto é, quando diferentes especialistas empregam esses critérios como ferramentas de avaliação, eles obtêm resultados mais parecidos, diminuindo, assim, a falta de sistematização.

Nesta seção serão detalhados os critérios ergonômicos, os quais são compostos por um conjunto de oito critérios que se subdividem em dezoito subcritérios, visando assim reduzir ao máximo a redundância na identificação e classificação de qualidades (CYBIS, 2010).

Os critérios principais, subcritérios e critérios elementares são mostrados a seguir:

1. **Condução**: visa favorecer o aprendizado e a utilização do sistema por usuários novatos. A *interface* deve nortear o usuário na interação para com o sistema. Este critério possui cinco subcritérios, são eles: convite, agrupamento, e distinção entre itens, legibilidade e *feedback* imediato.

2. **Carga de trabalho**: Este critério tem por ação reduzir ao máximo a execução das atividades repetitivas, desta forma minimizará a necessidade de percepção, cognição e movimentação. Pode-se aplicar esta qualidade evitando leituras redundantes e memorizações dispensáveis, além de mudança de formulários ou repetições. Reduzindo a carga de trabalho, há a redução de erros por distrações ou fadiga. Este critério se subdivide em: brevidade e densidade informacional.

3. **Controle Explicito**: Diz respeito ao processamento das ações dos usuários feitas pelo

sistema, bem como o controle que o usuário tem sobre o processamento de suas ações pelo sistema. Este critério se apresenta sob as subclassificações: ações explícitas do usuário e controle do usuário.

4. **Adaptabilidade**: Busca um equilíbrio para o nível de usabilidade propondo maneiras variadas de se propor uma tarefa, fazendo com que sua execução seja satisfatória. Para que este subcritério seja respeitado é necessário que se verifique a flexibilidade e a consideração da experiência do usuário.

5. **Gestão de erros**: diz respeito a todos os mecanismos que permitem evitar ou reduzir a ocorrência de erros, e quando eles ocorrerem o sistema deve oferecer a solução para correção. Para isso, são integrados a proteção contra erros, a qualidade das mensagens de erro e a correção de erros.

6. **Homogeneidade/consistência**: Usuários inexperientes geralmente utilizam um *software* se embasando em outra tela do próprio sistema. Nota-se assim a necessidade de padronizar o projeto de *interface*, mantendo uma estabilidade gráfica e comunicativa através de formatos, códigos, denominações e procedimentos contidos no *software*. Quando um usuário sabe onde, por padrão, estará certo botão, descrição, *interface* de ajuda e suporte, etc, a transição de uma tela para outra se tornará mais previsível e amigável.

7. **Significados de códigos e denominações**: diz respeito à relação entre o objeto ou a informação apresentada ou pedida e sua referência na *interface*.

8. **Compatibilidade**: Este critério propõe que o sistema deve ser compatível com usuário em suas características, tais como: preferências, cognição, hábitos, expectativas e de acordo com sua capacidade de utilização. Quando o usuário consegue se inserir no contexto do sistema há um desenvolvimento natural e melhor compreensão.

Segundo (BASTIEN; SCAPIN, 1993) os critérios ergonômicos não devem ser a única fonte de análise ergonômica de sistemas interativos, mas sim serem utilizados juntamente com outros métodos de avaliação para obtenção de resultados mais precisos.

3.2.2 Responsividade

Além da usabilidade e dos critérios ergonômicos, o sistema foi desenvolvimento utilizando também técnicas de responsividade, ou seja, a capacidade da interface se adaptar aos diferentes tamanhos de telas dos dispositivos móveis.

O *design* Responsivo está bastante notória atualmente. O aumento do número de dispositivos móveis, o aumento na velocidade da *Internet* para eles, como 3G MAX, *PLUS*, 4G e

a melhoria dos sistemas operacionais para os móveis contribuem para o uso do design responsivo, principalmente por conta dos tablets e celulares de tamanhos e resoluções cada vez mais variadas.

Neste novo contexto dos navegadores e várias resoluções, o design responsivo surge como uma evolução lógica do *design* de *sites*, também conhecido como *web design*. Antes grande parte da *Internet* era acessada por resoluções e navegadores muito semelhantes. Até pouco tempo atrás bastava fazer um *site* que funcionava em *Internet Explorer* com resolução máxima de 1024 x 768 *pixels* que tudo estava resolvido, claro existiam outras características, mas a grande maioria estava nesse mesmo grupo – no máximo havia os usuários de *Mozilla Firefox*. Hoje tudo mudou, temos TVs de 50" polegadas acessando *internet*, temos celulares que tem telas de 2" até 5", tablets de 6" até 11" polegadas e sem contar os próprios computadores, que tem telas de *netbook* até os mais novos *iMacs* da *Apple*, colocando uma margem de 11" até 26" polegadas(ALTERMANN, 2012).

O *site* responsivo usa um único código HTML e, por meio de media queries do CSS3 e outras técnicas de redimensionamento e tratamento de imagens, consegue manter a integridade essencial do *site*, trabalhando com apenas uma folha de estilo e, mesmo assim, preparando seu *site* para as diversas resoluções (BIAZOTTI, 2013).

Esta característica demonstra que um *site* pode ser visto de diversas formas e em diversos contextos, e é para isto que os *sites* devem estar preparados. O *design* responsivo, como o próprio nome já indica, consegue responder ao tamanho da tela para se adequar da melhor forma. Ao invés de criar dois *sites* separados, um para *mobile* e um para *desktops*, como era muito comum, hoje faz-se apenas um *site* que vai se adaptar muito bem a qualquer tela em que ele for carregado (ALTERMANN, 2012).

Além disso, existem algumas ferramentas úteis que facilitam a construção de *sites* responsivos tais como: *Responsive Web Design Sketch Sheets, Adobe Edge Inspect, Foundation, RWD Calculator, Responsive Layouts, Responsively Wireframed, Adaptive Images, Bootstrap, Retina Images, SimpleGrid, The 1140px CSS Grid, resizeMyBrowser, The Responsinator* etc. Cada uma com suas particularidades. Neste trabalho foi utilizado o *Framework* de *front-end Boostrap*, o qual sera descrito detalhadamente na seção seguinte.

Framework de front-end Bootstrap

O *Bootstrap* é um *framework* de *front-end*, ou seja, é uma abstração que une códigos comuns entre vários projetos de *software* provendo uma funcionalidade genérica (BIAZOTTI, 2013) e está relacionado apenas ao desenvolvimento de *layouts* de *softwares*.

O *Bootstrap* foi criado no *Twitter* pelos *designers* Mark Otto e Jacob Thornton com objetivo de facilitar na padronização e no desenvolvimento HTML/CSS e *Javascript*, tanto para

programadores iniciantes, quanto para avançados que desejam se aprofundar no desenvolvimento *web* mais complexo (MAGNO, 2012) . E um *framework front-end* de código aberto (*opensource*). Em palavras simples, é um conjunto de ferramentas criadas para facilitar o desenvolvimento de *sites* e sistemas *web*.

Compatível com HTML5 e CSS3, este *framework* utiliza a técnica de *design* responsivo, ou seja, consegue responder ao tamanho da tela para se adequar da melhor forma (ALTERMANN, 2012), dependendo da resolução da tela do dispositivo móvel do usuário.

Como esse sistema foi desenvolvido em *Rails*, então optou-se por utilizar uma gem chamada *sass-boostrap*, que tem as mesmas funcionalidades do *bootstrap* tradicional, porém foi criada especificamente para trabalhar integrada com *Rails*. Com isso, facilitou o desenvolvimento, tornando-o mais prático.

Para instalação dessas *gems*, faz-se necessários alguns procedimentos tais como: adicionar o nome da *gem* no arquivo *Gemfile*, que se encontra no projeto *rails*. Após isso, executa-se no terminal o comando *bundle install*, para carregar os novos pacotes requisitados, nesse caso o *sass boosttrap*. Feito isso, é necessário renomear o arquivo *application. css* para *application.css.scss*. Esse scss significa que esta sendo utilizada a versão *sass do bootstrap*. E então adiciona-se as linhas `@import "bootstrap";` e `//= require bootstrap` nos arquivos *application.css.scss e application.js* respectivamente. Com isso, o próprio *Rails* automaticamente certifica-se de encontrar os arquivos do *bootstrap* e integra-los ao projeto. Poupando assim trabalho ao programador.

3.2.3 Diagrama de casos de uso

O sistema é composto por dois tipos de usuários, o super usuário e o usuário comum como mostra a figura 8. O primeiro possui mais “poder” sobre o sistema. Porém o segundo possui algumas restrições como descrito a seguir.

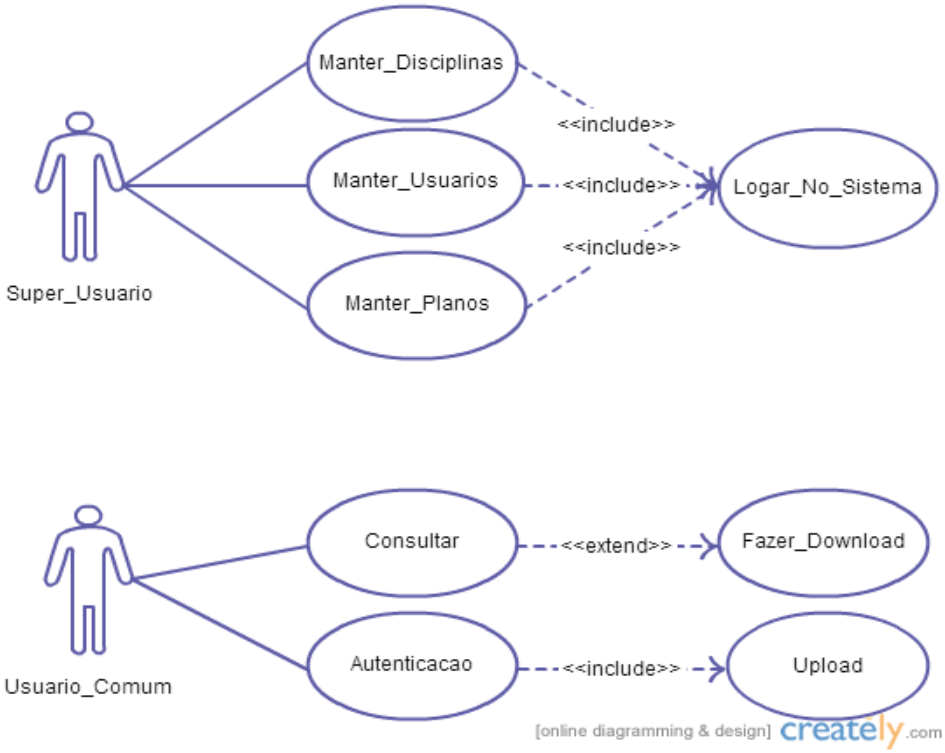


Figura 8 *– Diagrama de casos de uso*

Descrição dos casos de uso:

- **Fazer cadastro, editar e excluir disciplinas, planos de curso e usuários:**
  - **Ator:** apenas o super usuário está apto para executar tais ações.
  - **Descrição:** para executar tais ações o super usuário precisa obrigatoriamente está logado no sistema.
- **Consultar:**
  - **Ator:** o usuário comum está apto apenas para fazer consultas e download dos planos de curso cadastrados no sistema.
  - **Descição:** para executar tais ações, o usuário comum não precisa ser cadastrado no sistema.

3.2.4 Diagrama de classe

O sistema possui no total três tabelas como mostra na figura 9, que são elas: disciplina, plano de curso e usuário. Cada uma com seus atributos e métodos. A tabela disciplina possui o relacionamento de um para N com a tabela de planos de curso. Ou seja, uma disciplina poderá

ter mais de um plano de curso cadastrado referenciando-a. Já a tabela de plano de curso possui o relacionamento de um para um com a tabela de disciplinas, isso indica que um plano de curso pode pertencer apenas a uma disciplina. Por fim, a tabela usuário não possui relacionamento com nenhuma outra.

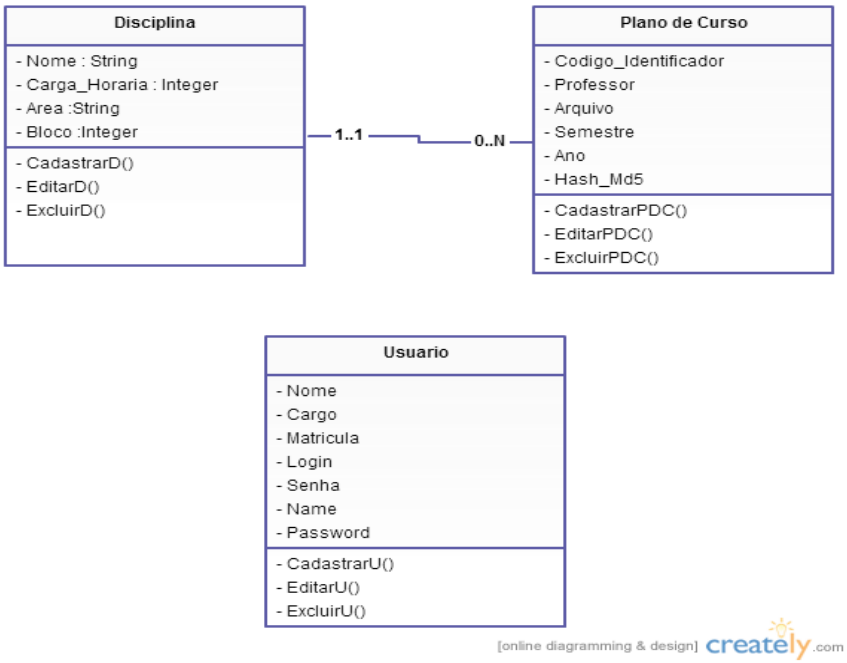


Figura 9 *– Diagrama de classes*

Descrição das classes:

- **Tabela Disciplina:** serve para armazenar todas as informações referente as disciplinas que os super usuários cadastrarem. Seus atributos são: nome, carga-horaria, área, bloco e cadastrar. Além disso, ela possui os métodos cadastrar, editar e excluir. Que são as ações que os super usuários podem executar.

