

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO SUPERIOR DE LICENCIATURA EM INFORMÁTICA**

GUSTAVO FELIPE LATTMANN

INTEGRANDO MOODLE E ANDROID COM APPINVENTOR

TRABALHO DE CONCLUSÃO DE CURSO

**FRANCISCO BELTRÃO
2016**

GUSTAVO FELIPE LATTMANN

INTEGRANDO MOODLE E ANDROID COM APPINVENTOR

Trabalho de Conclusão de Curso de graduação, apresentado à disciplina de Trabalho de Conclusão de Curso, do Curso Superior de Licenciatura em Informática, da Universidade Tecnológica Federal do Paraná, Campus Francisco Beltrão, como requisito parcial para obtenção do título de Licenciado.

Orientador: Prof. Gustavo Yuji Sato

FRANCISCO BELTRÃO
2016

GUSTAVO FELIPE LATTMANN

INTEGRANDO MOODLE E ANDROID COM APPINVENTOR

Trabalho de Conclusão de Curso, apresentado a Universidade Tecnológica Federal – Campus Francisco Beltrão, como parte das exigências para a obtenção do título de Licenciado em Informática.

UTFPR-FB, 20 de junho de 2016.

BANCA EXAMINADORA

Prof. Me. Gustavo Yuji Sato (Orientador)
Mestre em Ciências da Computação

Prof. Dra. Maici Duarte Leite (Convidado)
Doutora em Informática

Prof. Francisco A. F. Reinaldo (Presidente da Banca)
Doutor em Engenharia Electrotécnica e de Computadores

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso”

DEDICATÓRIA

Dedico esse trabalho a minha família, Rozimar Scorsatto, Walmor Agostinho da Silva e Rafaella Scorsatto da Silva que em todos os momentos difíceis estavam dispostos a me ouvir e me aconselhar, guiando sempre a fazer as escolhas certas. Sem esquecer é claro, dos avós, tios, primos, onde cada aprendizado ao longo da vida, foi necessário nessa fase.

A minha namorada, amiga e companheira Laislenen Rachurat, que teve compreensão nos momentos em que foi necessário abrir mão do lazer para me dedicar aos estudos, me dando sempre uma força incrível e o ombro nos momentos necessários.

Aos meus amigos que serviram de estímulo e referência para que não desistisse, em especial ao Jorge José Kleinubing, o qual inclusive me apresentou ao curso de Licenciatura em Informática.

AGRADECIMENTO

Agradeço inicialmente a Deus, por ter feito com que o meu caminho me trouxesse até aqui.

Agradeço aos professores que me acompanharam em todo o processo de ensino, e que me auxiliaram a criar uma base, na qual me apoiei para concluir mais essa etapa, sempre com paciência e dedicação, Em especial aos professores Gustavo Yuji Sato e Renato Hallal que aceitaram o convite para me orientar nesse trabalho. Ao professor Francisco Antonio Fernandes Reinaldo, que foi um facilitador nessa etapa do Trabalho de Conclusão de Curso.

Tudo é possível, o impossível apenas demora mais.

Dan Brown

RESUMO

LATTMANN, Gustavo Felipe. Integrando Moodle e Android com AppInventor. 2016. 117 f. Monografia (Trabalho de Conclusão de Curso) - Curso Superior de Licenciatura em Informática, Universidade Tecnológica Federal do Paraná, Campus Francisco Beltrão. Francisco Beltrão, 2016.

O levantamento realizado na Universidade Tecnológica Federal do Paraná, Campus Pato Branco, os professores do curso técnico-integrado em Agrimensura identificaram que embora os materiais trabalhados em aula possam ser postados no ambiente virtual de aprendizagem, existe a dificuldade de acesso a essas informações através de dispositivos móveis. Neste contexto surgiu a ideia de desenvolver um aplicativo para facilitar o acesso dos alunos a esses conteúdos. Ao realizar uma pesquisa para identificar o que já existe de aplicativos Mobile, foi identificado um aplicativo oficial do Moodle, porém o mesmo não apresentava facilidade no uso. Deste modo o presente trabalho propõe um aplicativo Android que se integre a um ambiente virtual de aprendizagem, no caso o Moodle. Deste modo permitindo uma melhora no acesso a informação por parte dos alunos, que atualmente vivem em um mundo totalmente conectado. O aplicativo foi desenvolvido utilizando o App Inventor, software criado pelo MIT que facilita o desenvolvimento de programas para Android. Para auxiliar a realização do trabalho, utilizou-se a abordagem com o ciclo de vida Incremental. Obteve-se por fim o aplicativo conforme proposto, que permite que os alunos baixem arquivos disponíveis no servidor do Moodle para seus celulares.

Palavras-chave: Educação a Distância. Moodle. App Inventor.

ABSTRACT

LATTMANN, Gustavo Felipe. Integrando Moodle e Android com AppInventor. 2016. 117 f. Monografia (Trabalho de Conclusão de Curso) - Curso Superior de Licenciatura em Informática, Universidade Tecnológica Federal do Paraná, Campus Francisco Beltrão. Francisco Beltrão, 2016.

According to a survey conducted at the Universidade Tecnológica Federal do Paraná, Campus Pato Branco, teachers of technical-integrated course in surveying found that although the materials worked in class can be posted in the virtual learning environment, there is the difficulty of access to this information through mobile devices. In this context the idea to develop an application to facilitate student access to these contents. By conducting a survey to identify what already exists Mobile application, an official application Moodle has been identified, but it was inefficient for students and contained many defects. Thus this paper proposes an Android application that integrates a virtual learning environment, where Moodle. Thereby enabling an improvement in access to information by students, who are currently living in a totally connected world. The application was developed using the App Inventor, software created by MIT which facilitates the development of programs for Android. To assist the performance of work, the approach used, with incremental life cycle. Was obtained finally the application as proposed, which allows students to download files available in the Moodle server to their mobile phones.

Palavras-chave: Distance Learning. Moodle. App Inventor.

LISTA DE FIGURAS

FIGURA 1 - OBJETO EM JSON	21
FIGURA 2 - LISTA ORDENADA DE VALORES EM JSON.....	21
FIGURA 3 - TELA DE ACESSO AO SISTEMA.....	26
FIGURA 4 - CAMPOS OBRIGATÓRIOS NÃO PREENCHIDOS.....	27
FIGURA 5 - CURSOS MATRICULADOS.....	27
FIGURA 6 – SEM ARQUIVOS PARA DOWNLOAD	28
FIGURA 7 - ARQUIVO PARA DOWNLOAD	28
FIGURA 8 - APP INVETOR (DESING DE INTERFACE)	30
FIGURA 9 - APP INVENTOR (BLOCOS DE CÓDIGO)	31
FIGURA 10 - CHAMADA WEB SERVICE	31

LISTA DE SIGLAS

EAD	Educação a Distância
XML	Linguagem Extensível de Marcação
HTTP	Protocolo de Transferência de Hipertexto
JSON	Notação de Objetos JavaScript
W3C	World Wide Web Consortium
WSDL	Linguagem de Descrição de Serviços Web
SOAP	Protocolo Simples de Acesso a Objeto
UDDI	Universal Description, Discovery, and Integration
REST	Transferência de Estado Representacional

SUMÁRIO

1 INTRODUÇÃO	11
1.1 CONSIDERAÇÕES INICIAIS	11
1.2 OBJETIVOS	12
1.2.1 Objetivos Específicos	12
1.3 JUSTIFICATIVA	13
1.4 ESTRUTURA DO TRABALHO	14
2 REFERENCIAL TEÓRICO	15
2.1 EDUCAÇÃO A DISTÂNCIA.....	15
2.2 AMBIENTES VIRTUAIS DE APRENDIZAGEM.....	16
2.3 MOBILE LEARNING (M-LEARNING)	17
2.4 WORLD WIDE WEB (WWW).....	18
2.4.1 Web Services	18
2.4.2 REST	20
2.4.3 Notação de Objetos JavaScript (JSON)	21
3.1 MATERIAIS	22
3.2 MÉTODO.....	23
4 RESULTADOS	25
4.1 ESCOPO DO SISTEMA	25
4.2 APRESENTAÇÃO DO SISTEMA	25
4.3 IMPLEMENTAÇÃO DO SISTEMA	29
4.4 DISCUSSÕES.....	32
4.5 DIFICULDADES ENCONTRADAS.....	32
4.5.1 Integração e Infraestrutura	33
4.5.2 Moodle.....	33
4.5.3 App Inventor	34
REFERÊNCIAS	36

1 INTRODUÇÃO

O presente capítulo tem como objetivo abordar alguns aspectos introdutório que se intermediam o trabalho.

