

José Pedro Schardosim Simão

**RELLE: SISTEMA DE GERENCIAMENTO DE
EXPERIMENTOS REMOTOS**

Trabalho de Conclusão de Curso
submetido à Universidade Federal de
Santa Catarina para a obtenção do
Grau de Bacharel em Tecnologias da
Informação e Comunicação.
Orientador: Prof. Dr. Juarez Bento da
Silva

Araranguá
2015

Ficha de identificação da obra elaborada pelo autor
através do Programa de Geração Automática da Biblioteca Universitária
da UFSC.

Simão, José Pedro Schardosim

RELLE: Sistema de Gerenciamento de Experimentos Remotos /
José Pedro Schardosim Simão; orientador, Juarez Bento da Silva -
Araranguá, SC, 2015.

87 p.

Trabalho de Conclusão de Curso (graduação) - Universidade
Federal de Santa Catarina, Campus Araranguá. Graduação em
Tecnologias da Informação e Comunicação.

Inclui referências

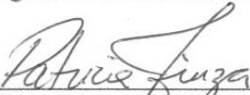
1. Tecnologias da Informação e Comunicação. 2. experimentação
remota. 3. sistema de gerenciamento. 4. acesso exclusivo. I. Silva,
Juarez Bento da. II. Universidade Federal de Santa Catarina.
Graduação em Tecnologias da Informação e Comunicação. III. Título.

José Pedro Schardosim Simão

**RELLE: SISTEMA DE GERENCIAMENTO DE
EXPERIMENTOS REMOTOS**

Este Trabalho de Conclusão de Curso foi julgado adequado para obtenção do Título de “Bacharel em Tecnologias da Informação e Comunicação”, e aprovado em sua forma final pelo Curso de Tecnologias da Informação e Comunicação.

Araranguá, 08 de dezembro de 2015.




Prof.^a Patricia Jantsch Fuza, Dr.^a
Coordenadora do Curso


Banca Examinadora:




Prof. Juarez Bento da Silva, Dr.
Orientador
Universidade Federal de Santa Catarina



Prof.^a Karine, dos Santos Coelho, M.^a
Escola de Educação Básica Apolônio Ireno Cardoso



Prof.^a Marta Adriana da Silva Cristiano, M.^a
Universidade Federal de Santa Catarina



Prof.^a Simone Meister Sommer Bilessimo, Dr.^a
Universidade Federal de Santa Catarina

Aos meus mestres, à minha família, e a todos que de alguma forma contribuíram na minha formação.

AGRADECIMENTOS

Agradeço primeiramente à minha família pela liberdade que sempre me foi dada e pelo apoio incondicional.

Ao meu orientador prof. Juarez, pelo apoio e pelas oportunidades que me proporcionou.

Aos colegas e amigos do Laboratório de Experimentação Remota, pelo companheirismo e trabalho em equipe.

A todos os meus professores, desde as séries iniciais, pelo incentivo a ir em busca dos meus sonhos.

À Universidade Federal de Santa Catarina, pela oportunidade de ter um ensino de qualidade sem ter de me deslocar a um grande centro.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior e ao programa Ciência sem Fronteiras por ter proporcionado uma experiência que mudou minha vida.

À Rede Nacional de Ensino e Pesquisa pelo apoio ao projeto.

Existem coisas melhores adiante do que qualquer
outra que deixamos para trás.
(C.S. Lewis)

RESUMO

Este trabalho trata do desenvolvimento de um sistema de gerenciamento de experimentos remotos, utilizando uma arquitetura baseada em módulos de hardware e software capazes de oferecer a usuários acesso remoto à laboratórios em diversas áreas do conhecimento. A solução foi concebida utilizando uma arquitetura baseada em soluções open source e de baixo custo, e que podem ser utilizados em diferentes níveis de ensino. A arquitetura é baseada em quatro módulos: o Laboratório Real, que trata dos experimentos, Computador Embarcado, que compreende a parte de disponibilização do experimento na rede, RELLE, que é o sistema de gerenciamento, e Cliente, que trata do modelo de acesso utilizado para manipulação dos experimentos. O trabalho foi desenvolvido em três fases nomeadas Planejamento, Protótipo, e Validação, baseadas no método Merise, que é uma metodologia que aborda todas as fases do desenvolvimento de sistemas de informação. O cliente para comunicação com os experimentos foi desenvolvido utilizando o framework PHP Laravel junto ao framework CSS Bootstrap. No desenvolvimento do cliente foi utilizada a metodologia ágil para desenvolvimento de software Scrum. O cliente pode ser acessado através de dispositivos móveis ou convencionais, e tem funções de gerenciamento de usuários e experimentos, além de integração com o ambiente virtual de aprendizagem Moodle, que fornece suporte documental e didático aos experimentos. O ambiente foi avaliado por especialistas da área de educação e tecnologias e experimentação remota, e foi aplicado em turmas experimentais no ensino básico e superior.

Palavras-chave: Laboratórios Remotos, Sistemas de Gerenciamento, E-Learning.

ABSTRACT

This work approaches the development of a remote labs management system using an architecture based on hardware and software modules, capable of offering users remote access to laboratories in various fields of knowledge. The solution was designed using an architecture based on open source and cost effective solutions, and can be used at different levels of education. The architecture is based on four modules: the Laboratory, referring to the experiments, Embedded Computer, providing the lab access through the internet, RELLE, which is the management system, and Client, about the access model used for the manipulation of experiments. The study was conducted in three phases named planning, prototype, and validation, based on Merise method, which is a methodology that addresses all phases of the development of information systems. The client for communication with the experiments was developed using the PHP framework Laravel and the CSS framework Bootstrap. It was used Scrum agile methodology on the client development. The client can be accessed through mobile or conventional devices, and has user management functions and experiments, as well as integration with the virtual learning environment Moodle, which provides documentary and educational support to the labs. The environment was evaluated by experts from the field of technology in education and remote experimentation, and was applied in experimental classes in primary, secondary and higher education.

Keywords: Remote Labs, Management Systems, E-Learning.

LISTA DE FIGURAS

Figura 1 - Modelos de funcionamento de laboratórios remotos.....	24
Figura 2 - Interface do osciloscópio no VISIR	26
Figura 3 - Protocolos utilizados em laboratórios gerenciados	28
Figura 4 - Arquitetura dos laboratórios gerenciados	29
Figura 5 - Arquitetura dos laboratórios não gerenciados	30
Figura 6 - Arquitetura do iLabs	32
Figura 7 - Interface dos Experimentos no Sahara	33
Figura 8 - Ciclo de vida de um projeto na metodologia Scrum	36
Figura 9 - Página inicial do RELLE	39
Figura 10 – Sistema acessado através de um smartphone Android	42
Figura 11 - Arquitetura do serviço.....	44
Figura 12 - Página de autenticação do usuário.....	45
Figura 13 - Página do Painel de Controle	46
Figura 14 - Página 'Novo Usuário'.....	47
Figura 15 - Página 'Criar experimento' - Aba Descrição	48
Figura 16 - Página 'Criar experimento' – Aba Detalhes.....	49
Figura 17- Página de logs	50
Figura 18 - Esquema de comunicação cross domain no uso da API.....	51
Figura 19 - Página do Experimento “Painel Elétrico CC”	51
Figura 20- Fila para acesso ao experimento.....	52
Figura 21 - Diagrama de estado de máquina da fila de acesso ao experimento	52
Figura 22 - Acesso ao Experimento Quadro Elétrico	53
Figura 23- Relatório de prática experimental exportado em PDF.....	54
Figura 24 - Ambiente Virtual de Aprendizagem no RELLE	55
Figura 25 - Diagrama de componentes das interações entre os serviços do Go- lab e o Smart Device.....	56
Figura 26- Experimento Painel Elétrico CC	58
Figura 27 - Experimento Painel Elétrico CA	59
Figura 28 - Experimento Condução de Calor em Barras Metálicas	60
Figura 29 - Experimento Meios de Propagação de Calor	61
Figura 30 - Experimento Microscópio Remoto	62
Figura 31 - Experimento Plano Inclinado	63
Figura 32 - Experimento Disco de Newton	64
Figura 33 - Experimento Ambiente de Desenvolvimento para Arduino.....	65
Figura 34 - Experimento Efeito Fotoelétrico	66
Figura 35 - Dados do teste “1.a”. Destaque para o tempo de resposta (linha azul) e a quantidade de clientes mantidos (linha verde).....	67
Figura 36 - Dados do teste “1.b”. Demonstrando as requisições por segundo respondidas com sucesso (em verde) e timeouts (em laranja)	68
Figura 37 - Dados do teste “2.a”. Destaque para o tempo de resposta (linha azul) e a quantidade de clientes mantidos (linha verde).....	69
Figura 38 - Dados do teste “2.b”. Demonstrando as requisições por segundo respondidas com sucesso (em verde) e timeouts (em laranja)	70