- **Tabela Plano de curso:** serve para armazenar os planos de curso que os super usuários cadastrarem e as informações referentes aos mesmos. Seus atributos são: código identificador, professor, arquivo, semestre, ano e *hash* md5. Além disso, ela possui os métodos cadastrar, editar e excluir. Essas ações podem ser escutadas apenas pelos super usuários.

- **Tabela Usuário:** serve para armazenar as informações dos super usuários que forem cadastrados no sistema. Seus atributos são: nome, cargo, matricula, *login*, senha, *name* e *password*.

3.3 Descrições da *Interface* e funcionamento da autenticação do site desenvolvido

Este sistema é um site, com algumas funcionalidades sendo a principal delas a autenticação de documentos digitais. Ele foi desenvolvido utilizando técnicas de adaptação através de design responsivo e utilizaram-se teorias de usabilidade estabelecidas por Nielsen para criação da *interface*. Com o objetivo de proporcionar uma interface fácil de usar, tornando o acesso mais simples para o usuário.

3.3.1 Descrição da *Interface*

Nessa seção é mostrada a *interface* do *site* bem como sua avaliação de usabilidade utilizando-se princípios e recomendações, os quais foram descritos na seção 3.2.1. Além disso, cada exemplo é composto de imagens da *interface* testadas tanto no *notebook* quanto no *smartphone*, a fim de evidenciar o *design* responsivo e mostrar como ela se dispõe em cada um deles.

Página Principal

A página principal é composta por um menu superior, o conteúdo e três colunas como mostram as figuras 10 e 11 dispostas no *notebook* e *smartphone* respectivamente.

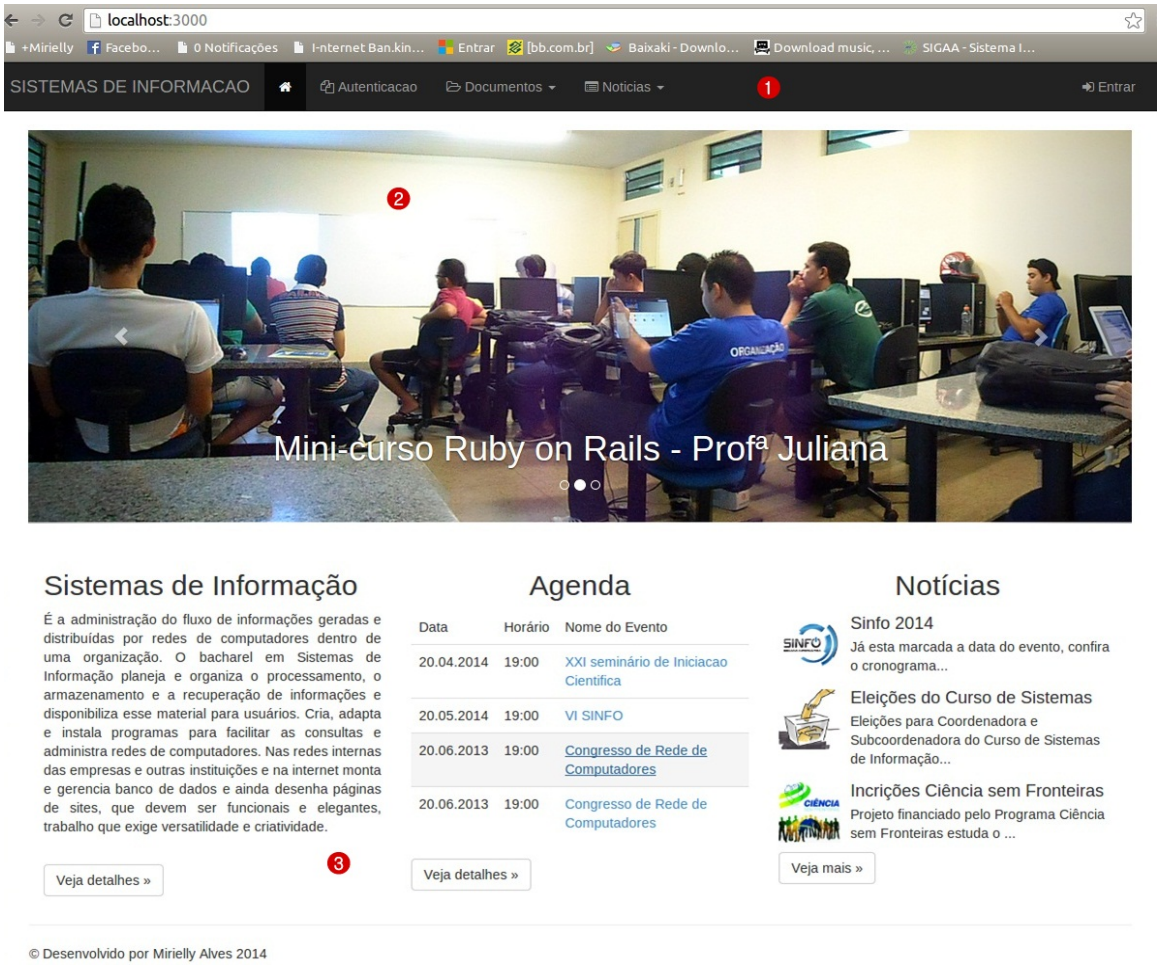


Figura 10 *– Página inicial disposta no notebook.*

Figura 11 *– Página inicial disposta no smartphone.*

1. **Menu superior:** contém *links* que direcionam para outras páginas, alguns em formato *dropdown*. Esses ficam ocultos e quando o usuário necessitar acessa-los, o mesmo clica

no botão e os *links* ocultos serão exibidos como mostra o item 1 das figuras 12 e 13 disposta no *notebook* e no *smartphone* respectivamente. Essa característica facilita ao usuário a leitura das informações contidas no *site*, devido ao menu não ocupar muito espaço, ser organizado e não acumular muitas informações, pois alguns *links* estão ocultos. Além disso, os *links* estão organizados em ordem alfabética como mostra o item 2 das figuras 12 e13. Essa característica facilita ao usuário reconhecer em vez de lembrar, diminuindo a carga cognitiva do mesmo (critério 6 da seção 3.2.1 - Heurísticas de Usabilidade). Outro fator relevante e o contraste obtido através da cor clara da fonte e o fundo escuro. Isso facilita a leitura para pessoas com problema de visão e idosos (subcritério, do critério 1 da seção 3.2.1 - Critérios Ergonômicos). Além disso, os *links* em forma de *dropdown* seguem o agrupamento e distinção por localização como mostra o item 3 das figuras 12 e 13 dispostas no *notebook* e *smartphone* respectivamente, (subcritério, do critério 1 da seção 3.2.1 - Critério Ergonômicos) favorecendo a aprendizagem e a utilização do *site* por usuários novatos. Por fim, após o super usuário efetuar o *login* aparecerá para ele mais dois *links* no menu que são eles: admin e sair (item 4 das figuras 12 e 13). Nota-se que utilizou-se palavras de conhecimento do usuário, a fim tornar a linguagem do site simples e ser o mais coerente possível com o modelo mental formulado pelo o mesmo (critério 2 da seção 3.2.1 - Heurísticas de Usabilidade).

Figura 12 *– Menu-navbar diposto no notebook.*

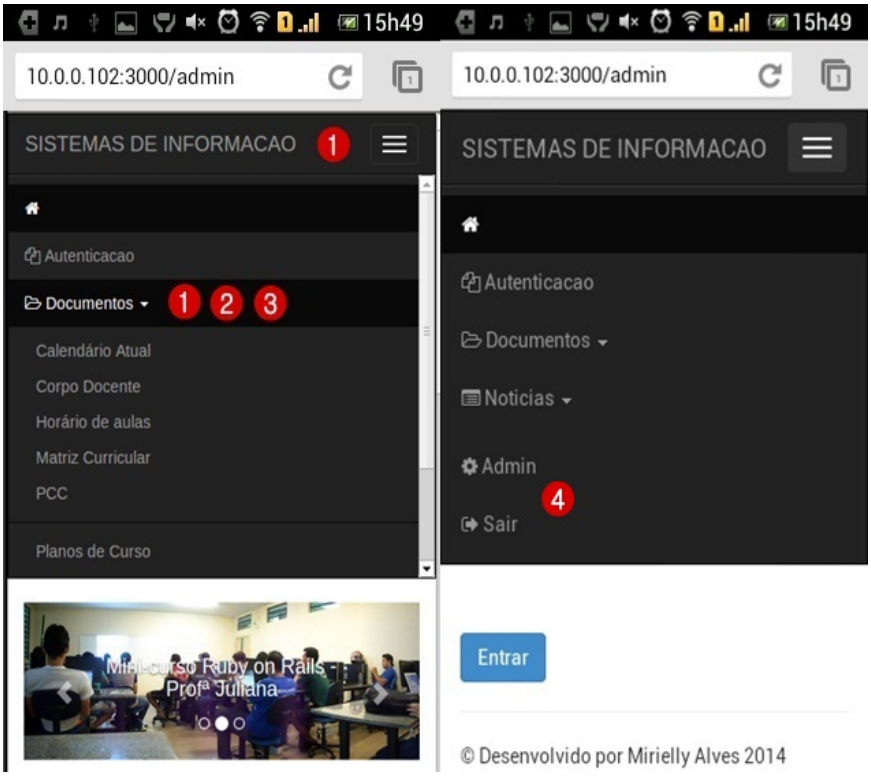


Figura 13 *– Menu-navbar disposto no smartphone.*

2. **Conteúdo:** contem o conteúdo principal da pagina, nesse caso é exibido um *slideshow* (Item 1 das figuras 14 e 15 dispostas no *notebook e smartphone* respectivamente), com fotos juntamente com *links* representando as notícias principais sobre o curso. O mesmo passa automaticamente, não é necessário o usuário manipula-lo. Dessa forma diminui-se a carga de trabalho do usuário (critério 2 da seção 3.2.1 - Critérios Ergonômicos). Além de, evitar que a interface fique tumultuada e as informações relevantes passem despercebidas (critério 8 da seção 3.2.1 - Heurísticas de Usabilidade).

Figura 14 *– Conteúdo da página inicial disposto no notebook.*

Figura 15 *– Conteúdo da página inicial disposto no smartphone.*

3. **Colunas:** contem três colunas, uma com informações sobre o curso, outra com a agenda dos eventos da área e a ultima com noticias relacionado ao curso (Figuras 16 e 11 dispostas no *notebook* e *smartphone* respectivamente). Essa distribuição em colunas facilita a organização da página principal, fazendo com que seja mais fácil ao usuário obter as informações. Na primeira coluna possui um pequeno texto sobre o curso de sistemas de informação e um botão a fim de direcionar o usuário a outra página com o texto completo. Tornando assim, o diálogo simples, pois os textos na pagina principal não são longos e evita-se a fadiga do usuário (Item 1 figuras 16 e 11 dispostas no *notebook e smartphone* respectivamente) (critério 8 da seção 3.2.1 - Heurísticas de Usabilidade). Essa característica diminui a quantidade de ações arrastar a tela cima/baixo que o usuário deve executar, minimizando a carga de trabalho do mesmo (critério 2 da seção 3.2.1 - Critérios Ergonômicos). Já a segunda coluna possui uma tabela com informações referente aos eventos do curso. Nota-se que é utilizado como padrão para os *links* a cor azul (critério 4 e 6 da seção 3.2.1 - Heurísticas de Usabilidade e Critérios Ergonômicos respectivamente), como exemplo tem-se os *links* da tabela agenda (Item 2 figuras 11 e 16 disposta no *notebook e smartphone* respectivamente). Pois facilita a identificação do usuário que tal cor esta associada aos *links*.

Figura 16 *– Colunas da página inicial dispostas verticalmente no notebook.*

Página de Listagem de planos de curso

Essa página é onde consultam-se os planos de curso cadastrados e faz-se *download* dos mesmos. Ela é composta por uma tabela onde estão dispostas as informações referentes aos planos de curso como disciplina, código identificador, professor, arquivo para *download* etc. No celular essa tabela se adapta e possui uma barra de rolagem para verificar as colunas. Também possui campos para pesquisa e botão para adicionar plano de curso. Porem para essa ultima ação só é possível para super usuários (Figuras 17 e 18 dispostas no *notebook e smartphone* respectivamente).