1.1 CONSIDERAÇÕES INICIAIS

A educação a distância aos poucos tem conseguido alterar a cultura e a forma tradicional da educação, pois permite que o aluno consiga ter acesso ao material didático pela internet, flexibilizando os horários de estudo, solucionando problemas de deslocamento e distanciamento. Surgiu de forma exitosa na Inglaterra por volta de 1970, aplicado ao ensino superior, e seguido como modelo no resto do mundo. Através de sua popularização sofreu adaptações e chegou ao ensino médio e aos demais níveis de ensino (NUNES, 2009).

No Brasil existem registros de jornais que circulavam por volta de 1900, que ofereciam cursos profissionalizantes por correspondência, os quais podem ser considerados como primeiros registros da existência da educação a distância no país. Baseado nesses moldes que ocorreu a evolução para chegar no cenário atual onde, os cursos a distância são ofertados totalmente online, com apenas provas presenciais (ALVES, 2009).

Nota-se então a presença marcante da informática na educação, e foi através da educação à distância que essa relação se estreitou, surgiram temas como AVA (Ambiente Virtual de Aprendizagem), TIC (Tecnologia da Informação e Comunicação) entre outros, além de recursos computacionais físicos, como os próprios computadores, projetores multimídia.

Com a disseminação de recursos educacionais digitais, as plataformas específicas à aprendizagem tornaram todo esse processo mais acessível, como o Moodle, plataforma de educação a distância, ou Ambiente Virtual de Aprendizagem, de código aberto e licença gratuita. Essas ferramentas estão a disposição para serem utilizadas como base da educação totalmente a distância, ou como ferramenta de apoio para educação tradicional. Levar então plataformas de educação a distância para a realidade dos alunos, através dos computadores e dos dispositivos móveis, poderia trazer aumentos significativos no número de usuário e

melhoria, assim o contato dos alunos aos conteúdos, buscando reduzir o índice de reprovação.

Baseado nesses fatos a aplicação de um ambiente virtual de aprendizagem, proporcionaria ao aluno mais recursos para estudar em contra turno, além de ter uma relação integrada de todas as atividades, que foram e ainda vão ser realizadas, com informações de notas parciais, tudo de forma clara e limpa.

Para levar os AVA para os dispositivos móveis será necessário o desenvolvimento de um aplicativo mobile, compatível com o sistema operacional Android.

Para realizar o processo de desenvolvimento de software, será necessário realizar as etapas previstas pela engenharia de software, como a definição do ciclo de vida, levantamento de requisitos, projeto, cronograma, documentação de testes entre outros.

1.2 OBJETIVOS

Como objetivo geral, este projeto pretende desenvolver um software para dispositivos móveis Android que se integre ao ambiente virtual de aprendizagem Moodle, para que seja utilizado pelos alunos do ensino fundamental, médio e superior utilizando App Inventor.

1.2.1 Objetivos Específicos

- Estudar como integrar ferramentas desenvolvidas externamente ao Moodle.
- Desenvolver um aplicativo mobile que se integre ao ambiente virtual de aprendizagem Moodle;
- Demonstrar como o AppInventor pode ser utilizado para desenvolver sistemas complexos.

1.3 JUSTIFICATIVA

O uso de recursos digitais na educação é defendido pelo MEC nos objetivos do Programa Nacional de Tecnologia Educacional (ProInfo) criado em 1997 e alterado em 2007. Segundo Relatório Educação para todos no Brasil 2000-2015 o ProInfo:

Trata-se de uma ação educacional do MEC, que visa fomentar o uso das tecnologias de informação e comunicação (TIC) como ferramentas de enriquecimento pedagógico no ensino público fundamental e médio, promovendo a inclusão digital e social das comunidades escolares atendidas.

Tem como objetivo inserir nas escolas de educação básica brasileiras, rurais e urbanas, as Tecnologias de Informação e Comunicação (TIC) no processo de ensino-aprendizagem com a instalação de laboratórios de informática, soluções tecnológicas baseadas em mídias digitais e conteúdos digitais de qualidade, com capacitação dos professores e alunos dessas escolas, promovendo, com isso, não só a melhoria do processo educacional, mas também a inclusão social e digital das comunidades escolares brasileiras, uma vez que, também contempla os requisitos de acessibilidade e disponibiliza recursos de tecnologia assistiva de uso pedagógico (BRASIL, 2014).

Com o auxílio de projetos como o citado acima, e o apoio do Estado, as tecnologias têm sido incorporadas a educação, e servem como grande recurso e aliado pois dinamiza e flexibiliza o processo de ensino.

Assim considerando, que a educação utiliza-se de recursos tecnológicos para aprimorar seus métodos, levando em conta que recursos tecnológicos não são somente equipamentos multimídias, ou projetores de slide, mas sim componentes que possam auxiliar o processo de gestão ou de ensino-aprendizagem de fato, nota-se que a tecnologia voltada a educação rompe os muros da sala de aula, ou deve romper, e estar disponível para o aluno no momento que o mesmo necessitar buscar conteúdo, ou buscar materiais que possam ajudar a obter conhecimento a compreender ou reforçar o que foi realizado em sala.

Portanto, o acesso rápido a informação se faz necessário, e como solução surgem conceitos como o de *Mobile Learning*, permitindo a aprendizagem em qualquer lugar.

1.4 ESTRUTURA DO TRABALHO

Nessa seção será abordado o fato da educação a distância servir de base e de fomento para o surgimento dos ambientes virtuais de aprendizagem, afinal foi da necessidade de evoluir a EAD, que se fizeram necessários os ambientes virtuais de aprendizagem.

Os ambientes virtuais de aprendizagem, são softwares utilizados no processo ensino educação, de um modo onde mesmo remotamente o professor ou tutor, consegue propor um ambiente controlado para o aluno.

Sistemas distribuídos são a base teórica para todo sistema que utilize uma estrutura de internet, onde o cliente esteja alocado em um computador, e o servidor que irá processar as requisições, esteja distante, até mesmo em outro país.

Para obter o resultado, de digitar um endereço no navegador e uma página ser exibida, são necessárias tecnologias e métodos.

Alguns materiais foram utilizados para a realização deste trabalho. Como base o App Inventor, aplicativo criado pelo MIT com a filosofia de permitir que qualquer usuário com um conhecimento básico de programação, consiga criar um aplicativo simples, para o sistema operacional de smartphones Android, em poucos minutos. Embora o uso do App Inventor seja simples, é uma ferramenta potente, que permite criar softwares complexos, os quais demandam mais empenho, e podem até mesmo integrar-se a outros sistemas.

O método incremental permite maior agilidade no desenvolvimento, gerando versões, permitindo o uso do software sem que o mesmo esteja completamente pronto.

Serão discutidos os resultados do trabalho, onde o aplicativo proposto como objetivo, foi desenvolvido, utilizando o App Inventor, um software muito didático que tem seu desenvolvimento feito através de blocos que se encaixam. Com o objetivo de integrar-se ao AVA, o aplicativo utiliza de conexões *web services* para se comunicar a um ambiente do Moodle já configurado.