Figura 39 - Dados do teste “3.a”. Destaque para o tempo de resposta (linha azul) e a quantidade de clientes mantidos (linha verde).....	71
Figura 40 - Dados do teste “3.b”. Demonstrando as requisições por segundo respondidas com sucesso (em verde) e timeouts (em laranja)	72
Figura 41 - Mapa de acessos ao RELLE.....	74

LISTA DE ABREVIATURAS E SIGLAS

ABNT – Associação Brasileira de Normas Técnicas
API – Application Programming Interface
ER – Experimentação Remota
GT-MRE – Grupo de Trabalho em Experimentação Remota Móvel
HTTP – Hypertext Transfer Protocol
IBGE – Instituto Brasileiro de Geografia e Estatística
IEEE – Institute of Electrical and Electronics Engineers
MERISE - Méthode d'Études et de Réalisation Informatique pour les Systèmes d'Entreprise
MIT - Massachusetts Institute of Technology
REST - Representational State Transfer
RExLab – Laboratório de Experimentação Remota
RELLE – Remote Labs Learning Environment
RFC – Request for Comments
RNP – Rede Nacional de Ensino e Pesquisa
RLMS – Remote Labs Learning Environment
SSH – Secure Shell
STEM – Science, Technology, Engineering and Mathematics
UML – Unified Modeling Language
VNC – Virtual Network Computing
W3C – World Wide Web Consortium
XML – Extensible Markup Language

SUMÁRIO

1	INTRODUÇÃO	15
1.1	CONTEXTO DA PESQUISA	16
1.2	JUSTIFICATIVA	17
1.3	MOTIVAÇÃO	18
1.4	PROBLEMA.....	19
1.5	OBJETIVOS	19
1.5.1	Objetivo geral.....	19
1.5.2	Objetivos específicos	19
1.6	ESTRUTURA DO TEXTO	20
2	REFERENCIAL TEÓRICO	21
2.1	LABORATÓRIOS VIRTUAIS E REMOTOS	21
2.1.1	Experimentação Remota	22
2.1.2	Experimentação Remota Móvel.....	22
2.2	MODELOS DE FUNCIONAMENTO DE LABORATÓRIOS REMOTOS	23
2.2.1	Acesso.....	24
2.2.2	Manipulação.....	25
2.2.3	Comunicação	26
2.3	SISTEMAS DE GERENCIAMENTO DE LABORATÓRIOS REMOTOS	27
2.3.1	WebLab-Deusto	27
2.3.2	iLab Shared Architecture (ISA)	31
2.3.3	Labshare Sahara	32
3	METODOLOGIA.....	35
3.1	CLASSIFICAÇÃO DA PESQUISA.....	35
3.2	ETAPAS DA PESQUISA.....	35
2.3.1	Planejamento	27
2.3.1	Protótipo	27
2.3.1	Validação	27

4	SOLUÇÃO DESENVOLVIDA	39
4.1	CONCEPÇÃO E TECNOLOGIAS UTILIZADAS.....	39
4.1.1.	Laravel.....	40
4.1.2.	jQuery	41
4.1.3.	Bootstrap	41
4.1.4.	Android WebView	42
4.1.5.	Node.js.....	42
4.1.6.	WebSocket	43
4.2	ARQUITETURA	43
4.2.1.	Módulo RELLE.....	44
4.2.2.	Módulo Cliente.....	50
4.2.3.	Módulo Computador Embarcado	55
4.2.4.	Módulo Laboratório Real.....	57
4.2.4.1.	Painel Elétrico CC	58
4.2.4.2.	Painel Elétrico CA	59
4.2.4.3.	Condução de calor em barras metálicas	59
4.2.4.4.	Meios de Propagação de Calor	60
4.2.4.5.	Microscópio Remoto.....	61
4.2.4.6.	Plano Inclinado	62
4.2.4.7.	Disco de Newton.....	63
4.2.4.8.	Ambiente para Desenvolvimento em Arduino.....	64
4.2.4.9.	Efeito Fotoelétrico	65
5	TESTES E VALIDAÇÃO.....	67
5.1	TESTES DE PERFORMANCE.....	67
5.2	ACESSOS.....	72
5.3	VALIDAÇÃO POR ESPECIALISTAS.....	75
5.4	VALIDAÇÃO EM CLASSES EXPERIMENTAIS	78
5.4.1.	Ensino Médio.....	78
5.4.2.	Ensino Superior.....	78
6	CONSIDERAÇÕES FINAIS.....	80

REFERÊNCIAS	81
APÊNDICE A – Formulário de Validação.....	85

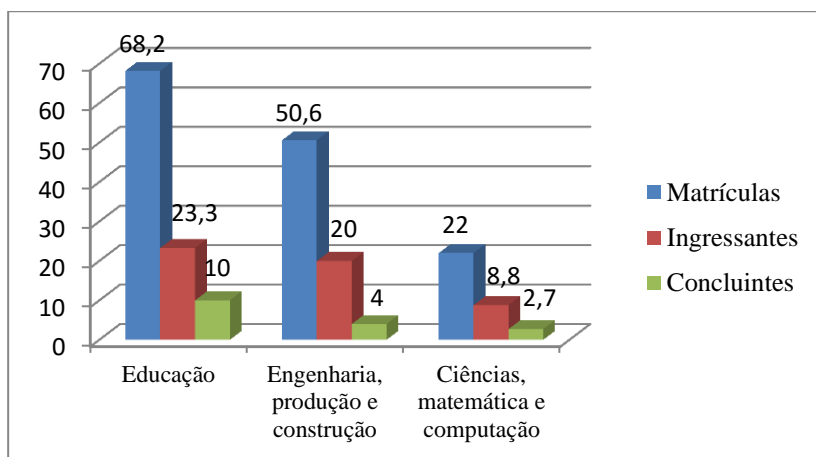
INTRODUÇÃO

É generalizada a preocupação com a escassez de profissionais nas áreas STEM. Segundo o relatório *STEM: Country Comparisons*, publicado pelo *Australian Council of Learned Academies* estas áreas têm sido preocupação central dos governantes e políticos em todo o mundo, pelo fato de gerarem benefícios econômicos diretos na medida em que ajudam a formar mão de obra qualificada. (ACOLA, 2013)

No Brasil, segundo o Conselho Federal de Engenharia, Arquitetura e Agronomia – Confea (2015), estima-se que há um déficit de cerca de 20 mil engenheiros por ano. Já quanto aos profissionais de tecnologia, a Associação Brasileira das Empresas de Tecnologia da Informação e Comunicação – Brasscom (2013) diz que o setor deverá criar mais de 750 mil novas vagas de trabalho no Brasil até 2020.

Esta realidade é ainda mais preocupante quando confrontada com os índices de matrícula e conclusão de cursos superiores no Brasil. Segundo o INEP (2014), o último censo da educação superior realizado em 2012 mostra que o número de concluintes em cursos das áreas STEM, para cada 10.000 habitantes, pode ser aproximadamente 12 vezes menor do que de alunos matriculados, conforme mostra o Gráfico 1.