1. **Tabela de listagem de planos de curso:** a princípio o site mantem um linguajar simples, como exemplo encontra-se as palavras utilizadas nessa página (Item 1 Figura 10 e 11 dispostas no*notebook e smartphone* respectivamente), que são palavra que fazem sentido para o usuário. Isso torna o *site* simples e coerente com o modelo mental feio pelo usuário do mesmo (critério 2 da seção 3.2.1 - Heurísticas de Usabilidade). Também atribui-se ao usuário comum e ao super usuário a liberdade e o controle no sistema, dando ao primeiro a possibilidade de optar por fazer ou não *download* dos planos de curso (Item 2 das figuras 17 e 18), ou apenas verificar as informações dos mesmos. E ao segundo a possibilidade de cadastrar, editar e excluir disciplinas, planos e usuários (critério 3 da seção 3.2.1 - Heurísticas de Usabilidade). Além disso, vale ressaltar que as tabelas seguem um padrão em todas as paginas do *site* (Figuras 17 e 18). Pois, busca-se que o sistema mantenha o padrão e nunca identifique a mesma ação com ícones ou palavras diferentes. Isso também acontece com os ícones das tabelas como, por exemplo, ícone da lixeira que representa excluir, do olho que significa ver etc (Item 3 da figuras 17 e 18). Já que, estes desempenham a mesma função em todas as páginas que estiverem presentes. Isso facilita

a identificação do usuário (critério 4 e 6 da seção 3.2.1 - Heurísticas de Usabilidade e Critérios Ergonômicos respectivamente). Ademais, os ícones da tabela segue um agrupamento e distinção por formato, a fim de fazer com que o usuário identifique de forma mais rápida a *interface* e reconheça as informações por cor, forma, estilo de escrita etc. (Item 4 figuras 17 e 18)(subcritério, do critério 1 da seção 3.2.1 - Critérios Ergonômicos). Por fim, antes de excluir um dado cadastrado, através do ícone da lixeira nas tabelas, aparece uma janela (Item 1 da 19 disposta no *notebook*), perguntando se o usuário realmente tem certeza de tal ação. Isso se faz importante na prevenção dos erros. Visto que, pode acontecer do usuário se descuidar e clicar sem querer em determinado ícone e acabar realizando uma ação indesejada (critério 5 seção 3.2.1 - Heurísticas de Usabilidade).

2. **Campo de pesquisa:** pode-se fazer pesquisas dos planos de curso (Item 5 figuras 17 e 18) através do semestre, do professor ou ano. Essa função busca reduzir ao máximo o trabalho do usuário em procurar todos os planos de curso cadastrados. Pois efetuando uma pesquisa ele restringe as opções e encontra mais rápido o que precisar. Dessa forma, evita-se leituras redundantes e memorizações dispensáveis, além de reduzir a carga de trabalhado do mesmo (critério 2 da seção 3.2.1 - Critérios Ergonômicos). Outro fator relevante são os botões: novo plano e pesquisa (Item 6 figuras 17 e 18), que além de estarem de acordo com os padrões impostos pelo *site*, indica um modo de obter brevidade na execução de tal ação. Pois em apenas um clique o super usuário ou o usuário comum conseguem abrir a página de cadastro e efetuar uma pesquisa respectivamente. Com isso, reduz-se ao máximo as *interfaces* pelas quais os usuários precisam passar, diminuindo o trabalho e os erros (subcritério, do critério 2 da seção 3.2.1 - Critérios Ergonômicos).

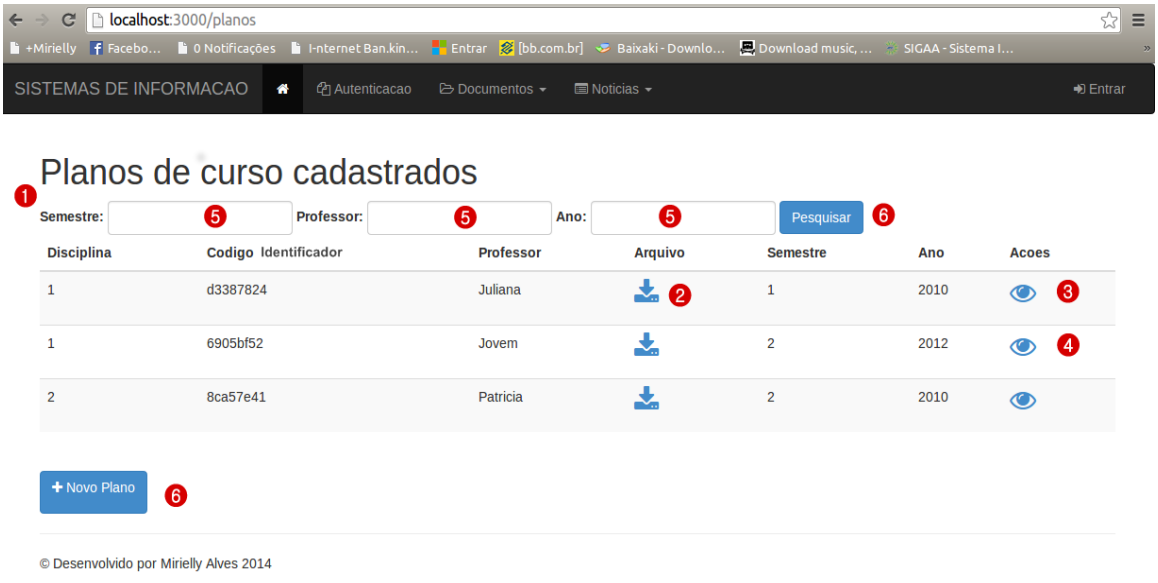


Figura 17 *– Listagem de planos de curso cadastrados disposta no notebook.*

Figura 18 *– Listagem de planos de curso cadastrados disposta no smartphone.*

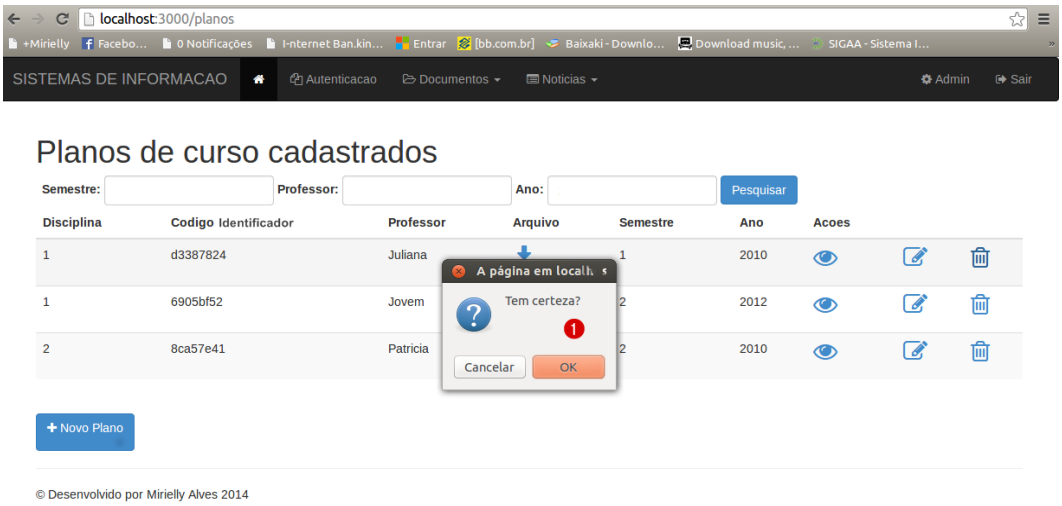


Figura 19 *– Mensagem de vericação se o usuário tem certeza que quer excluir determinado item.*

Página de Login

A página de *login* (Figuras 20 e 21 dispostas no *notebook e smartphone* respectivamente) é simples, o super usuário se autentica e com isso recebe permissões para executar ações adicionais no *site*, como: cadastrar, editar e excluir. Essa página é composta por apenas dois campos e um botão, para o usuário digitar o nome e a senha e acessar a página de administração do sistema. Ela segue o mesmo padrão das outras páginas, tanto em relação a formato de botões, cores etc. (critério 6 da seção 3.2.1 - Critérios Ergonômicos).

Para efetuar o *login* o usuário obrigatoriamente deve ser cadastrado no sistema, caso ele erre a senha ou nome a *interface* informa-lhe com uma mensagem simples e clara o possível

erro (Item 1 figuras 20 e 21). Com isso, busca-se orientá-lo através de *feedbacks* instantâneos. A fim de mantê-lo sempre informado do estado do sistema (critério 1, 9 e 5 da seção 3.2.1 - Heurísticas de Usabilidade e Critérios Ergonômicos respectivamente).

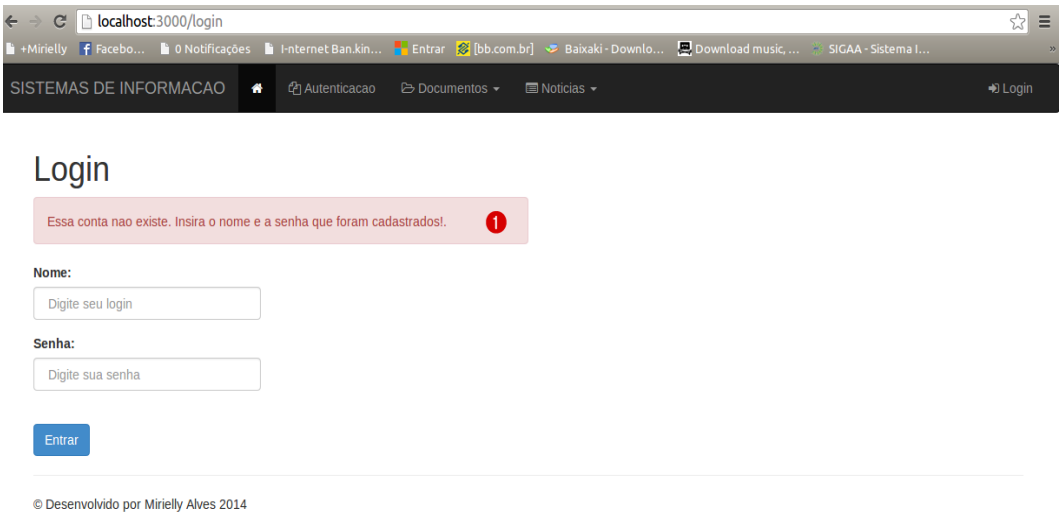


Figura 20 *– Tela de Login disposta no notebook.*

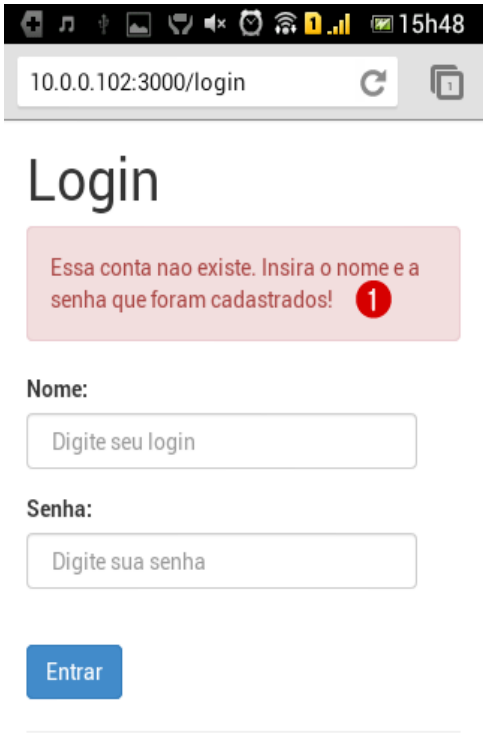


Figura 21 *– Tela de Login disposta no smartphone.*

Página do administrador

Através dessa página os super usuários gerenciam o sistema (Figuras 22 e 23 dispostas no *notebook e smartphone* respectivamente). Pois, nelas encontram-se os *links* para cadastrar,

exibir, editar e excluir disciplinas, planos de curso e usuários (Item 1 figuras 22 e 23). Possui uma *interface* simples que segue os padrões de cores e ícones do *site*. Além disso, apresenta três ícones cada um referenciando a uma função diferente. O boneco representa o usuário por padrão, o livro as disciplinas e o arquivo o plano de curso. Nota-se que são azuis, pois tratam-se de botões que *linkam* pra outras páginas (critério 7 da seção 3.2.1 - Critérios Ergonômicos). Com isso, busca-se atribuir aos ícones uma relação entre o objeto e sua referência na *interface* mantendo uma estabilidade gráfica comunicativa através de formatos, pois quando o usuário sabe onde, por padrão, estará ou significará certo botão, descrição etc, a transição de uma tela para outra se tornará mais previsível e amigável (critério 6 da seção 3.2.1 - Critérios Ergonômicos).

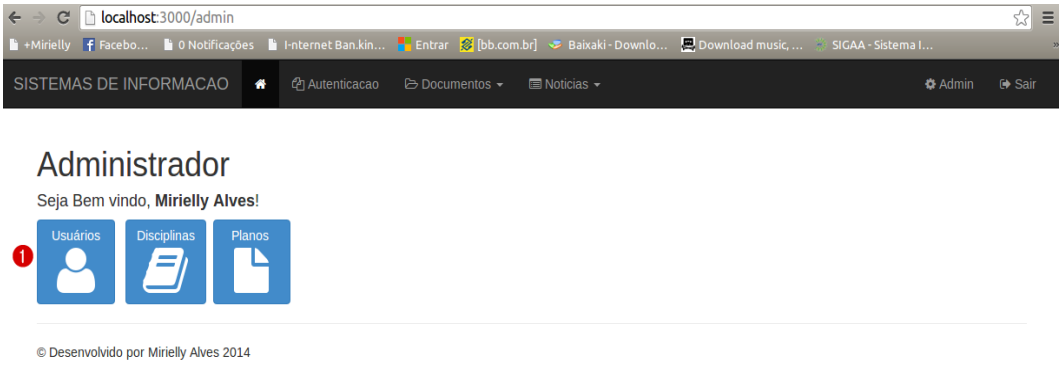


Figura 22 *– Tela do administrador disposta no notebook.*

Figura 23 *– Tela do administrador disposta no smartphone.*

Páginas de cadastros

Essas páginas apresentarão formulários onde o usuário preencherá os dados de acordo com o que estiver cadastrando, podendo ser uma disciplina, um plano de curso ou outro usuário. Essa parte diz respeito ao processamento das ações dos usuários feitas pelo sistema. O sistema executará apenas os comandos ordenados pelo usuário quando o mesmo quiser. Como exemplo, tem-se o cadastro de usuários e planos de curso (Figuras 24 e 25 dispostas no *notebook e smartphone* respectivamente), pois os dados são enviados apenas sob o comando do usuário, o sistema não salva sozinho. (critério 3 da seção 3.2.1 - Critérios Ergonômicos).