Deste modo utilizando o aplicativo desenvolvido, o aluno, pode ter acesso aos documentos disponibilizados pelos professores no AVA e fazer download em seu smartphone, de forma ágil e fácil.

2 REFERENCIAL TEÓRICO

A educação a distância é um tema de extrema importância para esse trabalho, pois através da mesma houve a necessidade de se criar ambientes virtuais de aprendizagem para obter uma atmosfera controlada nas modalidades onde o professor não está presente. Deste modo é importante citar autores que descrevem sobre o assunto, já que o Moodle é um AVA.

A internet serve como infraestrutura para esse trabalho, já que a integração entre Moodle e o aplicativo Android desenvolvido ocorrerá através de *web services* que utilizam de protocolos de internet para realizar a comunicação entre sistemas de forma mais rápida. Para simplificar ainda mais o uso de *web services* utiliza-se do REST que é um padrão básico de comunicação.

2.1 EDUCAÇÃO A DISTÂNCIA

Para chegar ao nível que se tem hoje, a educação a distância e os recursos educacionais digitais passaram por uma grande evolução. A educação a distância surgiu na Europa, com o objetivo de flexibilizar a educação e as formas de ensino, e inicialmente ocorria por meio de correspondência que era enviada para os alunos semanalmente (NUNES, 2009).

No Brasil a educação a distância chegou pouco antes de 1900, como mostram estudos realizados pelo IPEA, onde no Rio de Janeiro, jornais como o Jornal do Brasil ofereciam anúncios de cursos não presenciais. Em 1904 com instalação das Escolas Internacionais, houve uma grande popularização da modalidade, que também chegava aos alunos por correspondência, e eram ofertados basicamente a quem estava a procura de formação para obter emprego. Entre os cursos mais ofertados estava o curso de datilografia. Esta estrutura permaneceu por aproximadamente 20 anos, quando em 1923 iniciaram as ofertas de cursos pelo rádio com a Rádio Sociedade do Rio de Janeiro, uma iniciativa privada que obteve êxito durante os primeiros anos, mas depois por forte pressão e exigência do governo acabou perdendo seus fins comerciais e foi doada ao Ministério da Saúde e Educação (ALVES, 2009).

Outro marco para a EAD no Brasil, foi o uso da televisão para a prática da educação, o que iniciou por volta de 1960, e teve uma explosão em 1969 quando o Ministério das Comunicações ditou regras obrigando as emissoras comerciais a

ceder espaço para a transmissão de programas educativos como o popular Tele Curso 2000.

Os cenários da EAD no Brasil tiveram início então a uma nova fase, agora com o uso das tecnologias computacionais que se popularizaram no país junto aos computadores pessoais e o avanço da própria internet, possibilitando então, cursar uma graduação a distância. Foi a partir dessa fase que começam a surgir os recursos educacionais digital.

A partir desse ponto, a tecnologia democratizou-se e passou a fazer parte integrante da educação, não somente para atualizar a educação a distância, mas sim como recursos de apoio para os professores e alunos, levando a oportunidade de ter material de aprendizagem para fora dos muros das escolas, e dando dois rumos a educação a distância: o primeiro como já se conhecia, totalmente a distância, e o segundo como ferramenta de apoio ao professor.

2.2 AMBIENTES VIRTUAIS DE APRENDIZAGEM

Através do uso de Ambientes Virtual de Aprendizagem (AVA's), é possível expandir o espaço e tempo de aprendizagem para fora dos muros das escolas. Assim o aluno se torna mais independente pois passa a buscar a informação, e o professor passa a ser um moderador, que pode produzir ou indicar onde o aluno consegue a informação que necessita, também potencializa a comunicação entre professor e aluno, onde uma troca de mensagens pode ocorrer a qualquer horário, sem a preocupação com a formalidade do ambiente de sala de aula.

Deste modo o professor disponibiliza o conteúdo em ambiente virtual e o aluno pode acessar quando e de onde quiser, quantas vezes quiser, tornando o processo de ensino muito mais flexível. Esse conteúdo ou material de aprendizagem, como é possível chamar, não precisa ser propriamente um livro didático, mas vídeos, imagens, animações, simulações, software, jogos, páginas Web entre outros. Assim é possível abordar várias formas de aprendizagem, melhorando o processo de ensino-aprendizagem por utilizar de recursos visuais, sonoros etc.

Com o uso dos AVA's é possível proporcionar aos alunos novas experiências de aprendizagem e modificar o método tradicional de ensino que já percorre tantas décadas, pois em tempos modernos, devido ao fato de terem incorporado a tecnologias em sua rotina, as pessoas necessitam ter a informação na palma da

mão, no horário que preferirem, aproveitando melhor seu tempo e seu rendimento na absorção de conteúdo (JR.; MARQUESI, 2008).

Para então conseguir disponibilizar de forma ainda mais fácil e permitir que o aluno tenha a informação na mão quando puder estudar, nem sempre apenas recursos web como AVA's poderão auxiliar, pois hoje os recursos tecnológicos convergem para a portabilidade. É a partir desse ponto que surge o conceito de *Mobile Learning* que envolve o uso de *smartphone*, *tablet's* e outros dispositivos móveis "para permitir a aprendizagem a qualquer momento e em qualquer lugar" (UNESCO, 2015).

A UNESCO é uma forte aliada e defensora do conceito de *Mobile Learning*, pois acredita que pode melhorar a qualidade da educação, e organiza anualmente a conferência *Mobile Learning Week*, que reúne líderes educacionais de todo o mundo para discutir melhorias e como expandir esse conceito.

2.3 MOBILE LEARNING (M-LEARNING)

O aumento do uso de dispositivos móveis como *tablets* e *smartphones* tem modificado a forma de realizar várias tarefas. Esses novos dispositivos já conseguiram alterar o modo como as pessoas se comunicam, ouvem música. Tornaram o despertador, relógio, mapa, câmeras fotográficas obsoletas. E essa é uma tendência crescente que continuará a inutilizar outros equipamentos mais "rudimentares".

Deste modo as tecnologias estão sendo integradas a educação, como um caminho claro para melhorar a qualidade da eficiência educacional. Surge então a aprendizagem móvel, ou *Mobile Learning* em inglês na qual é definida pela UNESCO como "pessoal, portátil, colaborativa, interativa, contextual e situada; ela enfatiza a "aprendizagem instantânea", já que a instrução pode ocorrer em qualquer lugar e a qualquer momento".

Devido a todas essas vantagens que se atribuem ao *Mobile Learning* somado a democratização dos aparelhos móveis, se tem uma das tecnologias mais bem-sucedidas da humanidade. Afinal mais de 6 bilhões de pessoas no mundo já tem acesso a esse tipo de dispositivos.

Desta forma, a UNESCO acredita que a aplicação do *Mobile Learning* "surge como uma das soluções para os desafios enfrentados pela educação. Melhorar o

acesso e a qualidade da educação requer liderança política, planejamento e ação. As tecnologias móveis têm a chave para transformar a exclusão digital que existe atualmente em dividendos digitais, trazendo educação igualitária e de qualidade para todos”.

Por fim, é necessário que a educação aceite essa revolução tecnológica e use desses novos recursos, para que também ocorra uma revolução na educação. (UNESCO, 2015).

2.4 WORLD WIDE WEB (WWW)

Ao falar de uma estrutura de sistemas distribuídos funcionando na web, é importante entender a estrutura cliente-servidor existente na maioria dos casos. Os sites são um excelente exemplo dessa estrutura. Acessados por um software específico, chamado de browser - navegador em português, geralmente o Google Chrome, Internet Explorer, entre outros - o qual é responsável por apresentar o conteúdo de forma adequada.