Gráfico 1 - Matrículas, ingressantes e concluintes dos cursos de graduação no Brasil em 2012



Fonte: INEP (2014)

A carência na infraestrutura das instituições públicas de educação básica é um outro fator que pode afetar estes índices. Os dados do censo da educação básica brasileira, realizado pelo INEP (2015), apontam que no ano de 2014 as escolas contavam com a seguinte infraestrutura de suporte ao ensino:

- Internet (61%);
- Banda Larga (51%);
- Laboratórios de Informática (45%);
- Laboratórios de Ciências (11%);
- Média de computadores por escola para uso dos alunos (8.85).

Neste contexto, a experimentação remota pode ser utilizada como ferramenta de apoio ao ensino e aprendizagem nas disciplinas STEM, como forma de amenizar os problemas de infraestrutura.

1.1 CONTEXTO DA PESQUISA

O papel desempenhado pelas atividades práticas no ensino de ciências é de extrema importância. De acordo com Krasilchik (2000), o trabalho de laboratório motiva a aprendizagem, levando ao desenvolvimento de competências técnicas e, sobretudo, ajudar a proteger o conhecimento sobre os fenômenos.

Atividades práticas são conhecidas por ser um método eficaz no processo de ensino e aprendizagem, proporcionando situações de pesquisa e aumentando o interesse do aluno para a formação e aquisição de conhecimentos, e têm sido identificadas como um elemento importante de cursos de graduação em diversas áreas, especialmente nas engenharias e ciências aplicadas (SILVA *et al.*, 2014).

Experimentação e aprendizagem baseado na experiência são utilizados também em outras áreas, como por exemplo em economia, onde os alunos criam empresas fictícias para competir em um mercado simulado (AUER *et al.*, 2003). No entanto, a realidade das práticas laboratoriais foi profundamente alterada pelas Tecnologias de Informação e Comunicação. Ferramentas como simulações, realidade virtual e laboratórios remotos têm sido usados para fazer aulas mais interativas. Mesmo laboratórios tradicionais são, na maioria das vezes, mediados por computadores e outros dispositivos digitais.

Com o aumento da disponibilidade de infraestrutura avançada de telecomunicações e de acesso associado a aplicações baseadas na

Internet, Lowe *et al.* (2009) aponta um aumento recente no desenvolvimento de laboratórios remotos.

O acesso e manipulação de experimentos remotamente se diferencia de outras tecnologias empregadas no ensino de ciências como simulações, uma vez que o resultado obtido em cada experiência é real, sendo influenciado por uma série de variáveis de ambiente normalmente ignoradas pelas simulações.

A utilização de experimentos pode proporcionar a redução de custos dos recursos laboratoriais, uma vez que o mesmo experimento pode ser acessado por diversas escolas, em diversas localidades. Além disso, as configurações de equipamento necessárias para o acesso aos experimentos são mínimas, sendo o mais importante o acesso à internet.

Existem laboratórios remotos de diversos tipos e em diversas áreas, como física, química, eletrônica e robótica, e apesar de a grande maioria dos experimentos serem voltados para o ensino superior, existem algumas iniciativas para a educação básica. A utilização de laboratórios remotos pode ser uma alternativa aos problemas de infraestrutura das instituições públicas brasileiras de ensino básico.

1.2 JUSTIFICATIVA

Ao implantar um laboratório remoto, tem-se uma série de alternativas de sistemas de gerenciamento disponíveis, estas sendo geralmente gratuitas e de código aberto. Porém estas são em sua maioria soluções de difícil personalização, e não estão preparadas para diferentes conjuntos de experimentos, já que cada arquitetura de laboratório tem diferentes necessidades.

Por mais que hajam iniciativas que almejam a criação de um padrão para a implementação de experimentos remotos, nenhuma foi concretizada. Pode ser destacada a iniciativa do grupo de trabalho do *Institute of Electrical and Electronics Engineers* (IEEE) IEEE-SA P1876¹ (*Standard on Networked Smart Learning Objects for Online Laboratories*), que pretende criar um padrão para laboratórios remotos, porém o documento ainda está em fase de desenvolvimento.

Ainda assim, pode-se verificar que todos os modelos de gerenciamento de laboratórios remotos tratam de alguma forma o acesso múltiplo. Os problemas gerados pela interferência de terceiros na execução do experimento comprometem a usabilidade e eficiência da

¹ <https://iee-SA.imeetcentral.com/1876public/>

ferramenta. As alterações na plataforma de acesso aos experimentos do RExLab são necessárias, visto um recente aumento no número de usuários e a ausência de funcionalidades mínimas de controle na arquitetura anteriormente utilizada.

1.3 MOTIVAÇÃO

O Laboratório de Experimentação Remota (RExLab) da Universidade Federal de Santa Catarina provê acesso remoto a experimentos desde 1997. O primeiro experimento desenvolvido no laboratório, um depurador remoto para o microcontrolador 8051, foi descrito por Predko (1999, p. 515, tradução do autor) naquele momento como “um dos usos mais interessantes da internet”.

Anteriormente a este trabalho, a arquitetura dos experimentos disponíveis no laboratório era de acesso direto, onde o usuário controla o experimento através de uma página web enviando comandos para o dispositivo físico. Considerando a ausência de qualquer forma de autenticação, fila ou agendamento, o múltiplo acesso era um problema que poderia interferir na utilização dos experimentos do RExLab.

Este cenário foi agravado pelo recente aumento no número de usuários da plataforma. O RExLab, por meio do projeto “Proposta de integração tecnológica no ensino de disciplinas STEM na Educação Básica da rede pública” atende 4 escolas públicas na região do Vale do Araranguá, Santa Catarina, Brasil, com um total de 287 professores e 3.201 alunos, oferecendo suporte à utilização de experimentos remotos e ambientes virtuais de aprendizagem na educação básica.

Este trabalho se dá no âmbito do projeto “Grupo de Trabalho em Experimentação Remota Móvel” (GT-MRE), financiado pela Rede Nacional de Ensino e Pesquisa (RNP), que em seus objetivos se propõe desenvolver e implantar uma plataforma que consiste em um ambiente virtual de ensino e aprendizagem integrado a experimentos remotos, acessada utilizando dispositivos móveis. O autor deste trabalho atuou como estagiário na execução deste projeto, desenvolvendo o sistema de gerenciamento.

1.4 PROBLEMA

Muitos laboratórios remotos exigem o agendamento prévio para acesso aos experimentos, e durante o horário agendado, por mais que

ninguém esteja acessando, o experimento fica indisponível, que acarreta o desperdício de recursos.

A exigência de autenticação também é um problema, uma vez que a necessidade de cadastro dificulta o acesso, e torna o experimento invisível aos mecanismos de busca. A maioria dos sistemas de gerenciamento disponíveis não tem acesso com visitante implementado, ou utiliza soluções como usuário de demonstração.

O acesso múltiplo também pode comprometer a experiência do usuário quando executando um experimento remoto, já que pode alterar o resultado da experiência. Diante disso, a fila pode ser considerada a solução mais utilizada para evitar este tipo de problema, mas muitos laboratórios não têm esta função implementada, ou obrigam o usuário a agendar o experimento mesmo que seja para utilização naquele instante.

Além disso, o projeto GT-MRE tem especificidades em seus objetivos que demandam a criação de uma nova plataforma para atender aos usuários da RNP.

Portanto, identifica-se a seguinte pergunta de pesquisa:

“Quais as contribuições de um sistema de gerenciamento de experimentos remotos na experimentação remota?”

1.5 OBJETIVOS

1.5.1 Objetivo geral

Verificar as contribuições a partir do desenvolvimento e implantação de um sistema de gerenciamento de experimentos remotos baseado em blocos.

1.5.2 Objetivos específicos

1. Desenvolver uma interface de criação e utilização de experimentos remotos.
2. Implementar funcionalidades de criação e gerenciamento de usuários.
3. Criar uma interface de análise dos dados de utilização do sistema.
4. Realizar testes de performance e validação da plataforma.
5. Disponibilizar o código-fonte sob licença MIT em repositório público na internet.