Além disso, o usuário a qualquer momento pode cancelar, voltar, salvar, ou excluir os dados cadastrados (Itens 1, 2 das figuras 24 e 25). Cabe ao sistema deixar o usuário no controle dos processos, dando-lhe liberdade nas suas ações perante o sistema. No entanto, somente as opções necessárias para cada situação devem ser disponibilizadas (critério 3 da seção 3.2.1 - Critérios Ergonômicos). Por fim o sistema também mostra as mensagens quando uma disciplinas é cadastrada com sucesso (critério 1 da seção 3.2.1 - Heurísticas de Usabilidade) dando ao usuário maior visibilidade do estado do sistema.

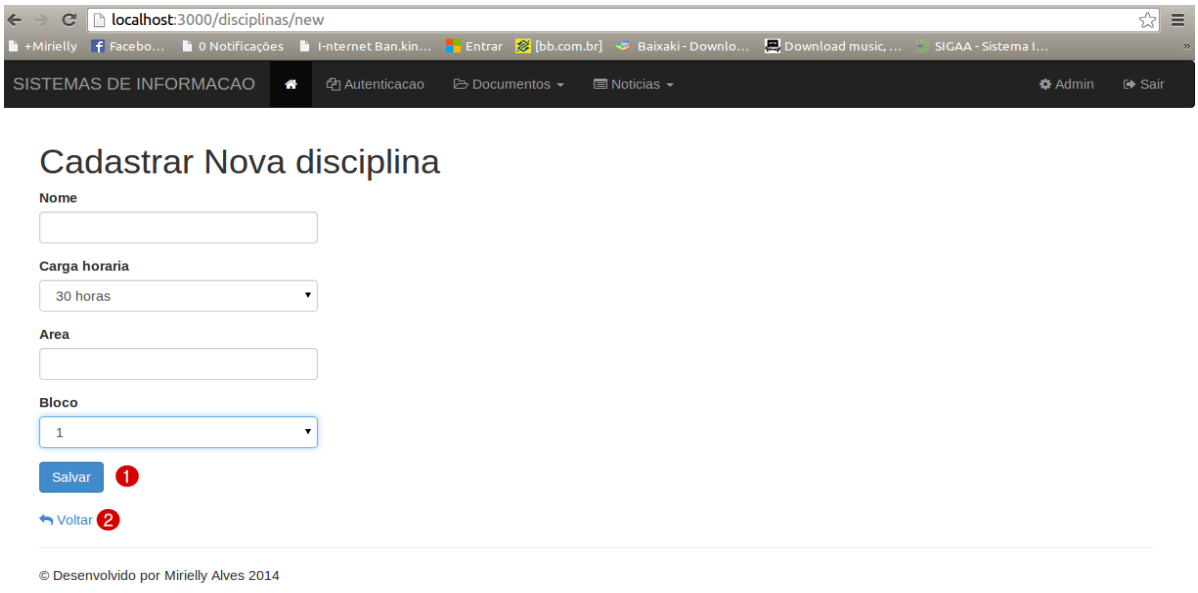


Figura 24 *– Cadastro de disciplinas disposta no notebook.*

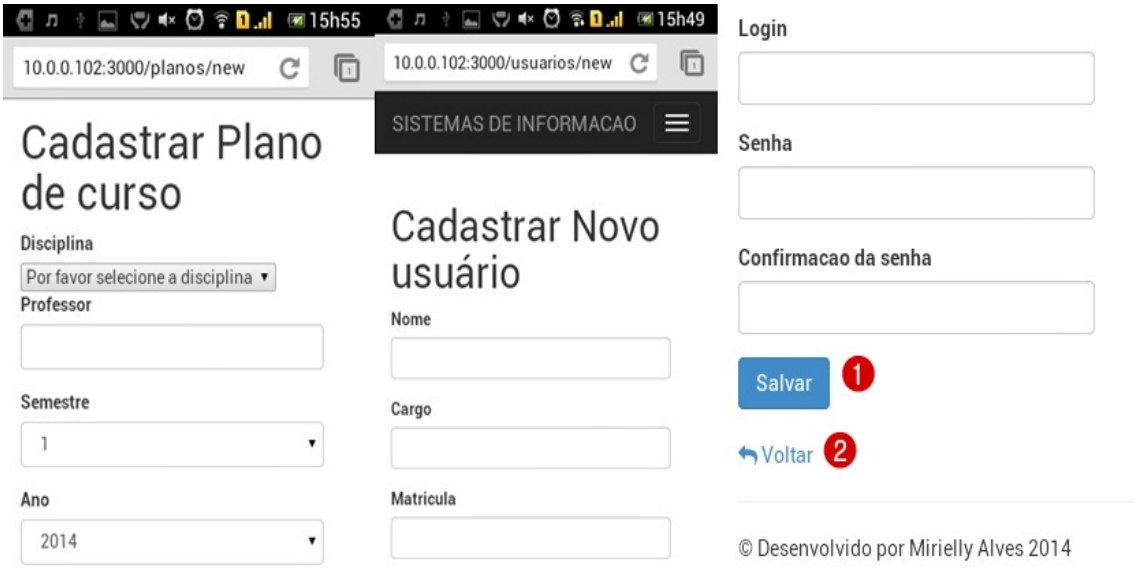


Figura 25 *– Cadastro de planos de curso e usuários dispostos no smartphone.*

Páginas de autenticação

Essa página apresenta um campo para *upload* de arquivo e um botão para enviá-lo (Figuras 26 e 27 dispostas no *notebook e smartphone* respectivamente). Através da mesma o usuário poderá verificar a integridade do conteúdo dos planos de curso que foram feitos *download* no *site*. Ela segue o mesmo padrão de cores das outras páginas (critério 6 da seção 3.2.1 - Critérios Ergonômicos). Além disso, atribui-se ao usuário o controle das ações de fazer *upload* e enviá-lo (Itens 1 e 2 figura 26 e 27). Uma vez que se específica, explicitamente, o controle das ações de entrada, as redundâncias e os erros são minimizados (critério 3 da seção 3.2.1 - Critérios Ergonômicos).

Outro fator importante é o *feedback* imediato que a *interface* oferece ao usuário (subcritério, do critério 1 e critério 1 da seção 3.2.1 - Critérios Ergonômicos e Heurísticas de Usabilidade respectivamente), pois se o arquivo que está sendo posto em verificação for íntegro a *interface* logo responderá ao mesmo com o *link* do arquivo e uma mensagem simples e clara que o mesmo foi encontrado e não foi modificado (Figuras 28 e 29 dispostas no *notebook e smartphone* respectivamente). Se não, a *interface* retornará uma mensagem notificando ao usuário que o arquivo não foi encontrado (Figuras 30 e 31 dispostas no *notebook e smartphone* respectivamente). Essas práticas mantém o usuário informado do atual estado do sistema.

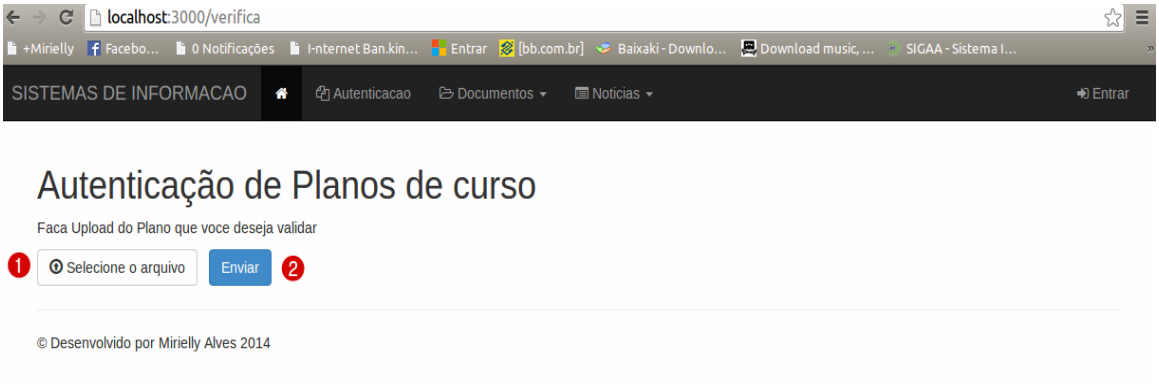


Figura 26 *– Autenticação dos planos de curso disposta no notebook.*

Figura 27 *– Autenticação dos planos de curso disposta no smartphone.*

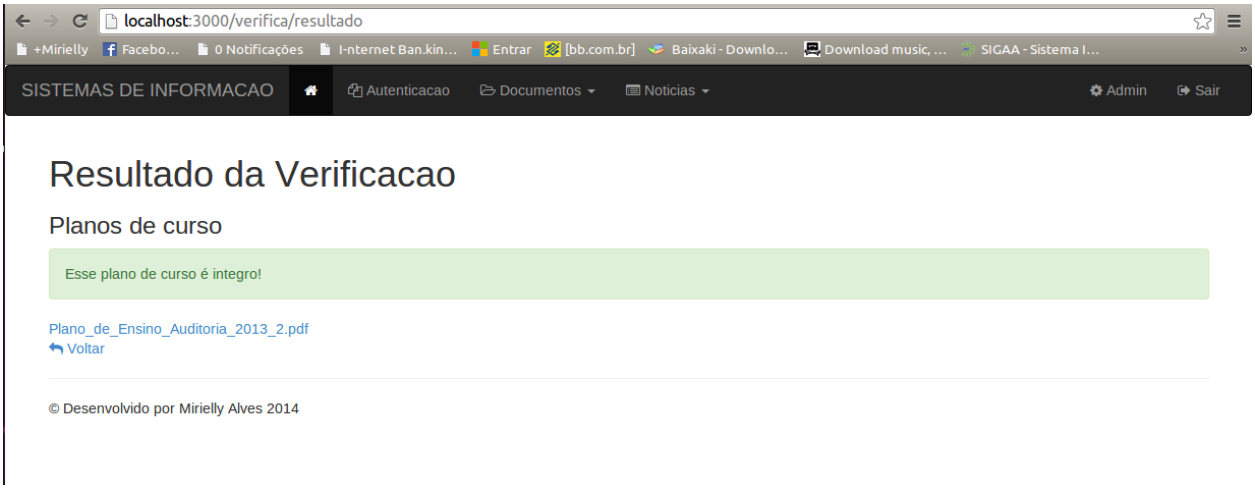


Figura 28 *– Resultado de uma verificação bem sucedida, disposta no notebook.*

Figura 29 *– Resultado de uma verificação bem sucedida, disposta no smartphone.*

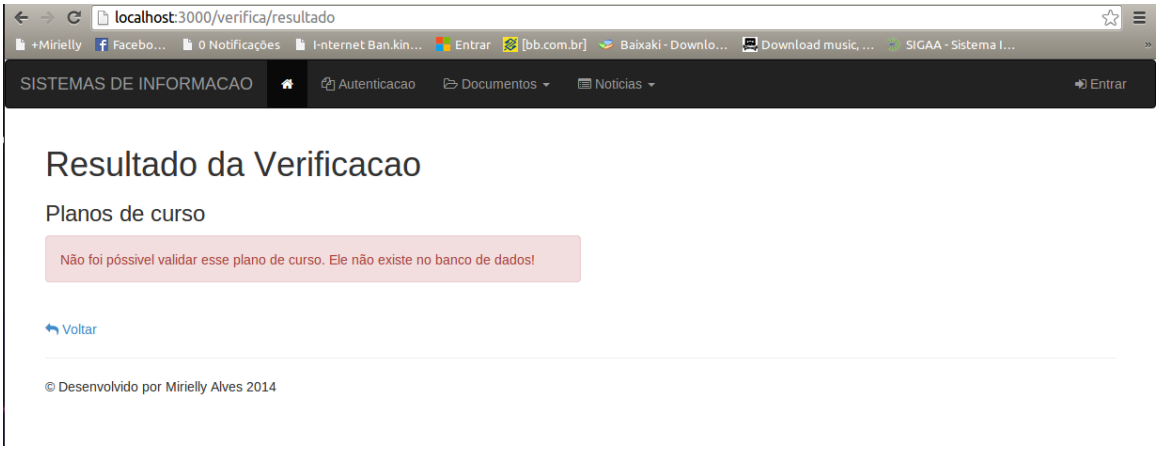


Figura 30 *– Resultado de uma verificação mal sucedida, disposta no notebook.*

Figura 31 *– Resultado de uma verificação mal sucedida, disposta no smartphone.*

Com isso, busca-se mostrar a importância das heurísticas de usabilidade e dos critérios ergonômicos sendo aplicados no desenvolvimento de uma *interface*. Portanto, a partir do momento que se leva em consideração as qualidades ergonômicas em que o usuário é o centro do projeto, evita-se o retrabalho por parte dos projetistas e tem-se como produto final uma *interface* simples que facilita e agiliza o tempo de execução das tarefas pelos usuários para que haja satisfação do mesmo.