Assim, ao acessar um site através do browser, utiliza-se uma URL, Uniform Resource Locator - localizador uniforme de recurso, em português - para chegar ao documento requisitado no servidor. Deve-se notar, que o navegador na maioria dos casos, está presente no cliente, e ao acessar um endereço web através de uma URL, irá fazer uma solicitação para um servidor, de um determinado documento. Não se esquecendo que documento está em um sentido que pode ser, página da internet, áudio, vídeo, imagem, texto, entre outros.

Para a comunicação entre cliente-servidor, browser-servidor, o protocolo padronizado é o HTTP, Hypertext Transfer Protocol - protocolo de transferência de hipertexto. (TANENBAUM, STEEN. 2009)

2.4.1 Web Services

Os Web Services popularizaram-se nos últimos anos. Devido ao avanço da internet. Obtém destaque atualmente sendo a principal tecnologia de integração entre sistemas, ou entre cliente e servidor. Por oferecer agilidade, eficácia e segurança entre outros benefícios. Essa popularização ocorreu devido a possibilidade de integrar sistemas heterogêneos, desenvolvidos em linguagens

diferentes de forma simples. Para isso basta apenas converter as informações a serem comunicadas para o formato XML ou JSON que são formatos utilizados para transmissão de dados via *Web Services*.

Sendo assim a definição da W3C sobre web service surge por volta de 2004 e diz que "Um serviço Web é um sistema de software projetado para suportar interação máquina-a-máquina através de uma rede. Ele tem uma interface descrita em um formato de máquina processável (chamado WSDL). Outros sistemas interagem com o serviço Web de forma prescrita pela sua descrição usando mensagens SOAP, normalmente transmitida através de HTTP com uma serialização XML em conjunto com outras normas relacionadas à Web". (W3C, 2004).

A partir do momento que se inicia a fala sobre web services, não se define mais como apenas o browser no cliente como forma de interação com um servidor web. Porém para que um serviço web seja considerado de fato como um *web service*, o mesmo deve adotar uma série de padrões para que possa ser descoberto e consumido por um cliente.

Dentre esses padrões um componente dentro da arquitetura dos *web services*, que trata-se de um base de dados que armazena as descrições dos serviços. A UDDI (Universal Description, Discovery and Integration), descreve qual será o layout dessa base de dados onde constará a descrição dos serviços. As descrições servirão de documentação para o cliente encontrar o serviço correto a utilizar.

Também, para a descrição dos *web services*, existe uma linguagem adequada. WSDL (Web Services Definition Language), em português linguagem de definição de serviços web. Essa linguagem é muito similar a outras linguagens de programação, e com ela são definidas quais as entradas de dados, qual o formato dos dados, entre outros. (TANENBAUM e STEEN. 2009).

Além das componentes citadas acima, é necessário também o protocolo pelo qual fará a comunicação, o SOAP, protocolo simples de acesso a objeto.

Embora parece pouco potente, os serviços web podem se tornando complexos e compondo uma grande estrutura, relacionando a serviços web de outros fornecedores, ou até mesmo ao serem requisitados, utilizarem mais de um servidor para processar a requisição invocada.

Assim é necessário perceber que os serviços web são utilizados diariamente por quem navega pela internet. Por exemplo, ao realizar uma compra online,

preenchendo o CEP, o site de compras irá se comunicar via web service com o sistema dos Correios, para obter as informações de valores do frete, prazo de entrega de forma sincronizada, consultando todas essas informações atualizadas. Em um cenário sem web services, o site de vendas teria que elaborar por exemplo, um arquivo, contendo essas informações, o que dependeria de uma "carga" para ser atualizado. Existem além da web service dos Correios muitos outros utilizados diariamente, mas que por características desses serviços, não diferem para os usuários. É o caso da nota fiscal eletrônica brasileira, onde o XML gerado no momento da transmissão da nota, é enviado para a receita através de web service. Sem contar nas transações dos cartões de créditos realizadas pela internet, as quais retornam o processamento rapidamente, devido aos serviços web.

Voltando ao cenário do *web service* dos Correios, os mesmos desenvolveram esse serviço para que seus clientes consigam consultar em tempo real as informações de frete. O *web service* dos Correios está sempre disponível para ser consultado, porém necessita de alguns parâmetros para conseguir retornar o valor do frete e prazo de entrega. Quem deseja consultar então essas informações, ou seja, consumir o recurso desse *web service* deve se comunicar com o mesmo, informar os parâmetros necessários, que seriam CEP de origem e CEP de destino e aguardar o retorno.

Tendo em vista tantas entidades renomadas que se integram através de *web service*, o ambiente virtual de aprendizagem Moodle, não poderia ser diferente.

2.4.2 REST

Além das definições de web services, existem técnicas para facilitar o desenvolvimento. Uma dessas técnicas é o REST (Representational State Transfer) desenvolvida por Roy Fielding em sua tese de doutorado. Neste modelo Roy defende a riqueza do protocolo HTTP. Propondo então que é possível criar serviços complexos usando puramente web, acessando serviços através apenas de uma URL. Porém o REST cria restrições sobre a web, para definir um padrão. (FIELDING, 2000).

Deste modo, pode-se enxergar o REST como um design de arquitetura para aplicações web, que traz uma série de vantagens e recomendações.

Baseando nessas recomendações, cada recurso a ser acessado no servidor pelo *web service* deve ter um acesso específico. Ou seja, para acessar os usuários cadastrados no servidor, pode ser utilizado `http://enderecodoservidor/users`. Entretanto é importante que esse acesso seja coeso, para facilitar o uso e documentações.

Ainda assim, após a solicitação ao servidor, o mesmo irá retornar os usuários, seguinte o exemplo, e esse retorno pode ser HTML, XML, JSON, TXT. No caso da integração com o Moodle, os retornos utilizados foram no formato JSON.

2.4.3 Notação de Objetos JavaScript (JSON)

Como já citado anteriormente, o JSON é um dos formatos utilizados para receber retorno de web services. Segundo o site oficial do JSON, está definido como “uma formatação leve de troca de dados. Para seres humanos, é fácil de ler e escrever. Para máquinas, é fácil de interpretar e gerar”.

Por ser apenas conjuntos de textos pode ser utilizado com qualquer linguagem de programação, e atualmente a maioria das linguagens já consegue trabalhar com JSON, como C, Java, Python.

O JSON é formado por duas estruturas. A primeira é uma lista que inicia com chaves e termina com chaves, e tem como conteúdo pares, de nome e valor, separados por dois pontos. Cada par é separado por vírgula. (JSON, 2016).

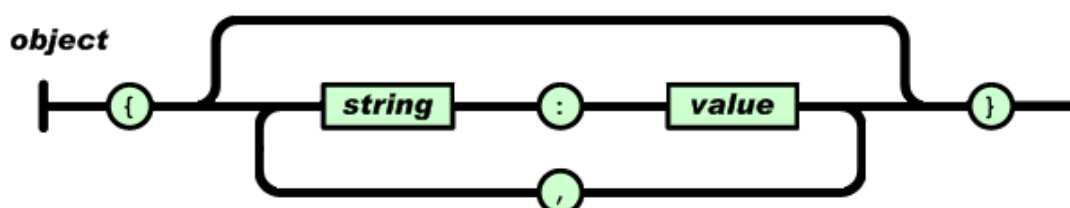


Figura 1 - Objeto em JSON

A segunda estrutura é uma lista ordenada de valores, ou um *array*. Quando um objetivo tem mais de um valor.

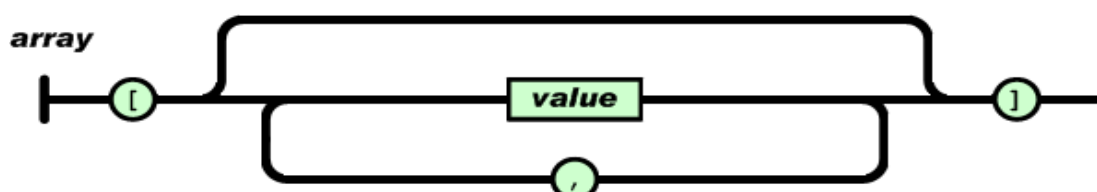


Figura 2 - Lista Ordenada de valores em JSON

3 MATERIAIS E MÉTODO

Para o desenvolvimento deste projeto serão necessários alguns materiais como computadores e softwares especificados a seguir. Os mesmos são listados abaixo, como requisitos mínimos para atingir os objetivos especificados.