1.6 ESTRUTURA DO TEXTO

Este documento, além desta introdução, está organizado em mais 5 (cinco) capítulos que abordam os seguintes conteúdos:

O Capítulo 2 faz um apanhado sobre conceitos básicos da experimentação remota, e sobre modelos de acesso, manipulação e comunicação de experimentos remotos. Este capítulo também trata dos sistemas de gerenciamento de experimentos remotos, trazendo os sistemas mais usados, suas arquiteturas e funcionalidades.

O Capítulo 3 descreve a metodologia utilizada no desenvolvimento e as ferramentas utilizadas na implementação do sistema descrito neste trabalho.

O Capítulo 4 traz a descrição do sistema desenvolvido, a modelagem do sistema, a concepção e funcionalidades.

O Capítulo 5 apresenta os resultados obtidos nos testes de performance e na validação da plataforma através de pesquisa feita com especialistas e aplicações em turmas experimentais.

O Capítulo 6 busca apresentar as considerações finais, a fim de se verificar se os objetivos propostos foram alcançados e indicar aspectos a serem consideradas em trabalhos futuros.

2 REFERENCIAL TEÓRICO

2.1 LABORATÓRIOS VIRTUAIS E REMOTOS

Segundo Silva (2006) o desenvolvimento de laboratórios para práticas baseados na Web recebeu um forte impulso a partir dos anos noventa, e atualmente são encontrados em centros como o MIT nos Estados Unidos e a Universidad de Siena na Itália, entre outros. Com a ascensão da Internet em popularidade, estes laboratórios estão cada vez mais ganhando aceitação para a utilização em sala de aula, bem como no ensino à distância, nas áreas de engenharia e ciências (YAOYE et al., 2008). O conceito de laboratórios web abrange laboratórios virtuais e laboratórios online (ou remotos).

De acordo com Scheckler (2003), laboratórios virtuais “usam modelos computadorizados e simulações para substituir atividades de laboratório presenciais”. Já os laboratórios remotos são definidos por Silva (2006) como um conjunto de recursos compartilhados em rede com a finalidade de permitir que os usuários possam pôr em prática o monitoramento dos experimentos e a gestão destes recursos, os conhecimentos adquiridos nas aulas das instituições de ensino sem ter que contar com material sofisticado ou com componentes caros e difíceis de serem obtidos. Auer et al. (2003) classificam os laboratórios quanto à localização do aluno e do experimento (Quadro 1).

Quadro 1 - Classificação dos laboratórios

		Aluno	
		Local	Remoto
Experimento	Real	Laboratório Tradicional (<i>hands-on</i>)	Laboratório Remoto
	Virtual	Simulação Local	Laboratório Virtual

Fonte: AUER et al. (2003)

2.1.1 Experimentação Remota

Um laboratório remoto é uma solução de software e hardware que permite que os alunos acessem equipamentos fisicamente localizados em uma universidade, escola ou centro de pesquisa. O usuário manipula as variáveis do experimento, pode observar os fenômenos através de streaming de vídeo, e recebe o resultado (MARCELINO, 2010).

A experimentação remota se difere de laboratórios baseados em simuladores, uma vez que estes só retornam valores e resultados previamente calculados, deixando de considerar fatores importantes como as variações no ambiente, contemplados pelos experimentos reais.

Silva (2006) afirma que as principais vantagens da experimentação remota são:

- A aprendizagem a distância por parte dos alunos sem restrições de tempo ou limitações de distância.
- A utilização por parte dos alunos de materiais caros aos quais não teriam acesso no laboratório tradicional.
- O compartilhamento de equipamentos entre laboratórios de diferentes universidades a nível nacional ou internacional.

Segundo Rochadel (2013), a experimentação remota é frequentemente utilizada para complementar as aulas expositivas nas áreas das ciências, tecnologia e engenharias no ensino superior, pois possibilita aos estudantes observarem fenômenos dinâmicos que são muitas vezes difíceis de explicar através de material escrito.

Um aspecto importante dos experimentos remotos descrito por Schaf (2006) é a flexibilização, tanto espacial, já que os alunos podem acessar os experimentos de outros lugares que não a escola, quanto temporal, já que o acesso pode ser realizado em qualquer horário.

2.1.2 Experimentação Remota Móvel

É possível explorar ainda mais os conceitos de experimentação remota e m-learning ao associá-los, somando os objetivos da experimentação remota aos recursos de mobilidade que estes têm a oferecer. Assim, a utilização de laboratórios remotos em dispositivos móveis torna-se parte de alternativas e métodos para atrair a atenção dos alunos e complementar aulas teóricas em disciplinas por muitas vezes julgadas difíceis e monótonas.

A expansão dos importantes recursos da experimentação remota nos dispositivos móveis propicia a imersão do usuário em um laboratório a qualquer momento e em qualquer lugar e utilizando o seu

próprio smartphone, deste modo, um novo espaço de interação que alia justamente a ideia de ambientes ubíquos (ambientes onde a tecnologia está presente em todo lugar).

Costa e Alves (2006) afirmam que a experimentação móvel garante o benefício concebido pela experimentação remota (ER) para a engenharia e ciências em contextos de m-learning, ou seja, a flexibilidade, a mobilidade e motivação aos alunos. Hoje em dia a qualidade de serviço de redes sem fio está em constante expansão e grande parte possuem um dispositivo móvel, o que garante a facilidade de acesso de novas aplicações desenvolvidas com base no conceito de experimentação móvel.

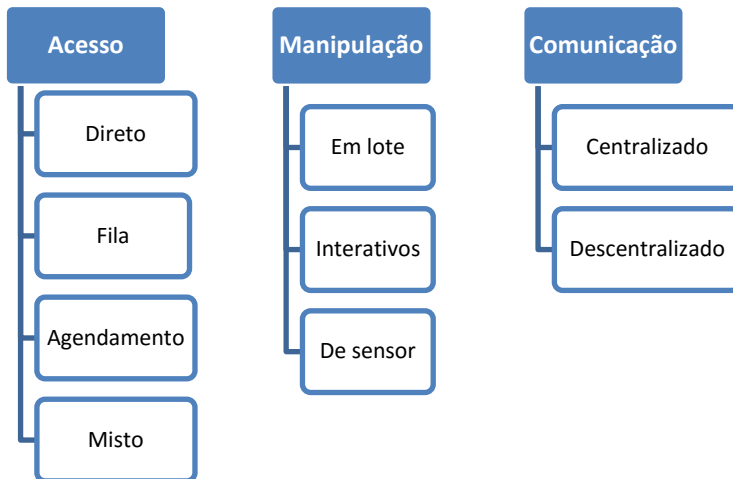
2.2 MODELOS DE FUNCIONAMENTO DE LABORATÓRIOS REMOTOS

Existem diversos modelos de funcionamento de laboratórios remotos, que podem diferir quanto ao controle de acesso, manipulação dos experimentos e comunicação. Dentre os fatores que devem ser levados em consideração na escolha de um modelo para a implementação de experimentos remotos, Lowe *et al.* (2011) destaca os seguintes:

- Riqueza de controle: um experimento controlado diretamente muitas vezes pode introduzir mais recursos e interface de controle mais personalizada.
- Aplicação de controle para o hardware já existente: se a aplicação de controle de plataforma já existe, é interessante utilizar controle descentralizado.
- Flexibilidade: se os requisitos são flexíveis, um software de controle mais geral pode ser utilizado. Requisitos mais especializados podem exigir um controlador direto a ser desenvolvido para atender às exigências.
- Infraestrutura: a estrutura existente e suas limitações (como plataformas suportadas e restrições de firewall) podem influenciar e limitar a escolha do modelo de funcionamento.
- Facilidade de modificações: um equipamento controlado diretamente separa a implementação de controle de hardware e a interface de usuário, o que permite que a interface do cliente seja facilmente alterada, sem ter de mudar o controle de hardware.

Neste trabalho, os diferentes modelos de funcionamento de laboratórios remotos serão tratados através da divisão em modelos de manipulação, acesso e controle, conforme

Figura 1 - Modelos de funcionamento de laboratórios remotos



Fonte: Autor

2.2.1 Acesso

O controle de acesso é uma das funcionalidades mais importantes de um sistema de gerenciamento de experimentos remotos, uma vez que é responsável por garantir que o experimento seja manipulado por apenas um usuário de cada vez.