3.3.2 Desenvolvimento da autenticação dos planos de curso

Esta seção descreve o funcionamento e a forma que foi implementado a autenticação dos planos de curso. O cadastro desses planos poderá ser feito pelos super usuários, que são os administradores do sistema. Estes possuem autorização para tal feito. Com isso, os planos estarão disponíveis para usuários comuns acessá-los e fazer *download* dos mesmos. Além disso, também estará disponível a verificação de integridade dos planos, caso uma terceira pessoa que tenha recebido o documento deseje autenticá-lo.

Descrição da implementação da autenticação dos planos de curso

Para a implementação da autenticação dos planos de curso, foi escolhido a técnica de *hash* (Capítulo 2, seção 2.2.2) que garante a integridade do conteúdo dos documentos. Visto que, o mesmo gera uma forma de assinatura única dos arquivos. Ou seja, qualquer alteração mínima no conteúdo original pode ser identificada através desse processo. No projeto foram criados alguns *Controllers*, que são classes que recebem uma ação das *Views* e executam algum tipo de lógica ligada a um ou mais modelos. Dentre eles encontram-se os *Controllers* de plano de curso e de verificação.

Controller do plano de curso

No *controller* dos planos de curso são definidos os métodos executados com os planos de curso. Que são eles: o `index, show, edit, create` etc. É dentro do método `create` (Figura 32) mais especificamente dentro da condição `if @plano.save` que começa a autenticação. Visto que, é nesse `if` que gera-se a assinatura *hash* do documento através do MD5 (Capítulo 2, seção 2.2.2) como mostra a figura 32 na linha 48. O *hash* gerado fica armazenado no campo *hash md5* criado na tabela do plano de curso, caso ele seja salvo no banco de dados da aplicação. Com isso, os planos estarão disponíveis para *download*.

planos_controller (1).rb

```
38        if @plano.save
39           pdf = Prawn::Document.new
40           pdf.move_down 700
41           pdf.text "Esse documento pode ser verificado em
     www.ufpi.br/planodecurso #{codigo}"
42           pdf.render_file "#{Rails.root}/app/stamp/#
     {codigo}.pdf"
43          require 'posix/spawn'
44          ::POSIX::Spawn::Child.new "pdftk", "#{Rails.root}/
     public/system/planos/arquivos/000/000/#{@plano.id.to_s.rjust
     (3,'0')}/original/#{filename}", "stamp", "#{Rails.root}/app/
     stamp/#{codigo}.pdf", "output", "#{Rails.root}/public/system/
     planos/arquivos/000/000/#{@plano.id.to_s.rjust(3,'0')}/
     original/copy-#{filename}"
45          ::POSIX::Spawn::Child.new "rm", "#{Rails.root}/public/
     system/planos/arquivos/000/000/#{@plano.id.to_s.rjust(3,'0')}/
     original/#{filename}"
46 ::POSIX::Spawn::Child.new "rm", "#{Rails.root}/app/stamp/#
     {codigo}.pdf"
47          ::POSIX::Spawn::Child.new "mv", "#{Rails.root}/public/
     system/planos/arquivos/000/000/#{@plano.id.to_s.rjust(3,'0')}/
     original/copy-#{filename}", "#{Rails.root}/public/system/
     planos/arquivos/000/000/#{@plano.id.to_s.rjust(3,'0')}/
     original/#{filename}"
48          @plano.codigo_md5 = Digest::MD5.hexdigest(File.read("#
     {Rails.root}/public/system/planos/arquivos/000/000/#
     {@plano.id.to_s.rjust(3,'0')}/original/#{filename}"))
49          @plano.save
50          format.html { redirect_to @plano, notice: 'Plano was
     successfully created.' }
51        else
52          format.html { render action: 'new' }
53          format.json { render json: @plano.errors,
     status: :unprocessable_entity }
54        end
55      end
56    end
```

Figura 32 *– Controller de plano de curso, método create.*

Controller de vericação

A verificação da integridade do plano de curso acontece dentro do *controller* de verificação (Figura 33). Com isso, após o usuário fazer *download* do arquivo o mesmo pode autenticá-lo no próprio *site*. Primeiramente, para dar início a autenticação deve-se fazer *upload* do arquivo que foi gerado pelo *site*.

Além disso, utiliza-se o número da *session* para atribuir o nome do arquivo que foi feito *upload* no servidor (Figura 33 linha 9). Isso garante que duas ou mais pessoas se estiverem fazendo a verificação ao mesmo tempo, os arquivos não sejam sobrescritos. Depois do *upload*, é feito um cálculo *hash* desse documento (Figura 33 linha 13), e gera-se uma assinatura *hash* do mesmo. Posteriormente, uma busca no banco de dados é feita através do campo *hash* (Figura 33 linha 15). É nesse momento a autenticação do documento ocorre, pois, se for encontrado no

banco de dados uma assinatura *hash* igual ao do arquivo que foi feito *upload*, isso significa que o documento verificado é íntegro, ou seja, o mesmo não teve seu conteúdo alterado. Se não, o sistema apresentará uma mensagem de erro explicitando que o documento verificado não é íntegro e não foi encontrado.

verificacao_controller.rb

Fechar documento

```
1  require 'digest/md
2  class VerificacaoController < ApplicationController
3    skip_before_filter :verify_authenticity_token
4    skip_before_action :authorize
5    def index
6    end
7
8    def resultado
9      name = "#{request.session_options[:id]}.pdf"
10     directory = "public/images/upload/"
11     path = File.join(directory, name)
12     File.open(path, "wb") { |f| f.write(params[:file].read) }
13     codigo = Digest::MD5.hexdigest(File.read(path))
14     ::POSIX::Spawn::Child.new "rm", path
15     @plano = Plano.find_by_codigo_md5(codigo)
16   end
17
18 end
```

Figura 33 *– Controller de verificação.*

Por fim, o documento que foi posto em verificação é deletado do banco de dados, pois após a verificação ele não será mais utilizado. Com isso, não tem-se a necessidade de mantê-lo salvo (Figura 33 linha 14).

3.4 Resultados

Nessa seção serão descritos através de exemplos fictícios todas as possibilidades de resultados obtidos através dos testes feitos com a aplicação.

3.4.1 Testes

Para demonstrar a eficiência dessa aplicação foram realizados testes de vários tipos tais como: testes de cadastro, de busca e de verificação de validade dos documentos no *notebook* e no *smartphone* para explicitar a eficiência do *site*, do serviço de autenticação e a adaptabilidade do mesmo

Testes de cadastro

Para fazer o teste do cadastro partiu-se do pressuposto que existem dois tipos de autorização para uso do *site*. A primeira está relacionada à possibilidade de efetuar cadastros de planos de curso, disciplinas e usuários. Nesse caso, o super usuário deverá cadastrar outros usuários previamente para estes adquirir tais permissões. Já a segunda está relacionado aos usuários que podem apenas fazer consultas e *downloads* dos planos de curso, que não precisam estar cadastrados.

Testes com pessoas não autorizadas

Supõe-se que certo dia um funcionário da coordenação do curso de Sistemas de Informação abra o *site* e queira efetuar o cadastro de um plano de curso (Item 6 das figuras 17 e 18). Porém, este não é cadastrado no *site*. Nesse caso, o mesmo não completará suas atividades e o sistema lhe redirecionará para uma página de *login*, e ao tentar se autentificar a *interface* lhe mostrará uma notificação de que ele não está cadastrado (Figura 20 e 21), pois o coordenador do curso deveria tê-lo cadastrado previamente.

Testes com pessoas autorizadas

Outro caso que poderá acontecer e desse mesmo funcionário (seção 3.5.1.1.1) já ter sido cadastrado pelo coordenador e então ter permissões de administrador (Figuras 22 e 23 dispostas no *notebook e smartphone* respectivamente) e efetuar o cadastro do plano de curso, da disciplina ou de algum usuário que desejar como mostram as figuras 17,18,34, 35, 36 e 37 dispostas no *notebook e smartphone* respectivamente.

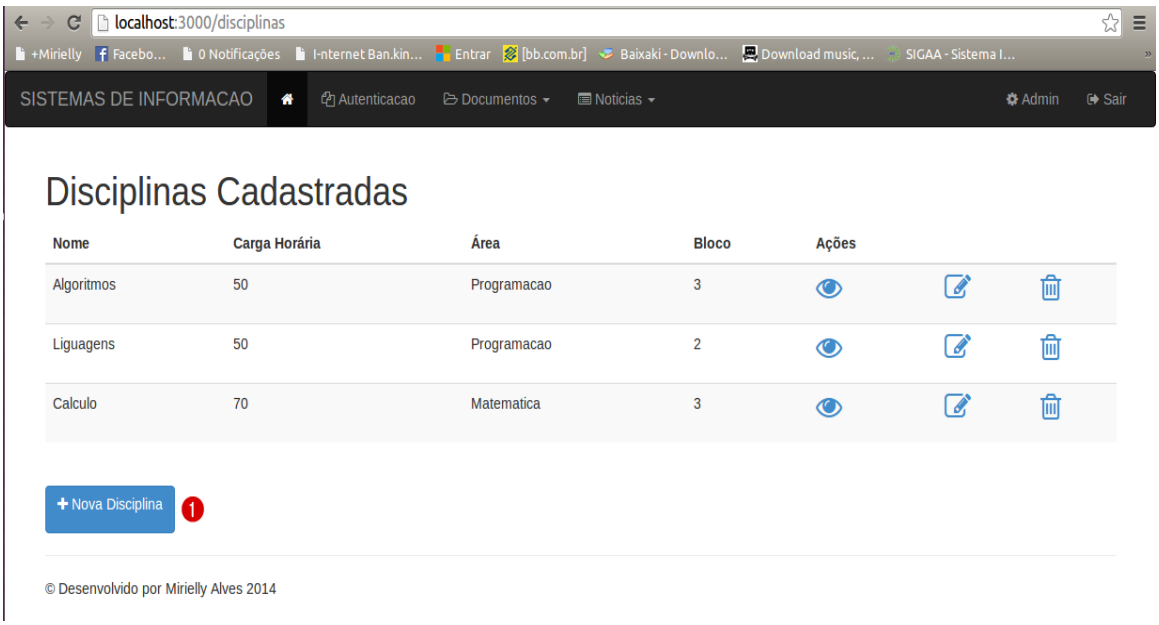


Figura 34 *– Disciplinas Cadastradas dispostas no notebook.*

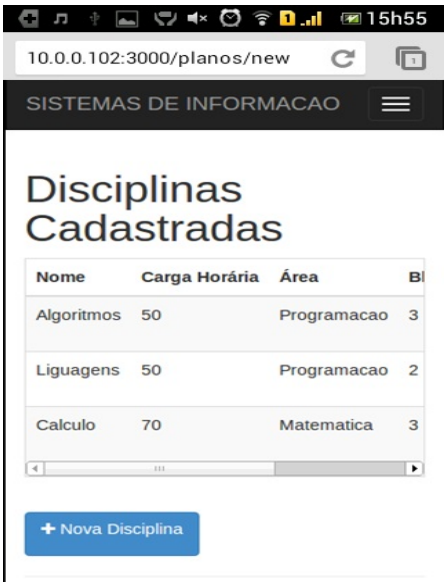


Figura 35 *– Disciplinas Cadastradas dispostas no smartphone.*

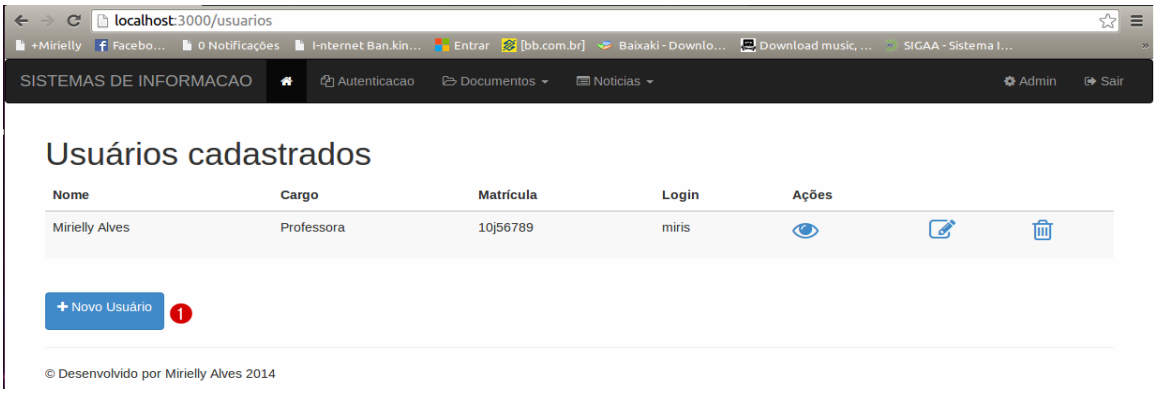


Figura 36 *– Usuários cadastrados dispostos no notebook.*

Figura 37 *– Usuários cadatrados dispostos no smartphone.*

Testes de Busca

Esse serviço de busca é voltado para os discentes. E suponha-se que uma aluna que estuda na universidade X situada em Picos-PI e quer se transferir ou é aprovada para outro curso na universidade Y em Fortaleza-Ce. Na nova grade curricular existem algumas disciplinas que essa aluna já cursou e que deseja pedir dispensa. Com isso, faz-se necessário que a mesma apresente o histórico que consta as disciplinas que cursou e os planos de curso referente às que desejar dispensar.