Todo o projeto envolve uso de computadores, mas a etapa em que se exige mais recursos é o momento do desenvolvimento, onde serão utilizados softwares para desenvolvimento em ambiente Android, como o App Inventor, o qual é gratuito e pode ser utilizado através do navegador. É compatível com a maioria dos navegadores do atuais.

Os testes devem ser realizados utilizando smartphone com sistema operacional Android versão 4.0 ou posterior, que pode ser utilizada via emulador, e acesso à internet. Neste caso não é especificado nenhum requisito mínimo de hardware do smartphone.

O software a ser desenvolvido utilizará de integração via webservice com um ambiente do Moodle, portanto é necessário que esse ambiente esteja preparado e configurado, com os webservices habilitados e em funcionamento. O servidor deste ambiente Moodle deve ter os seguintes requisitos: 160Mb de espaço livre de armazenamento em disco, porém o recomendado é 5Gb, processador de no mínimo 1Ghz, 2Ghz é o recomendado e 256Mb de memória, mas o recomendado é 1Gb (MOODLE, 2015).

3.1 MATERIAIS

O App Inventor é um software web desenvolvido pelo MIT com o objetivo de que qualquer pessoa, mesmo que um iniciante em programação consiga desenvolver seu aplicativo Android em questão de uma hora ou menos e é uma ótima ferramenta para ensinar/aprender programação.

Essa proposta tão otimista é possível de fato, devido ao fato de nenhuma linha de código precisar ser escrita. Todos os blocos de código, já estão disponíveis em forma de blocos. Permite encaixar os comandos, os quais estão estruturados e formam um algoritmo ao juntá-los, o que lembra muito a linguagem LOGO.

Do mesmo modo como o Scratch, software também desenvolvido pelo MIT que também usa o modelo de blocos, que se encaixam para gerar algoritmos, o App Inventor busca democratizar o desenvolvimento de software.

Criado em 2009 por Mark Friedman do Google e Hal Adelson professor do MIT, é mantido pela equipe de Mobile Learning do MIT, MIT Media Lab e Laboratório de Ciências da Computação e Inteligência Artificial do MIT. No ano de 2015 o App Inventor bateu a marca de mais de 3 milhões de usuários em 195 países.

Todo esse sucesso foi devido aos mais diversos grupos de usuário, no qual o App Inventor conseguiu chamar a atenção de:

Educadores formais e informais que usam o App Inventor para introduzir a programação aos seus alunos de Ciência da Computação, membros do clube de ciências, pós-escolares participantes dos programas, e campistas de verão. Muitos educadores também começaram a usar o App Inventor para desenvolver aplicativos em apoio dos seus próprios objetivos instrucionais.

Funcionários públicos e cívicos e voluntários que têm aproveitado o poder do App Inventor para desenvolver aplicativos personalizado, muitas vezes locais em resposta a desastres naturais e as necessidades de base comunitária.

Designers e gerentes de produto que viram o potencial que o App Inventor tem de apoiar o processo de design interativo via rápida prototipagem, teste e iteração.

Pesquisadores que usam App Inventor para criar aplicativos personalizados que podem processar a sua análise dados e de requisitos em uma ampla variedade de áreas.

Amadores que querem transformar rapidamente uma ideia em um aplicativo sem a curva de aprendizagem ou custo dos processos de criação de aplicativo mais tradicionais. (APP INVENTOR, 2016).

A escolha do Android como sistema operacional de foco deste trabalho trata-se a grande fatia do mercado ocupada pelo sistema da Google. Segundo o Gartner em 2014, 80.7% dos smartphones comercializados continham sistema operacional Android. Baseado neste dado, criar um aplicativo para o sistema mais comercializado do mercado, abrangeria maior quantidade da população (GARTNER, 2015).

3.2 MÉTODO

O método utilizado para o presente trabalho, apresentado por Sommerville (2011) na sua publicação Engenharia de Software, será o modelo de desenvolvimento iterativo e incremental, pois se sobressai sobre o modelo em

cascata devido a ser mais parecido com a forma que resolvemos os problemas no dia-a-dia, quando uma solução não é criada de maneira completa inicialmente. Como pontos positivos a serem ressaltados pela escolha do método incremental estão fato de que a quantidade de documentações é reduzida em relação ao método cascata, fica mais fácil para ser recebido o feedback em uma versão 1.0 do que em documentos de requisitos, e também é possível acompanhar a evolução com o passar das versões. É possível que ocorra o uso do software sem ele estar completamente pronto.

Utilizando o modelo incremental as três fases podem ocorrer simultaneamente após a definição do escopo. Essas fases são Especificação, Desenvolvimento e Validação. São geradas então versões intermediárias até se obter a versão final. (SOMMERVILLE, 2011).

A escolha dessa metodologia ocorreu devido ao fato de ser indicada para quando se não conhece profundamente a tecnologia a ser utilizada, e pode ser desenvolvido por pessoas que possuam menos conhecimento, devido os requisitos serem divididos em pequenas partes.

4 RESULTADOS

O resultado esperado para esse trabalho tratava-se de desenvolver um aplicativo que simplificasse o acesso dos alunos a informação, permitindo fazer download de arquivos disponíveis no ambiente Moodle, em seus smartphones Android.

Embora o aplicativo não tenha sido testado com os alunos, conseguiu-se desenvolver o que era proposto.

4.1 ESCOPO DO SISTEMA

O Moodle Aluno App foi um sistema desenvolvimento apenas para smartphones com sistema operacional Android. Tem como ideia central facilitar o acesso de alunos a informação disponível no Moodle, permitindo de forma simplificada o download de arquivos.

Assim, ao acessar o sistema com o usuário e senha já cadastrados no Moodle da instituição de ensino, o aluno terá acesso as disciplinas na qual está matriculado, e poderá realizar o download dos arquivos disponíveis.

Entretanto, o sistema se restringe ao download dos arquivos para o dispositivo do aluno, não se comprometendo com pré-visualizações e a visualização do arquivo de fato no aparelho.

4.2 APRESENTAÇÃO DO SISTEMA

A tela de *login* é exibida ao abrir o aplicativo. Nesta interface, o usuário irá informar o endereço do servidor do Moodle onde está acessando. Esse endereço é o mesmo que o informado para acessar o Moodle pelo navegador, conforme Figura 3.



The image shows a mobile application interface for Moodle login. At the top, there is a status bar with icons for signal, Wi-Fi, battery, and time (19:36). Below the status bar is a header labeled 'Login'. The main content area features the Moodle logo (a graduation cap above the word 'moodle' in orange). Below the logo is a 'Site URL' label followed by a text input field containing the example 'Ex: www.utfpr.edu.br'. Underneath is a 'Usuário' label followed by a text input field with the placeholder text 'Informe seu Usuário'. Below that is a 'Senha' label followed by a text input field. At the bottom of the form are two buttons: 'Login' and 'Limpar'.

Figura 3 - Tela de acesso ao sistema

Nos campos seguintes, deverá preencher com usuário e senha de acesso cadastrados no Moodle.

Nesta tela, todos os campos são obrigatórios, e caso não sejam preenchidos, o usuário não conseguirá realizar o *login*.

18:32 23%
Login

moodle

Site URL
moodle2.fb.utfpr.edu.l

Campo usuário não preenchido!

Senha

Login Limpar

Figura 4 - Campos obrigatórios não preenchidos

Com todos os campos preenchidos, ao pressionar no botão “login” o sistema irá tentar encontrar o servidor do Moodle, e verificar se o usuário e senha informados estão corretos. Caso não estejam, uma mensagem de notificação será retornada. No caso de todos os dados estarem corretos, uma tela com todas os cursos matriculados será exibida.