Nas arquiteturas que utilizam acesso direto, o usuário acessa o experimento sem necessidade de autenticação. Este modelo raramente é utilizado, pois pode gerar uma série de problemas se funcionalidades de acesso exclusivo não forem implementadas. Um usuário, ao acessar um experimento durante uma sessão em andamento, pode manipular as variáveis alterando o resultado da experiência de outro usuário.

Um modelo vastamente utilizado é o de agendamento, onde o experimento é alocado para o usuário que o agendou durante certo período de tempo, e durante este tempo somente ele poderá acessá-lo (LOWE E OROU, 2012). Se não houver tratamento para quando o usuário que agendou o experimento não acessar o experimento no horário marcado, pode ocasionar no desperdício de recursos, já que o experimento ficará indisponível sem que ninguém esteja o acessando.

Outra solução é a utilização de filas, onde se o experimento que o usuário está tentando acessar estiver ocupado, ele terá de esperar até o fim da sessão em curso, ou até o usuário que está acessando o experimento desconectar. Há ainda a possibilidade de se utilizar um modelo misto, onde há a fila de acesso ao experimento e o agendamento, e se um usuário que agendou o experimento o acessa durante o período agendado, este passa a ser o primeiro da fila.

2.2.2 Manipulação

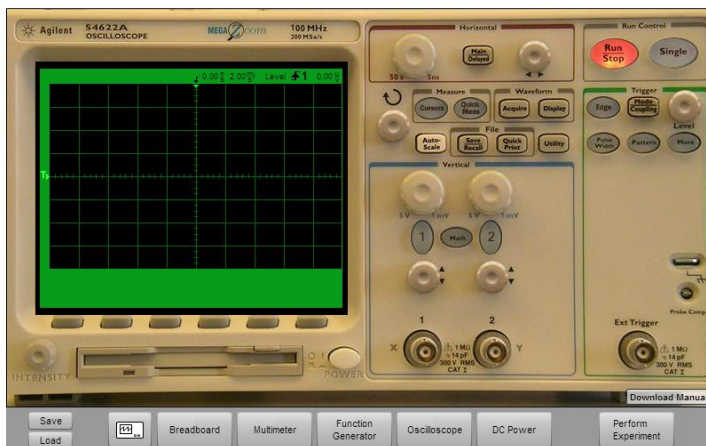
Segundo Harward *et al.* (2008), os experimentos são divididos em 3 tipos quanto à sua manipulação: experimentos em lote, experimentos interativos e experimentos de sensor.

Experimentos em lote são aqueles em que todo o decurso da experiência tem de ser especificado antes do seu início, como explanado por Harward *et al.* (2008). O usuário escolhe um set de configurações para o experimento, através de uma interface gráfica ou do upload de um arquivo. Após o envio dos parâmetros de execução, o experimento é executado, e retorna os resultados. Um exemplo é o experimento FPGA, descrito por Costa *et al.* (2012), onde o usuário envia um documento de extensão. vhd ou .vhdl que descreve o comportamento do hardware.

Já experimentos interativos são aqueles em que o usuário controla um ou mais aspectos da experiência durante a sua execução. Ao enviar um certo comando, o usuário altera uma variável do experimento, e pode visualizar as mudanças no fenômeno e resultado (HARWARD *et al.*, 2008).

Um exemplo clássico de experimento interativo é o VISIR (Figura 2), uma ferramenta que possibilita a criação da fiação e medição de circuitos elétricos e eletrônicos em uma placa de ensaio. Nestes experimentos o usuário faz o cabeamento em uma interface gráfica, escolhe os componentes, e um circuito físico é criado no laboratório. O usuário pode utilizar instrumentos como osciloscópio e multímetro para medição, e os valores reais são mostrados. Ainda é possível fazer upload de um arquivo com as configurações, ou ainda salvar o circuito criado em um arquivo (ODEH *et al.*, 2014).

Figura 2 - Interface do osciloscópio no VISIR



Fonte:

https://62.204.201.71/trunk/sites/electronics/public/experiment.php?sel=experiment_immediate&id=3

Os experimentos de sensor são aqueles onde o usuário apenas recebe, manipula e interpreta dados coletados por sensores. Nestes experimentos, não há manipulação de variáveis durante a execução da experiência. Um exemplo de experimento de sensores é uma estação meteorológica, onde os alunos podem verificar valores como velocidade e direção do vento, temperatura e umidade, porém não controlam nenhum atuador.

2.2.3 Comunicação

A comunicação trata dos métodos utilizados no transporte de informações entre o experimento e o cliente. No modelo de comunicação centralizado, os comandos enviados pelo usuário passam pelo servidor antes de serem executados pelo experimento. Normalmente baseados em uma API disponibilizada pelo RLMS, os comandos são tratados pelo servidor, que checa aspectos de segurança como, por exemplo, se quem está enviando os comandos é realmente quem diz ser, e se está autorizado.

Já no modelo de comunicação descentralizado, os comandos são enviados pelo cliente diretamente para o experimento, sem a intervenção de um servidor (*service broker*). Este modelo dispensa a utilização de

uma API específica para comunicação com o servidor, porém pode vir a apresentar problemas relacionados à segurança se verificações de identidade não forem realizadas pelo experimento (HARWARD *et al.*, 2008).

2.3 SISTEMAS DE GERENCIAMENTO DE LABORATÓRIOS REMOTOS

Apesar de existirem laboratórios com experimentos focando diferentes áreas, existem muitas similaridades quanto ao gerenciamento, já que estas características são independentes das especificações e funcionamento dos experimentos (ORDUÑA *et al.*, 2014).

De acordo com Orduña et al. (2014), para fins de replicação e aproveitamento de código, o desenvolvimento de laboratórios remotos foi dividido em dois blocos: os códigos do laboratório, que compreendem as funções utilizadas para a conexão com os experimentos e sua manipulação, e os códigos de gerenciamento, que são as ferramentas administrativas do laboratório. O sistema de gerenciamento de laboratório remoto, ou RLMS (do inglês *Remote Lab Management System*), é o software responsável pelo gerenciamento do laboratório e seus experimentos, e inclui funções como autenticação, agendamento e rastreamento de usuário.

Nas subseções abaixo, são apresentados os sistemas de gerenciamento de laboratórios remotos mais utilizados:

2.3.1 WebLab-Deusto

Desenvolvido na Universidade de Deusto, o WebLab-Deusto é um RLMS que provê uma API para criação de novos experimentos, autenticação, agendamento, rastreamento e gerenciamento de usuários, além de permitir o compartilhamento de experimentos entre diferentes instituições. O WebLab-Deusto permite o compartilhamento de laboratórios remotos fornecidos por outras universidades utilizando a plataforma, além de permitir que outras universidades ou escolas usem seus experimentos.

Este RLMS está disponível em uma série de línguas, incluindo português, inglês e espanhol, disponibiliza seu código na web sob licença BSD 2-Clause e possui documentação vasta², esta porém disponível apenas em inglês.

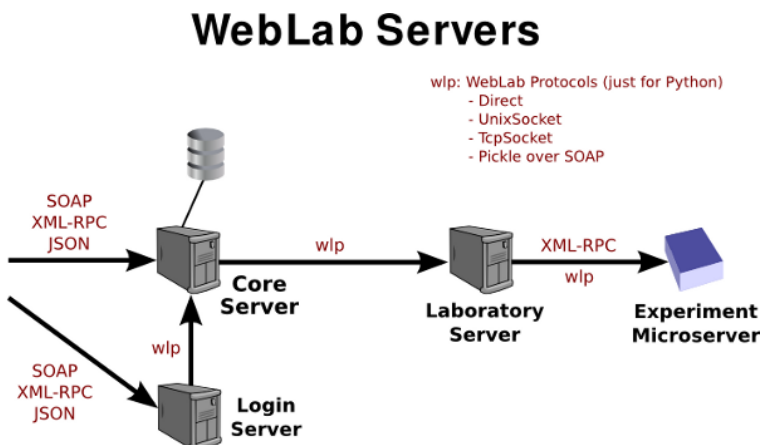
² <https://weblabdeusto.readthedocs.org/en/latest/>

Segundo Orduña *et al.* (2011), o modelo adotado por este RLMS é dividido em duas sub-arquiteturas: laboratórios gerenciados e não gerenciados.