Deste modo, essa aluna poderá acessar o *site* direcionar-se para página de planos de curso (Figuras 17 e 18) e então o sistema disponibilizará para ela buscas através do ano que o documento foi emitido, do semestre que foi cursada a disciplina ou do professor que lecionou a mesma (Item 5 das figuras 17 e 18). Os resultados serão mostrados a seguir. E para melhor compreensão dos testes as figuras 17 e 18 mostram os planos de curso cadastrados no *site* com as quais foram feitas as buscas.

Testes de Busca por semestre

Suponha-se que essa aluna não se lembre do ano que cursou a disciplina e nem do nome do professor que a lecionou. Mas lembra do semestre. Com isso, ela efetua uma pesquisa através do semestre cursado e então o sistema apresentará os planos de curso referente a tal semestre (Figuras 38 e 39 dispostas no *notebook e smartphone* respectivamente). Isso, agiliza a busca dessa aluna, pois filtram-se os resultados fornecendo apenas os planos cadastrados no semestre pesquisado por ela.

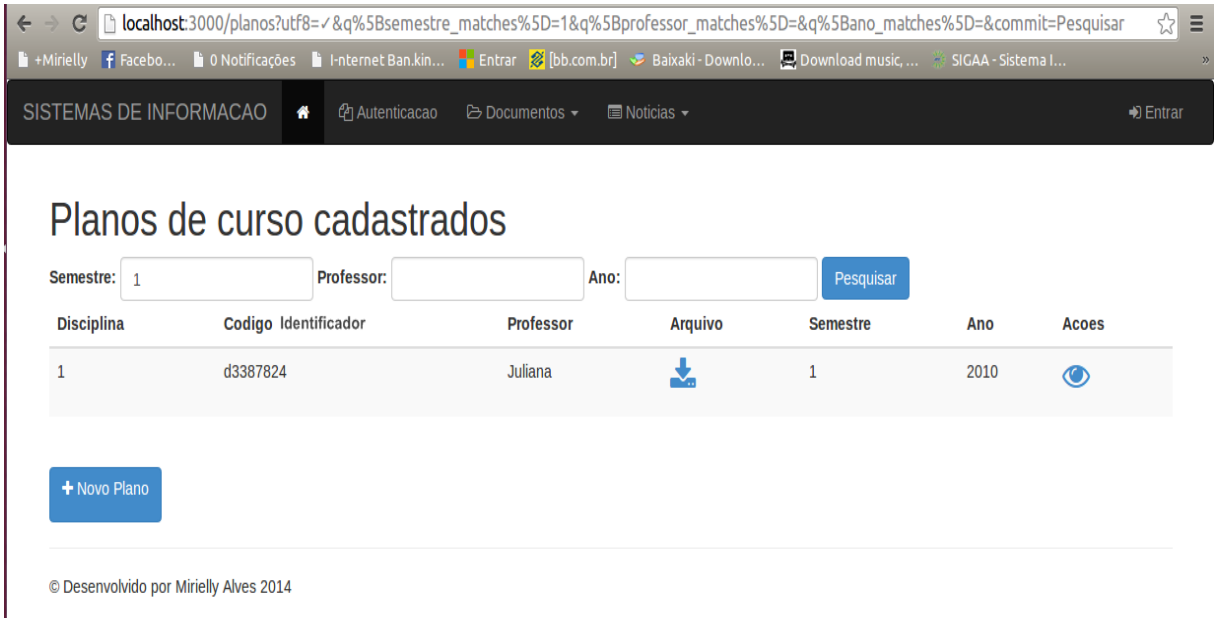


***Figura 38** – Pesquisa feita através do 1º semestre, disposta no notebook.*

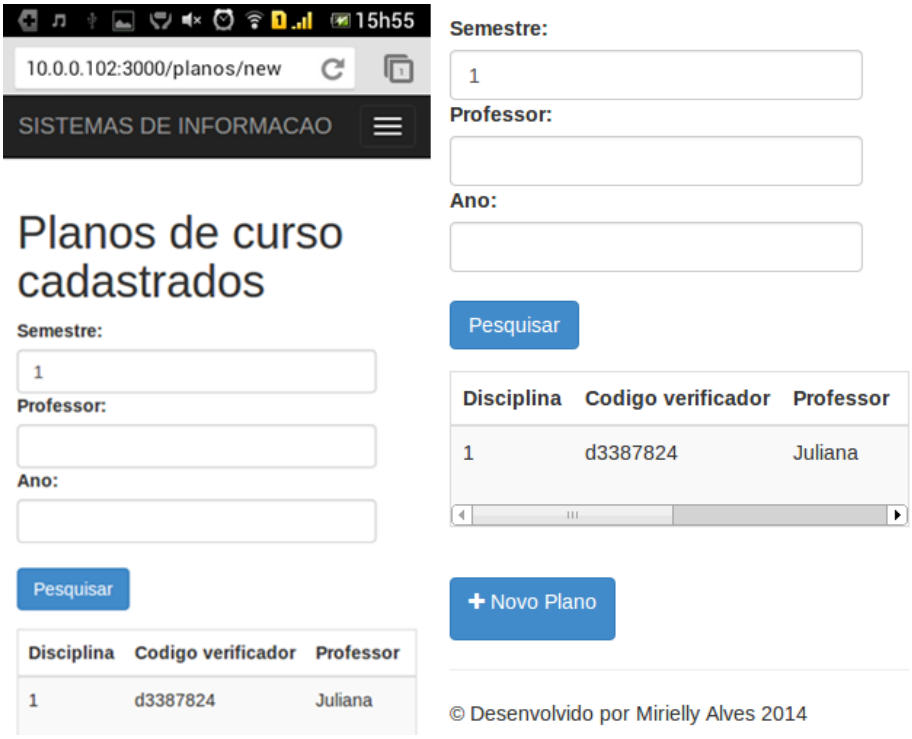


Figura 39 *– Pesquisa feita através do 1º semestre, disposta no smartphone.*

Testes de Busca por ano

Nesse caso acontece o mesmo da seção 3.5.1.2.1, porém a aluna só lembra-se do ano que cursou a disciplina. Com isso, ela efetua uma pesquisa através do ano cursado e então encontrará o plano de curso desejado como mostra às figuras 40 e 41 dispostas no *notebook e smartphone* respectivamente.

Figura 40 *– Pesquisa feita através ano, disposta no notebook.*

Figura 41 *– Pesquisa feita através ano, disposta no smartphone.*

Testes de Busca por professor

Nesse caso acontece o mesmo da seção 3.5.1.2.1, porém a aluna só lembra-se do professor que lecionou a disciplina. Com isso, ela efetua uma pesquisa através do nome do professor e então encontrará o plano de curso desejado como mostra às figuras 42 e 43 dispostas no *notebook e smartphone* respectivamente. Esse processo garante a praticidade na hora de encontrar os planos de curso. Devido ao sistema sempre filtrar de acordo com o que o usuário estabelecer.

Figura 42 *– Pesquisa feita através do nome do professor, disposta no notebook.*

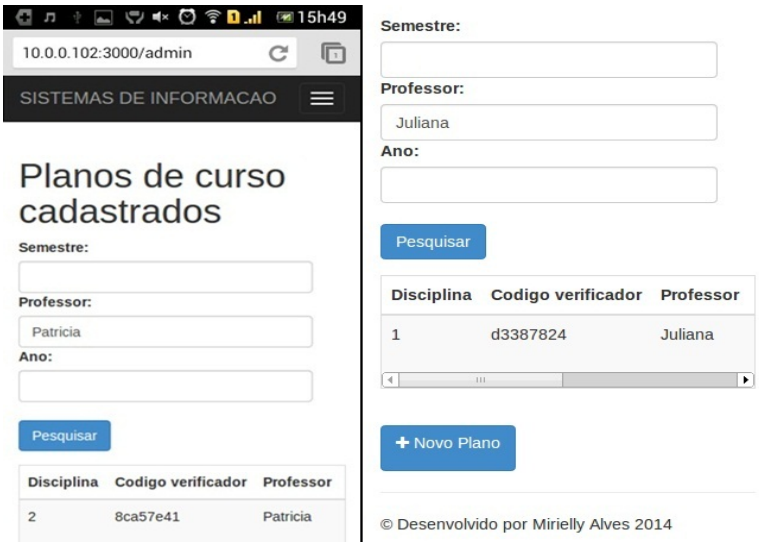


Figura 43 *– Pesquisa feita através do nome do professor, disposta no smartphone.*

Testes de Busca por semestre e professor

Nesse caso a aluna filtra ainda mais a busca, pois além de lembrar-se do semestre ela lembra-se do professor. Com isso, alcançam-se resultados ainda mais rápidos, pois quanto mais informações pra restringir a busca melhor. Essa filtragem é mostrada nas figuras 44 e 45 dispostas no *notebook e smartphone* respectivamente.

Figura 44 *– Pesquisa feita através do semestre e professor, disposta no notebook.*

Figura 45 *– Pesquisa feita através do semestre e professor, disposta no smartphone.*

Testes de Busca por semestre, ano e professor

Por fim, a última busca disponível. Nesse caso a aluna filtra ainda mais, pois ela lembra-se de três informações que são elas: o ano, o semestre e o professor. Com isso, alcançam-se resultados ainda mais rápidos, pois quanto maior o número de informações pra restringir a busca mais rápida o resultado será alcançado. Essa busca e mostrada nas figuras 46 e 47 dispostas no *notebook e smartphone* respectivamente.

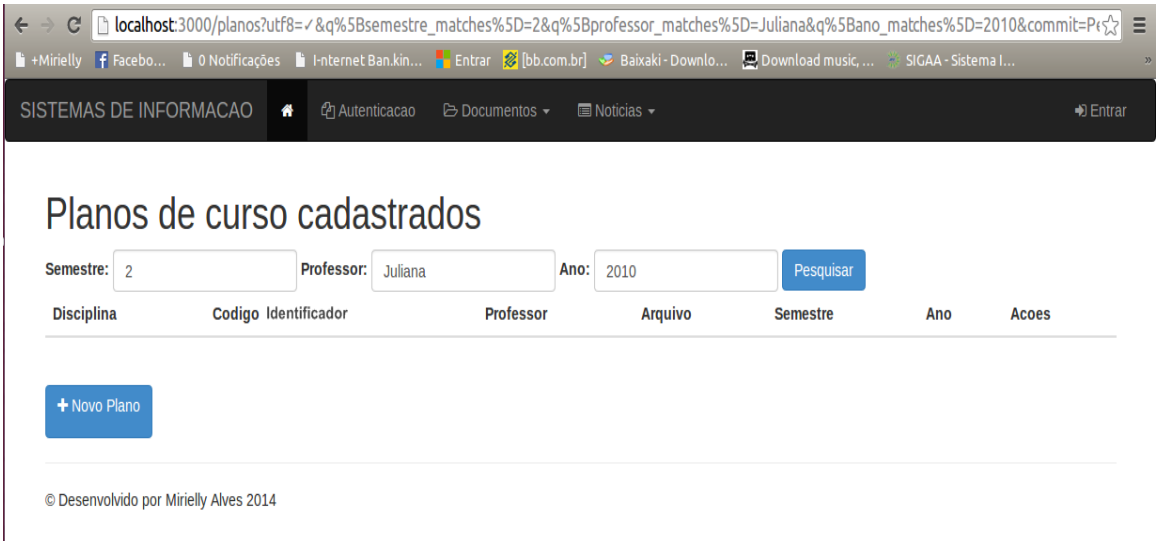


Figura 46 *– Pesquisa feita através do semestre, ano e professor, disposta no notebook.*

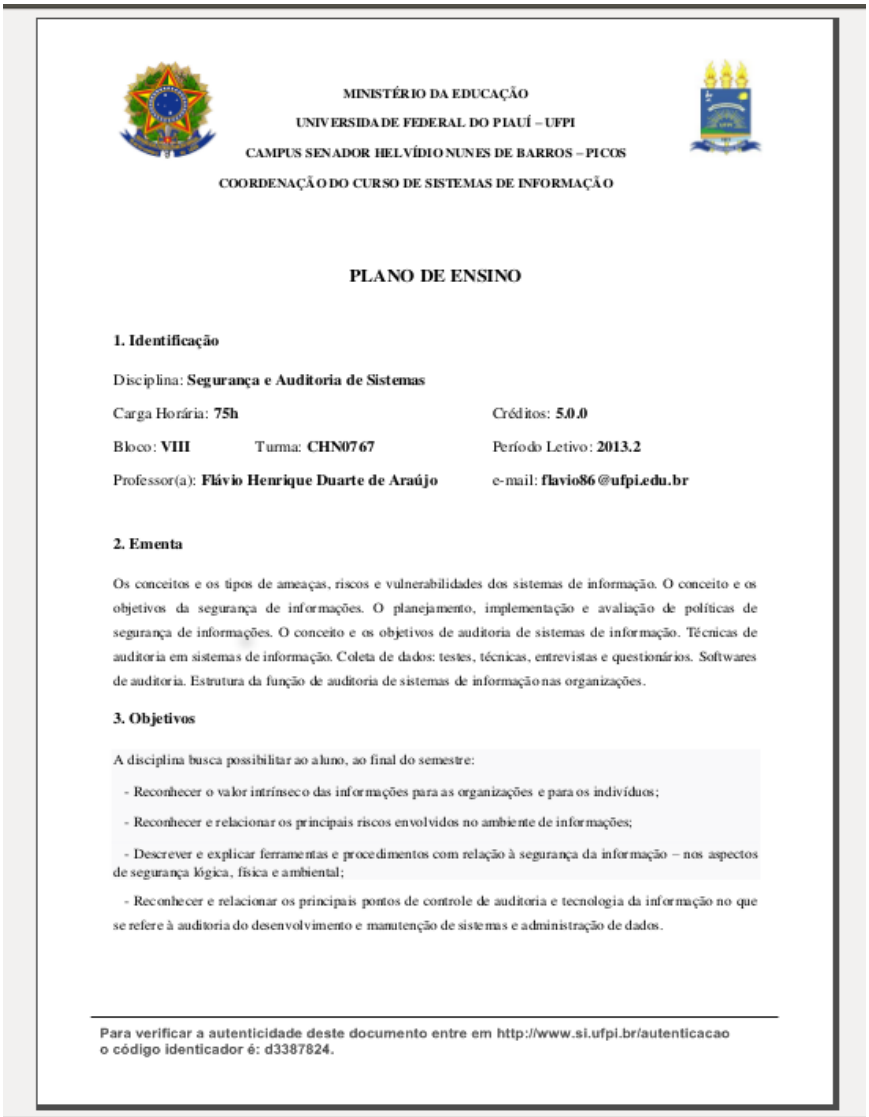
Figura 47 *– Pesquisa feita através do semestre, ano e professor, disposta no smartphone.*

Testes de Verificação de integridade

Esses testes estão relacionados à autenticação dos documentos digitais e garantia ou não de sua integridade. Serão mostradas em seguida algumas das possibilidades de verificação de integridade por partes das instituições de ensino que se deparará com casos de transferências de alunos ou casos de alunos que já cursaram algumas disciplinas em outra instituição que desejem dispensá-las.