22:05 76%
Menu

Search list...

4 Banco de Dados

3 Programação I

2 Trabalho Conclusão de Curso

Figura 5 - Cursos Matriculados

Nesta segunda tela, são exibidos apenas os cursos matriculados do aluno que está autenticado. Dentre os recursos disponíveis é possível filtrar os cursos, em caso de uma lista extensa.

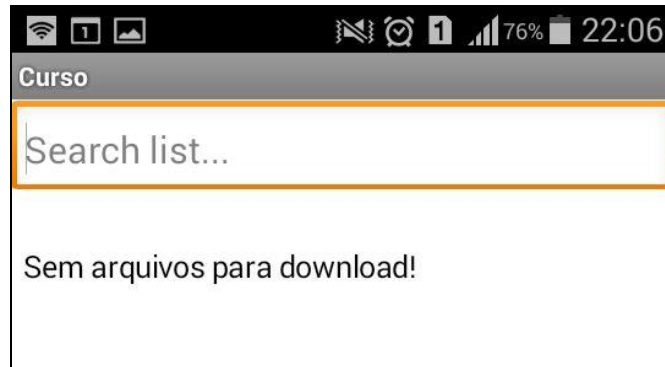
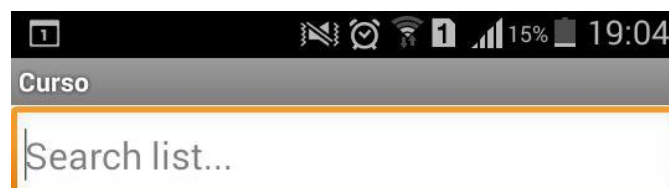


Figura 6 – Sem arquivos para download



Simulado

Figura 7 - Arquivo para download

Conforme possível identificar na Figura 5 as disciplinas matriculadas, ao pressionar em uma das disciplinas da lista, o aplicativo irá exibir se existe arquivos para ser baixados Figura 7, ou não, como na Figura 6.

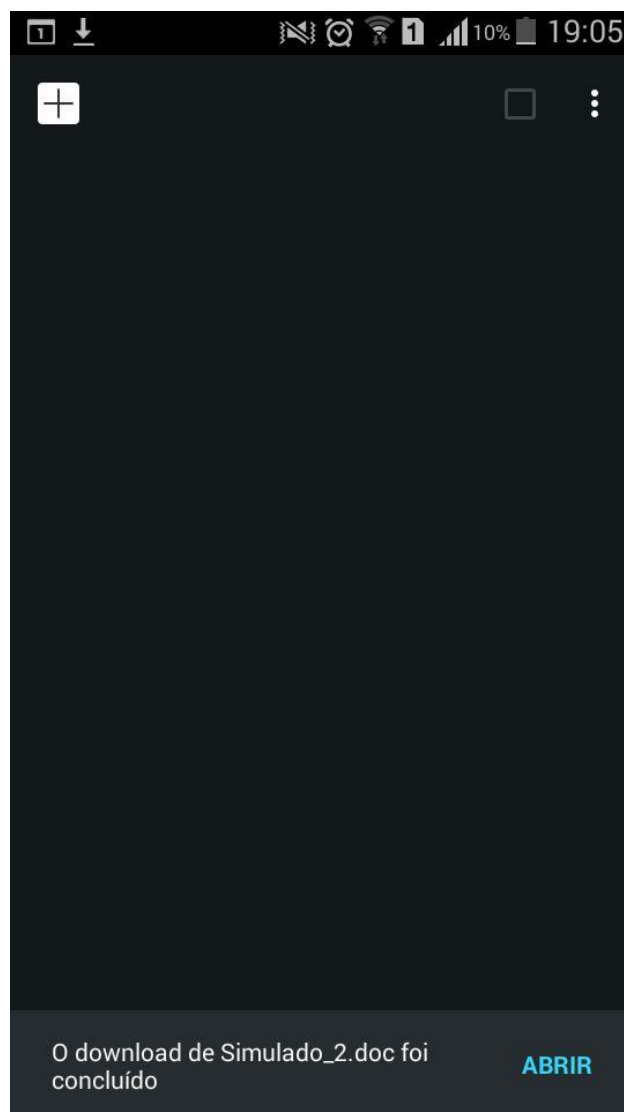


Figura 8 - Download Realizado

Considerando que existe um arquivo para ser baixado na disciplina selecionada, ao pressionar sobre o nome do arquivo, o mesmo será baixado. Dessa forma é possível que o aluno visualize esse arquivo em seu *tablet* ou *smartphone*.

4.3 IMPLEMENTAÇÃO DO SISTEMA

A maioria dos aplicativos Android são desenvolvidos utilizando a linguagem de programação Java, utilizando um IDE denominado Android Studio. Isso ocorre quase que por um padrão, por ser indicado pela Google, empresa responsável pelo sistema operacional.

Porém para se desenvolver um aplicativo Android utilizando Java, é necessário dominar de forma intermediária ou avançada a linguagem, além de requisitar um computador com recursos mais avançados para a utilização da IDE.

Em contrapartida o MIT desenvolveu um software para que qualquer pessoa com um conhecimento básico de programação, consiga desenvolver um aplicativo Android, esse software é o App Inventor.

Considerando todas essas vantagens que o App Inventor oferece, motivaram a escolha do mesmo para o desenvolvimento do aplicativo proposto nesse trabalho.

Utilizando o App Inventor as linhas de códigos, viram blocos encaixados, ou componentes prontos, que bastam ser configurados, conforme figuras abaixo.

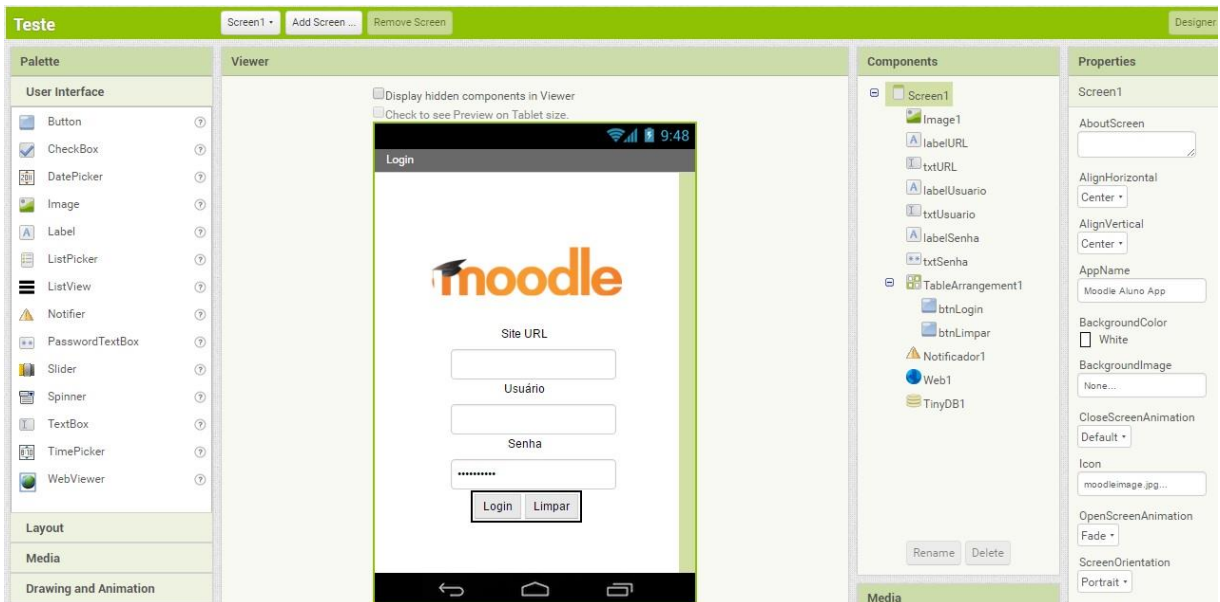


Figura 9 - App Inventor (desing de interface)