2.3.1.1 Laboratórios Gerenciados

Estes laboratórios são desenvolvidos utilizando a API do WebLab-Deusto. Eles têm basicamente duas partes: um cliente, desenvolvido utilizando uma das bibliotecas fornecidas por Weblab-Deusto, e um servidor, desenvolvido utilizando uma das bibliotecas de servidor fornecidas ou usando XML-RPC diretamente.

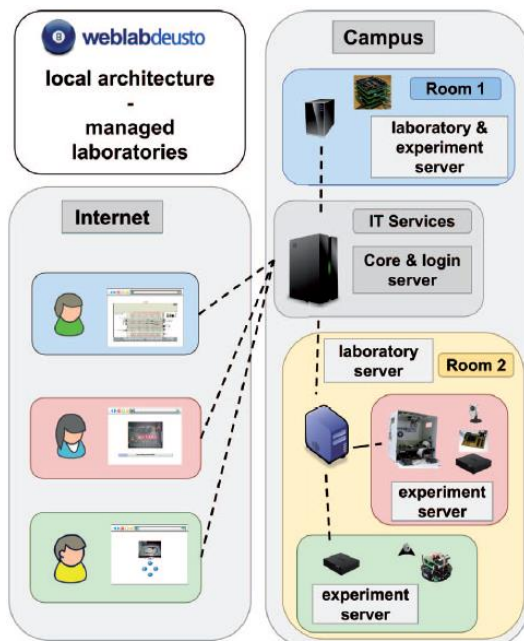
Figura 3 - Protocolos utilizados em laboratórios gerenciados



Fonte: <https://weblabdeusto.readthedocs.org/en/latest/technical.html>

Desta forma, o cliente será executado no navegador da Web e, basicamente, exibir a interface do usuário. Sempre que a interface de usuário requer acesso ao equipamento, ele irá usar a API fornecida para enviar um comando e obter uma resposta. Por exemplo, uma aplicação típica pode executar uma ação quando o usuário pressiona um botão. Este botão irá enviar uma mensagem de comando usando a API, e o RLMS vai chamar um método em particular no lado do servidor com essa mensagem particular.

Figura 4 - Arquitetura dos laboratórios gerenciados



Fonte: ORDUÑA *et al.*, 2014

Nos laboratórios gerenciados, é utilizada uma arquitetura cliente-servidor, na qual o cliente é responsável pela interação com o usuário e o envio de comandos para o servidor, onde estes comandos serão utilizados no controle do hardware. A comunicação entre o cliente e o servidor passa pelo servidor do WebLab-Deusto, onde são realizadas verificações de segurança.

As principais vantagens dos laboratórios gerenciados são:

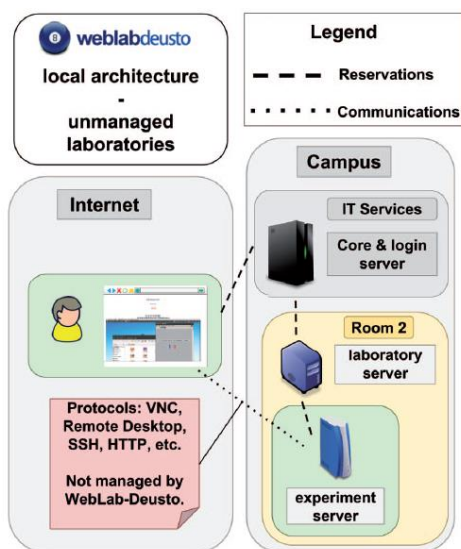
- O desenvolvedor dos experimentos não tem de se preocupar com a comunicação, uma vez que esta é feita pelos servidores do WebLab-Deusto.
- Todas as informações de acesso são armazenadas em um banco de dados, o que possibilita a análise das informações de aprendizagem. Os professores têm acesso aos comandos enviados pelos alunos e os resultados obtidos no experimento.

- Laboratórios gerenciados suportam balanço de carga, o que permite que os laboratórios tenham múltiplas instâncias de um mesmo experimento.

2.3.1.2 Laboratórios Não Gerenciados

Nos laboratórios não gerenciados, a comunicação entre cliente e servidor se dá de forma direta, sem a intervenção do RMLS. Os comandos são enviados do cliente diretamente para o servidor final.

Figura 5 - Arquitetura dos laboratórios não gerenciados



Fonte: ORDUÑA *et al.* (2014)

Os principais exemplos de laboratórios não gerenciados são experimentos baseados em máquinas virtuais, onde uma máquina virtual é utilizada para manipular o experimento. Ferramentas como Virtual Box e VMware são utilizadas para criar a máquina virtual, que pode rodar um sistema Linux ou Windows, sendo acessado através de SSH, VNC ou Remote Desktop. Outro exemplo são os experimentos baseados na plataforma LabVIEW. Apesar do WebLab-Deusto suportar experimentos criados utilizando LabVIEW, esta funcionalidade ainda está instável.

As principais desvantagens dos laboratórios não gerenciados são:

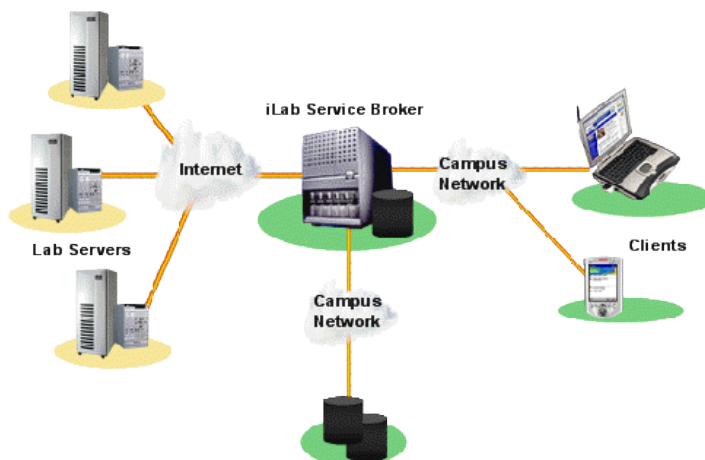
- Eles podem causar problemas com proxies ou firewalls, já que a comunicação é gerenciada pelo sistema final.
- A funcionalidade de rastreamento de usuários é reduzida: ainda são registrados quais recursos o aluno utiliza, e quando estes recursos são utilizados, mas não tudo o que o usuário fez durante a sessão.
- A funcionalidade de balanceamento de carga é reduzida ou até mesmo ausente.
- Tem a implementação mais complexa que laboratórios gerenciados.

2.3.2 iLab Shared Architecture (ISA)

O iLab Shared Architecture foi desenvolvido pelo MIT, e disponibiliza laboratórios remotos em microeletrônica, engenharia química, cristalização de polímeros, engenharia estrutural e de processamento de sinal, e sua arquitetura se divide em três partes distintas: servidor, cliente e *service broker*.

O servidor está conectado ao equipamento no laboratório e lida com o funcionamento real do equipamento. O cliente, instalado no computador do usuário, é a interface para o funcionamento do laboratório. Já o *service broker* media as comunicações entre o servidor e o cliente através da Web e fornece serviços como autenticação e rastreamento de usuários (ORDUÑA *et al.*, 2014).