Teste com documento íntegro

Suponha-se que acontece o mesmo caso da seção 3.5.1.2 (Testes de Busca). Onde a aluna precisará dos planos de curso para dispensar algumas matérias cursadas. Com isso, ela acessará o *site* desenvolvido e efetuará o *download* dos planos de curso que precisar (figuras 17 e 18). A mesma não os modificará, mantendo-os íntegro (Figura 48). Posteriormente, a instituição que recebeu os planos sujeitou-lhes ao serviço de autenticação de documentos do *site*, a fim de testificar se os mesmos seriam encontrados e validados (Figura 49 e 50 dispostas no *notebook e smartphone* respectivamente). O resultado é mostrado na figura 28 e 29 e nota-se que o documento foi encontrado e valido com sucesso , ou seja, o sistema notificou que o mesmo é íntegro e pode ser usado como prova que tal aluna cursou a disciplina que a mesma quer dispensar.



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PIAUÍ – UFPI
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS – PICOS
COORDENAÇÃO DO CURSO DE SISTEMAS DE INFORMAÇÃO

PLANO DE ENSINO

1. Identificação

Disciplina: **Segurança e Auditoria de Sistemas**

Carga Horária: **75h** Créditos: **5.0.0**

Bloco: **VIII** Turma: **CHN0767** Período Letivo: **2013.2**

Professor(a): **Flávio Henrique Duarte de Araújo** e-mail: **flavio86@ufpi.edu.br**

2. Ementa

Os conceitos e os tipos de ameaças, riscos e vulnerabilidades dos sistemas de informação. O conceito e os objetivos da segurança de informações. O planejamento, implementação e avaliação de políticas de segurança de informações. O conceito e os objetivos de auditoria de sistemas de informação. Técnicas de auditoria em sistemas de informação. Coleta de dados: testes, técnicas, entrevistas e questionários. Softwares de auditoria. Estrutura da função de auditoria de sistemas de informação nas organizações.

3. Objetivos

A disciplina busca possibilitar ao aluno, ao final do semestre:

- Reconhecer o valor intrínseco das informações para as organizações e para os indivíduos;
- Reconhecer e relacionar os principais riscos envolvidos no ambiente de informações;
- Descrever e explicar ferramentas e procedimentos com relação à segurança da informação – nos aspectos de segurança lógica, física e ambiental;
- Reconhecer e relacionar os principais pontos de controle de auditoria e tecnologia da informação no que se refere à auditoria do desenvolvimento e manutenção de sistemas e administração de dados.

Para verificar a autenticidade deste documento entre em http://www.si.ufpi.br/autenticacao o código identicador é: d3387824.

Figura 48 *– Pdf original baixado do site.*

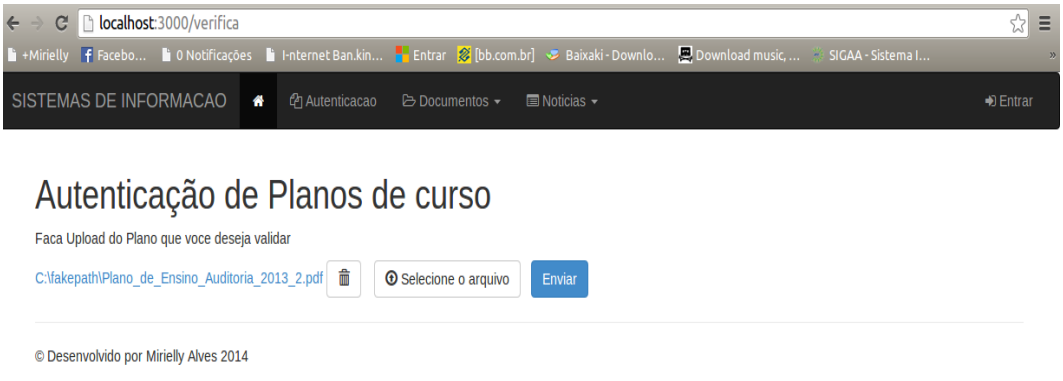


Figura 49 *– Verificação de integridade, disposta no notebook.*

Figura 50 *– Verificação de integridade, disposta no smartphone.*

Teste com documento modificado

Suponha-se que acontece o mesmo caso da caso da seção 3.5.1.2 (Testes de Busca). Porém, ao contrário do teste feito com documento íntegro. Essa aluna tentou usar de esperteza e utilizou artifícios ilegais para modificar o conteúdo do plano de curso, a fim de comprovar que cursou tal disciplina. E então a mesma editou o plano de curso que foi feito *download* no *site*, em um editor de pdf *online* como esse: www.pdfescape.com, mudou a carga horaria da disciplina de 70 horas para 90 horas (Figura 51) e por fim, enviou o documento para instituição que quer dispensar algumas matérias.

1. Identificação

Disciplina: **Segurança e Auditoria de Sistemas**

Carga Horária: **90 horas**

Bloco: **VIII**

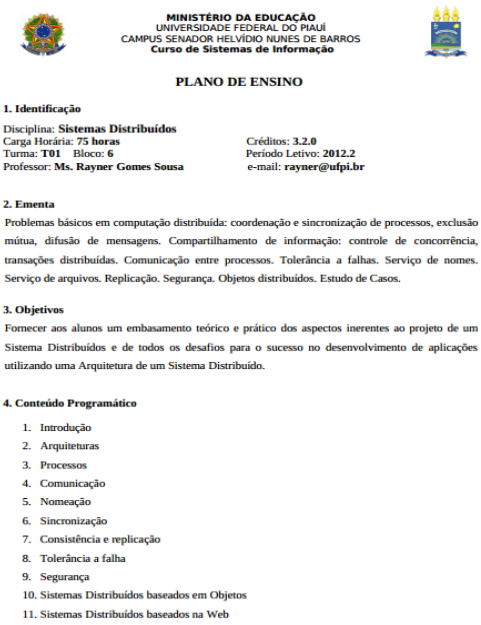
Professor(a): **Flávio Henrique Duarte de Araújo**

Figura 51 *– Pdf modicado através e editor de texto online.*

Ao receber o plano de curso a instituição de destino sujeitou o documento ao sistema de validação do *site* (Figura 49 e 50). Feito isso, obteve-se que o plano de curso avaliado não foi encontrado, logo a instituição se deu conta que o mesmo não é íntegro e possivelmente teve seu conteúdo alterado (Figura 30 e 31). Isso prova a eficiência, a agilidade e a segurança desse sistema de autenticação. E vale ressaltar que a aluna fez uma mudança mínima no documento, de apenas um número. Mas, mesmo assim, ela não passará despercebida, pois a técnica utilizada através da função *hash* consegue garantir se o conteúdo do documento é íntegro ou não.

Teste com documentos não cadastrados

Nesse caso, buscou-se mostrar a possibilidade de um aluno enviar a instituição que deseja se transferir um plano de curso não oriundo do *site* desenvolvido, ou seja, que não foi cadastrado por ninguém no mesmo (Figura 52). Sabe-se que esse plano de curso não foi cadastrado no sistema, pois, a disciplina do mesmo não consta na lista de disciplinas cadastradas (Figuras 34 e 35) e em seu rodapé não possui o número de verificação impresso pelo sistema em cada plano de curso gerado por ele. Sendo assim, como a instituição não sabe que o plano não foi cadastrado no sistema, ela efetuará a verificação da mesma forma (Figura 49 e 50). Porém, o documento não será encontrado como mostra as figuras 30 e 31, pois o mesmo não foi previamente cadastrado. O aluno por sua vez, se não utilizou-se de má fé para com a instituição, sabendo do acontecido, poderá procurar a coordenação do seu curso e solicitar para que alguém autorizado cadastre os planos que ele necessita.



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO PIAUÍ
CAMPUS SENADOR HELVÍDIO NUNES DE BARROS
Curso de Sistemas de Informação

PLANO DE ENSINO

1. Identificação

Disciplina: **Sistemas Distribuídos**
Carga Horária: **75 horas** Créditos: **3.2.0**
Turma: **T01** Bloco: **6** Período Letivo: **2012.2**
Professor: **Ms. Rayner Gomes Sousa** e-mail: **rayner@ufpi.br**

2. Ementa

Problemas básicos em computação distribuída: coordenação e sincronização de processos, exclusão mútua, difusão de mensagens. Compartilhamento de informação: controle de concorrência, transações distribuídas. Comunicação entre processos. Tolerância a falhas. Serviço de nomes. Serviço de arquivos. Replicação. Segurança. Objetos distribuídos. Estudo de Casos.

3. Objetivos

Fornecer aos alunos um embasamento teórico e prático dos aspectos inerentes ao projeto de um Sistema Distribuídos e de todos os desafios para o sucesso no desenvolvimento de aplicações utilizando uma Arquitetura de um Sistema Distribuído.

4. Conteúdo Programático

1. Introdução
2. Arquiteturas
3. Processos
4. Comunicação
5. Nomeação
6. Sincronização
7. Consistência e replicação
8. Tolerância a falha
9. Segurança
10. Sistemas Distribuídos baseados em Objetos
11. Sistemas Distribuídos baseados na Web

Figura 52 *– Plano de curso não cadastrado no sistema.*

Por fim, através de todos esses testes feitos testifica-se a importância desse estudo, devido os alunos não preocuparem-se com locomoção, economizarem tempo, dinheiro e agilizarem o processo de obtenção dos planos, pois tudo é feito por meio da *Internet*. Além de poder acessar de qualquer dispositivo móvel, tornando o serviço muito mais cômodo ao usuário. E os coordenadores que recebem esses documentos também obtêm vantagens, pois é muito mais prático receber um plano de curso por *e-mail*, sujeitá-lo a verificação de autenticidade em um sistema que utilize técnicas de autenticação como o *hash*, garantindo-lhe a integridade do mesmo. Do que, ler diversas folhas de um plano de curso, a fim de testificar que nenhum dado foi alterado. Além de que, esse tipo de serviço é ecologicamente correto, pois preserva o meio ambiente não desperdiçando folhas de papel.

4 Conclusões e Trabalhos Futuros

O sistema utilizado pela Universidade Federal do Piauí, SIGAA não disponibiliza os planos de curso *online* e autenticados. Esse processo é feito de modo presencial. Por isso, os alunos que necessitarem de tais documentos recorrem à coordenação de seu curso, que por sua vez disponibilizam os planos assinados pelo coordenador, constando o carimbo da instituição, sem nenhum uso de meios eletrônicos. Diante disso, faz-se necessário que haja uma maneira mais simples e ágil de disponibilização desses documentos autenticados. É nesse contexto que surgiu a ideia do sistema de autenticação de documentos digitais para *sites* adaptáveis com *design* responsivo.

Para dar início ao estudo foram feitas várias pesquisas sobre autenticação de documentos digitais. Com isso, buscou-se tomar conhecimento das técnicas de autenticação existentes, antes de escolher a que mais se adequasse a essa situação. As técnicas estudadas foram: criptografia simétrica e assimétrica, função *hash*, assinatura digital e certificação digital. Dentre essas, a que mais se adequou ao presente estudo foi à função *hash*, pois o código gerado é único e diferente para cada documento, não há necessidade de utilização de chaves como na criptografia e assinatura digital e uma simples alteração no conteúdo do documento original produz um *hash* completamente diferente. Com relação a isso, pode-se observar através dos exemplos apresentados no capitulo 3 várias situações, e em todas essas a utilização da função *hash* garantiu integridade e autenticidade dos documentos em questão.

Além disso, também foram pesquisadas quais ferramentas seriam apropriadas para o desenvolvimento dessa aplicação. Deste modo, foi escolhido o *Framework Ruby on Rails*, por ser voltado para desenvolvimento de aplicações *web* de forma ágil e também por ter integração com MD5 que é o algoritmo que gera a função *hash*. Além de também se integrar com *Framework de Front-End Bootstrap*, que é responsável pela adaptabilidade do *site*, através do *design* responsivo.