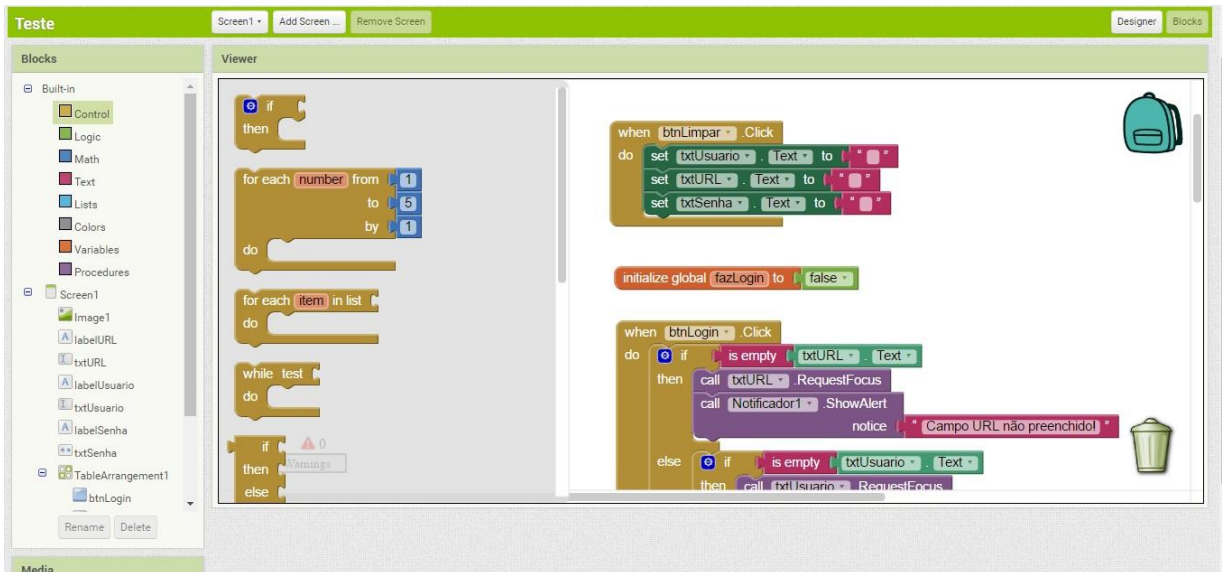


Figura 10 - App Inventor (blocos de código)

É possível perceber na Figura 5, o desenvolvimento da tela de *login* do aplicativo, onde os componentes que já existem (a esquerda), são simplesmente arrastados para compor a tela.

Na Figura 6, é possível ver os blocos de código, ou seja, a parte lógica por trás da tela de *login*. Neste caso, os blocos de códigos como “If then”, já existem e também são arrastados e encaixados ao resto do código.

O trecho de código possível de perceber, o formato de organização do código criado no App Inventor, e a facilidade em desenvolver.

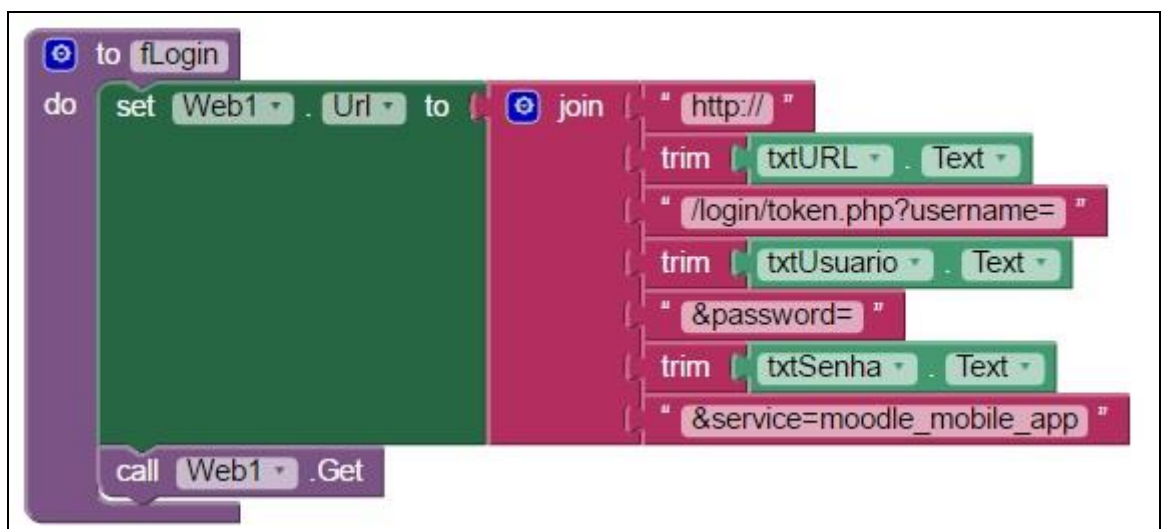


Figura 11 - Chamada Web Service

Acima, é possível verificar na Figura 7, a função (ou procedure) responsável pelo acionamento do *web service*. Nota-se a utilização de REST, onde é criada uma

URL, contendo todos os parâmetros requisitados. O *web service* chamado, é o responsável por validar se o *login* informado no aplicativo está existe e está correto.

4.4 DISCUSSÕES

O aplicativo desenvolvido como objeto deste trabalho, inicialmente destaca-se por ser desenvolvido com o App Inventor, uma ferramenta extremamente didática que permite que qualquer iniciante em programação, consiga desenvolver seu aplicativo.

Outro fator de destaque em relação ao aplicativo desenvolvido, que se difere dos outros disponíveis, como o aplicativo oficial do Moodle para *smartphones*, é o fato de ser um aplicativo desenvolvido exclusivamente para alunos, e que a rotina central, é o *download* de arquivos do Moodle.

Assim com o aplicativo desenvolvido, no propósito de facilitar o acesso dos alunos a informação, espera-se que os alunos tenham maior acesso a informação. Deste modo, o limitador de não ter acesso ao conteúdo no smartphone, será resolvido.

Com a informação mais acessível, nos seus dispositivos móveis, os alunos conseguem otimizar o tempo, e ler arquivos disponibilizados pelos professores, enquanto estão no transporte, enquanto estão em uma fila.

O aplicativo deve servir de incentivo para facilitar o uso do Moodle, e servir de estímulo para que os professores disponibilizem conteúdo, por saberem que os alunos terão acesso facilidade a essa informação, o que não é tão simplificado via navegador.

A necessidade de leitura por parte de alunos já é conhecida, e com o uso do aplicativo, o objetivo é promover maior conhecimento, ou de fato acesso simplificado ao conhecimento.

4.5 DIFICULDADES ENCONTRADAS

É esperado que todo trabalho de pesquisa e desenvolvimento tenha suas dificuldades. Quando se trata de um trabalho no qual o foco é a integração de dois

sistemas que funcionam em servidores diferentes, com arquiteturas diferentes, se comunicarem, as dificuldades podem ser ainda maiores.

4.5.1 Integração e Infraestrutura

A integração entre esses softwares é realizada por serviços web, os quais foram a maior dificuldade. Ao tentar utilizar o ambiente do Moodle de produção da UTFPR Campus Francisco Beltrão (www.moodle2.fb.utfpr.edu.br), testando as conexões conforme a documentação disponível no site oficial do Moodle (www.moodle.org) retornavam erros, geralmente de tempo de conexão atingido. Neste ponto é importante ressaltar que a documentação para as conexões via web services e API disponíveis no site do Moodle é muito genérica e incompleta o que também gerou atraso para esse trabalho.