Figura 6 - Arquitetura do iLabs



Fonte: <https://wikis.mit.edu/confluence/display/ILAB2/Home>

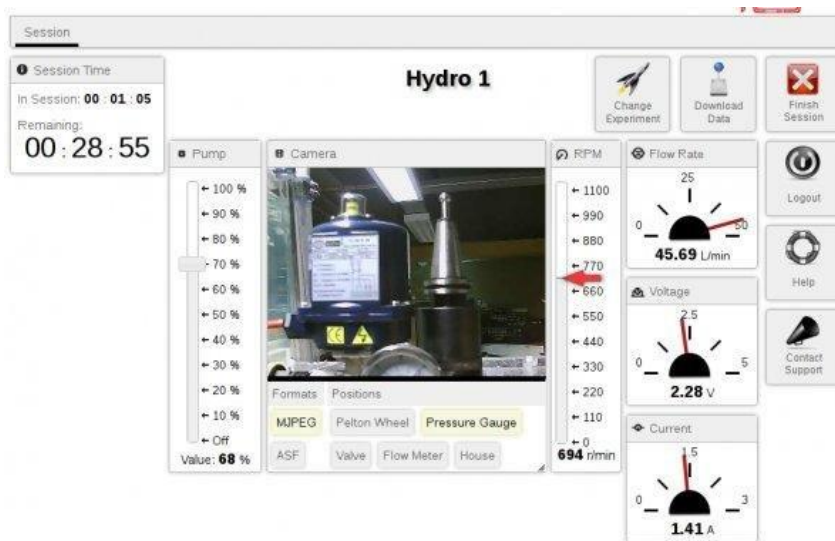
Nos experimentos em lote, o usuário configura os comandos a serem mandados ao experimento, e recebe os resultados quando o experimento estiver disponível para executar sua solicitação. Nos experimentos interativos o usuário interage com o experimento, controlando variáveis e monitorando os resultados durante a execução. Já nos experimentos de sensores, o usuário analisa um fluxo de dados em tempo real³ sem influenciar nos fenômenos sendo mensurados.

2.3.3 Labshare Sahara

Desenvolvido pela Universidade Tecnológica de Sydney (UTS) a partir do consórcio Labshare, o Sahara é um sistema que permite o acesso a experimentos remotos. O usuário, após autenticação, tem acesso a experimentos durante um determinado tempo (sessão). Além disso, o Sahara oferece funcionalidades de agendamento e suporte à múltiplas instâncias de um experimento.

³ <https://wikis.mit.edu/confluence/display/ILAB2/Developers>

Figura 7 - Interface dos Experimentos no Sahara



Fonte: <http://sourceforge.net/projects/labshare-sahara/>

Segundo Lowe *et al.* (2011), o Sahara se divide em três modelos de acesso: controle periférico, controle direto (ou primitivo), e controle em lotes.

2.3.3.1 Controle Periférico

No controle periférico o acesso ao experimento por uma interface externa ao Sahara e o sistema não tem controle sobre a comunicação com o hardware, mas é responsável por atribuir o acesso ao programa de controle para o usuário, revogando o acesso, iniciando a experiência e monitorando os resultados dos testes e detecção de atividade na plataforma.

Este tipo de controle é recomendado para quando o controle da plataforma for previamente implementado, e alterar sua funcionalidade seria dispendioso em tempo e esforço. Além disso, é adequado quando o laboratório requer baixa latência para elementos gráficos, como gráficos que precisam ser atualizados com frequência (em milissegundos). Nestes casos, o tratamento pode ser feito por processadores de alta velocidade no próprio equipamento para evitar que pequenos atrasos que o Sahara pode ter.

Uma implementação típica deste tipo plataforma seria uma experiência que já funciona através de uma interface de computador (utilizando programas de controle como o LabVIEW), embora não de forma remota. Para adaptar-se que este seja um laboratório remoto, ele pode ser incorporado em Sahara, e o acesso pode ser dado ao atribuir ao usuário uma sessão de Área de Trabalho Remota para o controlador existente. Neste caso, o Sahara não tem conhecimento do controle para a plataforma (LOWE *et al.*, 2011).

2.3.3.2 Controle Direto ou Primitivo

No controle direto, a aplicação cliente é quem controla o hardware. Este é o modelo de controle mais adequado onde não exista um módulo de controle previamente desenvolvido, ou onde a baixa latência para elementos gráficos não é essencial. Também é adequado quando há restrições de segurança, já que o equipamento é acessado apenas através do servidor web Sahara e portas adicionais não precisam ser abertas.

Neste modelo, de acordo com Lowe *et al.* (2011), o controle é feito através da implementação de classes Java utilizadas para enviar comandos de controle no equipamento. A interface entre o cliente e o hardware depende dos detalhes da plataforma. Um exemplo seria quando o controle é feito através de uma interface intermediária (*middleware*) e não diretamente para o hardware, por exemplo, um webservice LabView que é endereçada diretamente do cliente Rig e que por sua vez, comunica com o hardware.

2.3.3.3 Controle em Lotes

No controle em lotes o usuário não interage com o experimento. Estas experiências são feitas por upload de um arquivo de controle que tem as instruções e valores para a execução do experimento. Nestes casos, o cliente é responsável por carregar o arquivo de controle, para verificar o seu conteúdo e para analisá-lo para que a ação apropriada é feita na plataforma (LOWE *et al.*, 2011).

3 METODOLOGIA

3.1 CLASSIFICAÇÃO DA PESQUISA

Este trabalho seguiu os preceitos da pesquisa aplicada, que conforme Farias Filho e Arruda Filho (2013, p. 62) tem “seus resultados voltados à aplicação prática”, de abordagem quali-quantitativa, que Figueiredo (2009, p. 97) define como “método que associa análise estatística à investigação dos significados das relações humanas, privilegiando a melhor compreensão do tema a ser estudado”.

Quanto aos seus objetivos, esta pesquisa pode ser classificada como descritiva, que de acordo com Farias Filho e Arruda Filho (2013, p. 63) “visa descrever as características de determinada população ou fenômeno”. Já no que se refere aos procedimentos técnicos, este trabalho consiste em um estudo de caso, que é descrito por Gil (2002, p. 37) como “o estudo profundo e exaustivo de um ou poucos objetos, de maneira que permita seu amplo e detalhado conhecimento”.

3.2 ETAPAS DA PESQUISA

A execução se deu em três etapas nomeadas Planejamento, Protótipo, e Validação, baseadas no método Merise (*Méthode d'Études et de Réalisation Informatique pour les Systèmes d'Entreprise*), que é uma metodologia que aborda todas as fases do desenvolvimento de sistemas de informação (LESCOURRET *et al.*, 1992).

Na etapa de Planejamento, foi realizado um estudo das ferramentas utilizadas; na etapa de Protótipo foi desenvolvido um protótipo do sistema de gerenciamento; e na etapa de Validação, se deu a elaboração de manuais e tutoriais, execução de testes e a validação do protótipo por especialistas e em turmas experimentais.

3.2.1 Planejamento

Nesta fase foram levantados os requisitos do sistema, através de reuniões entre os desenvolvedores e membros do projeto. A partir dos requisitos, foram determinadas as histórias de usuário (*user stories*), onde são definidas ações a serem desempenhadas por um ator (usuário com um papel definido) dentro do sistema.

Após o levantamento dos requisitos e definição das histórias de usuário, se deu a modelagem do sistema, onde foram criados diagramas

na linguagem UML para melhor visualização do que deveria ser implementado, além da documentação do projeto.

Além disso, foi realizado um estudo profundo das ferramentas de desenvolvimento e linguagens de programação utilizadas, a fim de definir a melhor abordagem a ser utilizada na implementação.

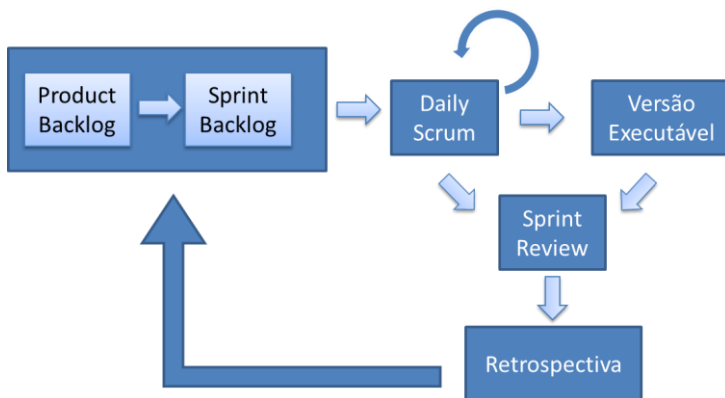
3.2.2 Protótipo

Partindo do levantamento de requisitos e modelagem do sistema proposto, o desenvolvimento se deu com a utilização da metodologia de desenvolvimento ágil de software Scrum, que é definida por Schwaber e Sutherland (2013) em sua documentação como “um *framework* para desenvolvimento e sustentação de produtos complexos”.