Ademais, com o decorrer do desenvolvimento tiveram algumas dificuldades em relação à manipulação de pdfs, por questões de não encontrar as configurações adequadas e por falta de material em português que fale onde e como configurá-las. Outra dificuldade foi à questão de aprender uma nova linguagem de programação, até então nunca vista e entender como a

arquitetura do *Rails* funciona, em relação as suas camadas e a comunicação entre elas. No entanto, com o decorrer do tempo e através de muitas pesquisas e estudos essas dificuldades foram sanadas.

Outros fatores importantes que foram abordados nesse trabalho são a usabilidade e responsividade para os dispositivos móveis, eles irão proporcionar um maior uso da aplicação e facilitar para os alunos e as instituições tanto para a que cadastra os planos de curso, como para a que recebe os planos, possibilitando assim, o acesso aos mesmos. Além disso, também vale ressaltar que não foram feitos testes com usuários reais, como funcionários e alunos, esse tipo de teste ainda precisa ser feito, além de verificar a usabilidade do mesmo, para investigar a necessidade de treinamento do mesmo.

Em relação a trabalhos futuros têm-se novos objetivos que são a criação de serviços como: disponibilizar um sistema para elaborar horários conforme a preferência dos professores e adequar de acordo com a necessidade da maioria dos alunos. Além de registrar o nome dos usuários que realizarem ações sobre o sistema tais como: cadastros, exclusões, edições, etc. A fim de, garantir o não repúdio, ou seja, garantir que o autor não negue ter feito tais ações.

Referências

AKITA, F. *Repensando a WEB com Rails.* [S.l.: s.n.], 2006.

ALECRIM, E. Entendendo a certificação digital. abr. 2011. Disponível em: <http://www.infowester.com/assincertdigital.php>. Acesso em: 14 Fev. 2014.

ALTERMANN, D. Design Responsivo: Entenda o que é e a técnica e como ela funciona. 2012. Disponível em: <http://www.midiatismo.com.br/o-mobile/design-responsivo-entenda-o-que-e-a-tecnica-e-como-ela-fun ciona>.

ARAUJO, Wagner;; VIEIRA, Renato. Assinatura de documentos eletrônicos utilizando cartificados digitais. *Biblionline*, v. 8, n. esp, p. 290–302, 2012. Disponível em: <http://periodicos.ufpb.br/ojs2/index.php/biblio/article/view/14204/8109>. Acesso em: 10 Dez. 2013.

BASTIEN, C.; SCAPIN, D. *Ergonomic Criteria for the Evaluation of Human-Computer Interfaces.* [S.l.: s.n.], 1993.

BIAZOTTI, R. O que é design responsivo – Seo Google. 2013. Disponível em: <http://www.rbmedia.com.br/artigos/o-que-e-design-responsivo-seo-google>.

BRASIL, Ângela Bittencourt. Assinatura digital não é assinatura formal. 2000. Disponível em: <http://www.egov.ufsc.br/portal/sites/default/files/anexos/3002-2996-1-PB.htm>.

BURBECK, S. Applications Programming in model-view-controller(mvc). 1992. Acesso em: 20 Jul. 2013.

Apostila curso: Desenvolvimento Ágil para Web com Ruby on Rails. 2013. Disponível em: <http://www.caelum.com.br/apostila-ruby-on-rails>.

CARLSON, L; RICHARDSON, L. Ruby Cookbook. O'Reilly Media, 2006.

CERTISIGN. Por dentro da Certificação Digital. 2012. Disponível em: <http://www.certisign.com.br/certificacao-digital/por-dentro-da-certificacao-digital>.

CHRISTOL, J. *Les logiciels, un travail pour l'ergonome? Les Cahiers Technologie.* [S.l.]: Emploi, Travail, 1987.

CUSTÓDIO, F. Ricardo; DIAS, Júlio S.; ROLT, Carlos R. Assinatura Confiável de Documentos Eletrônicos. 2009.

CYBIS, W. Ergonomia e Usabilidade - Conhecimentos, Métodos e Aplicações. 2010.

FRIEDMAN, S. IronRuby Unleashed: An Interview with Shay Friedman. abr. 2010. Disponível em: <http://www.informit.com/articles/article.aspx?p=1577449>.

FUENTES, V. B. Ruby on Rails: Coloque sua aplicação web nos trilhos. In: ____. 1 ed.. ed. [S.l.: s.n.], 2013. p. 76–78p.

GANDINI, João Agnaldo Donizeti; SALOMÃO, Diana Paola da Silva; JACOB, Cristiane. A segurança dos documentos digitais. *In: Âmbito Jurídico*, fev. 2002. Disponível em: <http://www.ambito-juridico.com.br/site/index.php>. Acesso em: 20 Nov. 2013.

GARFINKEL, Simson. Comércio e Segurança na Web. p. 378, 1999.

GOVERNOELETRÔNICO. Cartilha de Usabilidade. 2010. Disponível em: <http://www.governoeletronico.gov.br/acoes-e-projetos/padroes-brasil-e-gov/cartilha-de-usabilidade/index/?searchterm=usabilidade>.

ITI. Certificado Digital Como Obter. 2013. Disponível em: <http://www.iti.gov.br/twiki/bin/view/Certificacao/CertificadoObterUsar>.

JIMENE, C. Como utilizar de forma segura a documentação eletrônica. *ANFAC*, 2013. Disponível em: <http://www.anfac.com.br/v3/informativos-noticias.jsp?id=1024>. Acesso em: 18 fev. 2014.

KENAI, P. JRuby Compiler. 2013. Disponível em: <http://kenai.com/projects/jruby/pages/JRubyCompiler>.

LACORTE, Christiano Vitor de Campos. A validade jurídica do documento digital. *Revista Jus Navigandi - Doutrina e Peças.*, 2006. Disponível em: <http://jus.com.br/artigos/8524/a-validade-juridica-do-documento-digital>.

MAGNO, Alexandre. Globo Bootstrap. 2012. Disponível em: <http://globo.com>.

MONTEIRO, E; MIGNONI, M. Certificados Digitais. Conceitos e Práticas. 2007.

MONTMOLLIN, M. de. *L'Intelligence de la tâche: éléments d'ergonomie cognitive.* . [s.n.], 1986. Disponível em: <http://books.google.com.br/books?id=AWhnAQAACAAJ>.

NIELSEN, J. *Usability engineering*. [S.l.: s.n.], 1993.

OLIVEIRA, Danila Feitosa de C. *Levantamento e Desenvolvimento de website para Dispositivo Móvel de Acordo com Teorias de Usabilidade.* Tese (Monografia (Bacharelado em SIstemas de Informação)) — Universidade Federal do Piauí-UFPI, 2013.

OLIVEIRA, E. R. *Ruby - Conhecendo a Linguagem.* 1 ed.. ed. [S.l.: s.n.], 2006.

PINHEIRO, D.B.M; NETO, F.R.C. A utilização da certificação digital em documentos, transações comerciais e jurídicas. *Revista Fasem Ciências*, v. 2, n. 2, dez. 2012. Disponível em: <http://www.fasem.edu.br/revista/index.php/fasemciencias/article/download/19/pdf$_1$>.

RUBY-LANG.ORG. About Ruby. jul. 2013. Disponível em: <http://www.ruby-lang.org/pt/about>. Acesso em: 11 Jul. 2013.

RUBY, S.; THOMAS, D.; HANSSON, D. *Agile Web Development with Rails*. [S.l.]: Pragmatic Bookshelf, 2009.

SCHNEIER, Bruce. *Applied Cryptography.* 2.ed.. ed. [S.l.: s.n.], 1996. 675 p.

SILVA, L. G. C Da; SILVA, P.C Da; E.M., BATISTA. *Certificação Digital. Conceitos e Aplicações. Modelos Brasileiro e Australiano*. [S.l.: s.n.], 2011.

SOBRAL, F. Certificação Digital. nov. 2012. Disponível em: <http://biblioo.info/certificacao-digital>. Acesso em: 10 Out. 2013.

STEWART, B. An Interview with the Creator of Ruby. nov. 2001. Disponível em: <http://linuxdevcenter.com/pub/a/linux/2001/11/29/ruby.html>. Acesso em: 11 Jul. 2013.

STSC. Seguranca da Informação. 2013. Disponível em: <http://esaj.tjsc.jus.br/WebHelp/id$_{s}eguranca_{d}a_{i}nformacao.htmCertificado$>.$Acessoem$: $20Dez.2013$.

TADAMO, Katiucia Y. *GED: assinatura digital e validade jurídica de documentos eletrônicos*. Tese (Doutorado), 2002. Disponível em: <http://www.arquivar.com.br/espaco$_{p}rofissional/sala_{l}eitura/teses-dissertacoes-e-monografias/GED_{A}ssinatura_{D}igital.pdf/at_{d}ownload/file$>.$Acessoem$: $13Jul.2013$.

TEIXEIRA, Fabrício. O que e Responsive Web Design? dez. 2011. Disponível em: <http://arquiteturadeinformacao.com/mobile/o-que-e-responsive-web-design>. Acesso em: 20 Out. 2013.

TIOBE, software. The coding standards company. jul. 2013. Disponível em: <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>.

VIOTTI, Giovanni de Melo. Tecnologia vs. Sustentabilidade- Meios eletr{ônicos na economia de papel. Jun 2011. Disponível em: <http://www.tiespecialistas.com.br/2011/06/tecnologia-versus-sustentabilidade-meios-eletronicos-economia-de-papel>. Acesso em: 18 Out. 2013.

VOLPI, M. M. Assinatura digital: Aspectos técnicos, práticos e legais. 2001.

WEBB, Collin. The role of preservation and the library of the future. National Library of Australia, 2000. Disponível em: <http://www.nla.gov.au/openpublish/index.php/nlasp/article/view/1341/1625>. Acesso em: 16 Jul. 2013.

APÊNDICE A – *Constroller* de Planos de Curso

Esse *Controller* tem a função de manipular os planos de curso, e nele que são definidos os métodos executados com os planos de curso. A autenticação de documentos tem inicio nesse controlador.

planos_controller.rb

```
require 'securerandom'
require 'prawn'

class PlanosController < ApplicationController
  before_action :set_plano, only: [:show, :edit, :update, :destroy]
  skip_before_action :authorize, only: [:index, :show]

  # GET /planos
  # GET /planos.json
  def index
    @q = Plano.search(params[:q])
    @planos = @q.result(distinct: true)
  end

  # GET /planos/1
  # GET /planos/1.json
  def show
  end

  # GET /planos/new
  def new
    @plano = Plano.new
  end

  # GET /planos/1/edit
  def edit
  end

  # POST /planos
```

Figura 53 *– Controller de plano de curso (Parte 1).*

```
30  # POST /planos.json
31  def create
32    @plano = Plano.new(plano_params)
33    @plano.codigo_verificador = SecureRandom.hex(4)
34    codigo = @plano.codigo_verificador
35    filename = @plano.arquivo_file_name
36    respond_to do |format|
37      if @plano.save
38         pdf = Prawn::Document.new
39         pdf.move_down 700
40         pdf.text "Esse documento pode ser verificado em www.ufpi.br/
planodecurso #{codigo}"
41         pdf.render_file "#{Rails.root}/app/stamp/#{codigo}.pdf"
42        require 'posix/spawn'
43        ::POSIX::Spawn::Child.new "pdftk", "#{Rails.root}/public/system/
planos/arquivos/000/000/0#{@plano.id}/original/#{filename}", "stamp", "#
{Rails.root}/app/stamp/#{codigo}.pdf", "output", "#{Rails.root}/public/
system/planos/arquivos/000/000/0#{@plano.id}/original/copy-#{filename}"
44        ::POSIX::Spawn::Child.new "rm", "#{Rails.root}/public/system/
planos/arquivos/000/000/0#{@plano.id}/original/#{filename}"
45 ::POSIX::Spawn::Child.new "rm", "#{Rails.root}/app/stamp/#{codigo}.pdf"
46        ::POSIX::Spawn::Child.new "mv", "#{Rails.root}/public/system/
planos/arquivos/000/000/0#{@plano.id}/original/copy-#{filename}", "#
{Rails.root}/public/system/planos/arquivos/000/000/0#{@plano.id}/original/
#{filename}"
47        format.html { redirect_to @plano, notice: 'Plano was successfully
created.' }
48      else
49        format.html { render action: 'new' }
50        format.json { render json: @plano.errors,
status: :unprocessable_entity }
51      end
52    end
53  end
54
```

Figura 54 *– Controller de plano de curso (Parte 2).*

```ruby
  # PATCH/PUT /planos/1
  # PATCH/PUT /planos/1.json
  def update
    respond_to do |format|
      if @plano.update(plano_params)
        format.html { redirect_to @plano, notice: 'Plano was successfully updated.' }
        format.json { head :no_content }
      else
        format.html { render action: 'edit' }
        format.json { render json: @plano.errors, status: :unprocessable_entity }
      end
    end
  end

  # DELETE /planos/1
  # DELETE /planos/1.json
  def destroy
    @plano.destroy
    respond_to do |format|
      format.html { redirect_to planos_url }
      format.json { head :no_content }
    end
  end

  private
    # Use callbacks to share common setup or constraints between actions.
    def set_plano
      @plano = Plano.find(params[:id])
    end

    # Never trust parameters from the scary internet, only allow the white list through.
    def plano_params
      params.require(:plano).permit(:disciplina_id, :codigo_verificador, :professor, :arquivo, :semestre, :ano)
    end

end
```

Figura 55 *– Controller de palno de curso (Parte 3).*