Entretanto, após vários testes realizados, foi constatado que por motivos de segurança alguns dos *web services* necessários para o aplicativo desenvolvido, como o de listar disciplinas por aluno, estavam desativados no Moodle da UTFPR, portanto não era possível o acesso.

4.5.2 Moodle

Para contornar o problema dos *web services* do Moodle da UTFPR estarem desativados, foi então instalada a versão do Moodle em uma estação Windows, para servidor de um ambiente local. Após a instalação do ambiente oficial para criar uma base específica para esse trabalho, ao iniciar o serviço do Moodle, um erro era exibido no navegador informando que a página estava sendo redirecionada e gerando um loop.

Neste momento, foi muito importante o auxílio da comunidade Moodle. Através de fóruns, usuários postam suas dúvidas e são auxiliados, por outros usuários. Desta forma foi possível encontrar a solução para o problema de *loop*.

O ambiente Moodle por padrão após instalado, tem uma configuração que acaba ocasionando esse loop. Com o problema de redirecionamento corrigido, foi possível então configurar o ambiente e habilitar os *web services* necessários para o

desenvolvimento do trabalho. Também foi possível habilitar as documentações oficiais da versão do Moodle instalada, onde é permitido ter acesso aos parâmetros necessários e uma documentação completa.



Figura 12 - Lista com métodos disponíveis

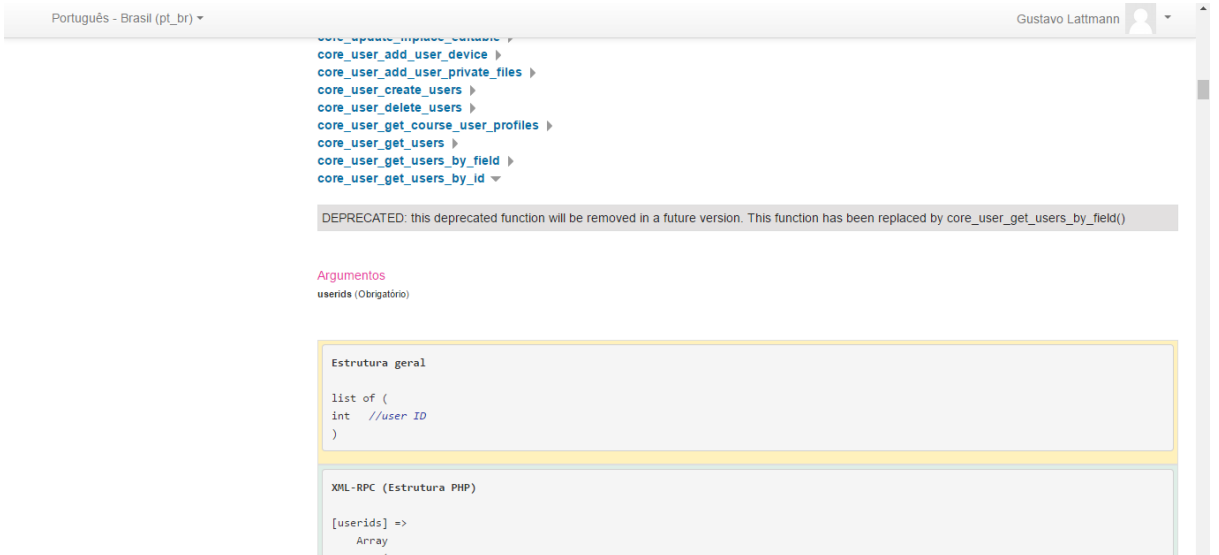


Figura 13 - Documentação Detalha Método

4.5.3 App Inventor

Com todos os problemas citados até então corrigidos, e sem nenhuma restrição por parte das integrações, foi dado início nos trabalhos na ferramenta App Inventor, utilizada para desenvolver o aplicativo. Embora a ferramenta App Inventor

seja muito rica, a complexidade da comunicação entre dois sistemas, utilizando web services, e retornos no formato JSON, começaram a surgir novas dificuldades.

Uma dessas dificuldades, foi tratar o retorno do *web service* do Moodle. Ao obter esse retorno do *web service*, no formato JSON, o App Inventor formata para uma espécie de lista, a qual foi de grande trabalho para manipular apenas os dados necessários, e modulá-los a ponto de serem exibidos de forma ordenada na tela.

5 CONCLUSÃO

Com a evolução da educação a distância e da tecnologia, os ambientes virtuais de aprendizagem começaram a estarem integrados a EAD, por proporcionarem realmente um ambiente completo, onde o professor consegue romper as barreiras da sala de aula, consegue romper as limitações geográficas.

Porém nem sempre os ambientes virtuais de aprendizagem conseguem acompanhar o avanço da tecnologia, que caminha para a internet das coisas, e para o mundo Mobile. Esse é o caso do Moodle, um dos ambientes virtuais de aprendizagem mais conhecidos, mas o qual seu aplicativo para smartphones deixa a desejar.

Deste modo é possível que o aluno consiga acessar o ambiente, baixar os arquivos disponíveis para estudar em seu celular, de maneira muito mais simplificada e com uma interface adaptada para seu dispositivo.

Por fim, para obter-se resultados excelentes não cabe apenas ao aplicativo, mas sim um trabalho de planejamento por parte de professores e o apoio dos alunos. É fato que com o aplicativo o acesso a informação ficou muito fácil, o mesmo tem como objetivo proporcionar conhecimento basta ser operado de forma concisa.

REFERÊNCIAS

APP INVENTOR, Sítio. 2015. **Sobre Nós**. Disponível em <http://appinventor.mit.edu/explore/about-us.html>. Acesso em 10 mar 2016.

ALVES, J. R. M. A história da EAD no Brasil. In: _____ (Org.). **Educação a Distância: o estado da arte**. São Paulo, 2009, p. 9-13.

BRASIL, MEC. **Relatório Educação para Todos no Brasil 2000-2015: versão preliminar**. Disponível em: http://portal.mec.gov.br/index.php?option=com_docman&task=doc_download&gid=15774&Itemid=. Acesso em 11 mai 2015.

FIELDING, Roy Thomas. 2000. **Architectural Styles and the Design of Network-based Software Architectures**. Disponível em: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm. Acesso em 28 mar 2016.

GARTNER, Inc. **Gartner Says Smartphone Sales Surpassed One Billion Units in 2014**. 2015. Disponível em: <http://www.gartner.com/newsroom/id/2996817>. Acesso em 10 mai 2015.

JR., C. F. A.; MARQUESI, S. C. Atividades em ambientes virtuais de aprendizagem: parâmetro de qualidade. In: _____ (Org.). **Educação a Distância: o estado da arte**. São Paulo, 2009, p. 358-368.

JSON, Sítio. **Introdução ao JSON**. 2016. Disponível em <http://www.json.org/json-pt.html>. Acesso em 24 jun 2016.

MOODLE, Sítio. **Instalando Moodle**. 2015. Disponível em https://docs.moodle.org/29/en/Installing_Moodle#Requirements. Acesso em 11 mai 2015.

NUNES. I. B. A história da EAD no mundo. In: LITTO, F. M.; FORMIGA, M. (Org.). **Educação a Distância: o estado da arte**. São Paulo, 2009, p. 2.

SOMMERVILLE, Ian. **Engenharia de software**. 9. ed. São Paulo, SP: Pearson Prentice Hall, 2011.

TANENBAUM, Andrew S.; STEEN, Maarten Van. **Sistemas distribuídos: princípios e paradigmas**. 2. ed. São Paulo: Pearson Prentice Hall, 2007.

UNESCO. 2015. **Mobile Learning**. Disponível em: <http://www.unesco.org/new/en/unesco/themes/icts/m4ed/>. Acesso em 11 mai 2015.

W3C. 2004. **Web Services Architecture**. Disponível em: <https://www.w3.org/TR/ws-arch/>. Acesso em 01 mai 2016.