O Scrum tem por objetivo controlar os conflitos de interesse a fim de desenvolver sistemas e produtos de forma incremental e iterativa, considerando que os requisitos mudam frequentemente. O método Scrum é conhecido por melhorar a comunicação, maximizar a cooperação e por ser escalável (SCHWABER E SUTHERLAND, 2013).

O desenvolvimento se deu em três (3) ciclos conhecido como *sprints*, onde durante períodos determinados de tempo, entre duas e quatro semanas, uma nova funcionalidade é adicionada ao projeto e este é entregue, mesmo incompleto. Deste modo, o software pode ser avaliado e pode retornar ao desenvolvimento para que possa ser melhorado e então reavaliado (Figura 8).

Figura 8 - Ciclo de vida de um projeto na metodologia Scrum



Fonte: Adaptado de Nyström (2011, tradução do autor)

No planejamento para o sprint (*sprint planning*) foi definido o *product backlog*, que é uma lista de todos os itens necessários no produto e é a única fonte de requisitos para todas as mudanças a serem feitas ao produto. Os itens são definidos a partir dos requisitos levantados para o software a ser desenvolvido, e são conhecidos como histórias de usuário (*user stories*).

As histórias de usuário são ordenadas de acordo com sua prioridade, para que as tarefas mais importantes sejam executadas primeiro. A partir desta lista é definido o *sprint backlog*, que são os itens a serem implementados no *sprint*. Os itens de backlog neste momento são todos novas funcionalidades, ou *features*, mas a partir do primeiro *sprint* os itens de *backlog* podem ser classificados em problemas, alterações ou melhorias técnicas.

Durante um *sprint*, de acordo com Schwaber e Sutherland (2013), normalmente são realizadas reuniões diárias (*daily scrum*), a fim de sincronizar as atividades dos desenvolvedores, e criar um plano para as próximas 24 horas. Porém, estas reuniões não se deram devido ao cronograma de atividades dos bolsistas do RExLab, que não trabalham de forma síncrona, no mesmo horário e local.

As tarefas designadas foram organizadas em um quadro conhecido como *Scrum Task Board*, onde estas são agrupadas por caso de uso e divididas em tarefas a serem feitas (*to do*), em progresso (*in progress*) e finalizadas (*done*). Quando uma tarefa é iniciada, seu status passa de “a ser feita” para “em progresso”. Ao final de cada *sprint* as funcionalidades presentes naquele *release* são marcadas como finalizadas. (SCHWABER E SUTHERLAND, 2013)

Para a gestão do projeto de software foi utilizado o software JIRA, desenvolvido pela Atlassian, que permite a inserção de itens de *backlog* de produto e de *sprint*, casos de uso e tarefas. Esta solução permite a visualização das tarefas no formato de quadro, e gera gráficos de acompanhamento que podem ser utilizados para avaliação de desempenho.

No final de cada *sprint* se deu uma reunião (*sprint review*), onde uma versão executável do sistema é apresentada e avaliada. Nesta reunião o *product backlog* pode ser adaptado se necessário.

3.2.3 Validação

Nesta fase foram criados os documentos do projeto, contendo as informações levantadas na fase de Planejamento, além dos manuais técnico e de utilização. No manual técnico estão os requerimentos para

instalação do sistema, a organização dos arquivos e diretórios e a descrição das principais funções e classes. Já no manual de utilização, estão descritas todas as operações disponíveis no sistema, com capturas de tela e outras informações úteis ao usuário.

Quanto à validação, foram executados testes de performance no servidor do sistema, além da realização de um estudo com o intuito de validar a plataforma. Foram realizados testes de carga e estresse, a fim de verificar a quantidade de acessos suportada pelo servidor onde está o sistema.

Também foi aplicado um questionário de validação dos experimentos com especialistas em experimentação remota e tecnologias na educação. Além disso, nesta fase a plataforma foi utilizada por turmas experimentais no ensino médio e superior.

3 SOLUÇÃO DESENVOLVIDA

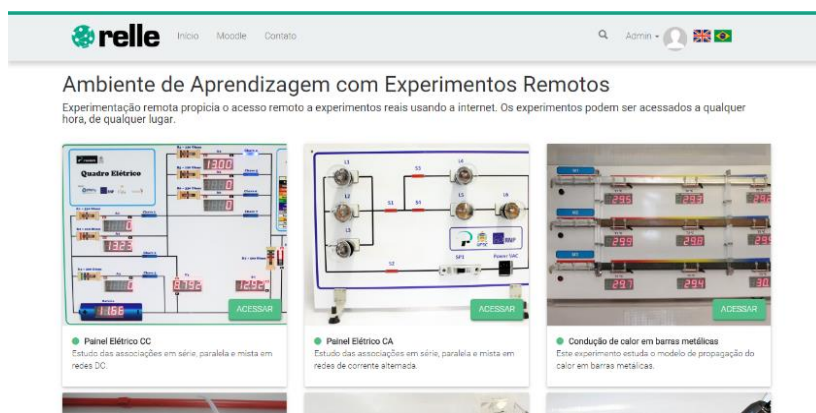
Dentre os sistemas de gerenciamento, a falta de flexibilidade em funções como o acesso sem autenticação e o agendamento opcional pode dificultar a replicação da tecnologia em diferentes laboratórios. Isso se deve ao fato da maioria das soluções terem sido implementadas levando em consideração a arquitetura de um laboratório específico.

Neste sentido, foi desenvolvida uma ferramenta baseada em módulos, onde o usuário pode acessar e gerenciar experimentos remotos.

3.2 CONCEPÇÃO E TECNOLOGIAS UTILIZADAS

A solução recebeu o nome de RELLE, acrônimo em inglês para Ambiente de Aprendizagem para Experimentos Remotos (*Remote Labs Learning Environment*), fazendo também referência aos relés utilizados no processo de automação dos experimentos.

Figura 9 - Página inicial do RELLE



Fonte: Autor

O sistema foi desenvolvido utilizando a linguagem de programação web PHP versão 5.5 através do framework Laravel 5, em conjunto com a linguagem de marcação HTML versão 5, folha de estilos CSS3 e linguagem de scripts JavaScript usando a biblioteca jQuery. O estilo utilizado foi desenvolvido tendo como base o framework para *front-end* Bootstrap. Nas próximas seções serão abordadas as tecnologias utilizadas para o desenvolvimento deste trabalho.

4.1.1. Laravel

Laravel é um *framework* (conjunto de ferramentas) para desenvolvimento de sistemas web utilizando PHP, criado por Taylor Otwell em 2011. A criação do Laravel se deu a partir da ausência de algumas funcionalidades consideradas essenciais nos frameworks mais utilizados até então. (Surguy, 2013)

Segundo He (2014), Laravel torna o processo de desenvolvimento padronizado, processando automaticamente algumas tarefas mais complicadas, permitindo que os desenvolvedores se concentrem na aplicação da lógica de negócios do sistema.

Este *framework* segue o padrão de arquitetura de software MVC (acrônimo do inglês *model-view-controller*), onde o sistema se divide em 3 (três) camadas que se comunicam entre si: modelo (*model*), visão (*view*) e controlador (*controller*).

Nos modelos ficam as relações entre os controladores e a base de dados, e tem por principal finalidade executar consultas e outras funcionalidades ligadas à manipulação de dados no banco. As classes modelo no Laravel são ligadas à uma tabela na base, e já trazem implementadas os métodos de manipulação de dados através da *Eloquent ORM (object-relational mapping)*, e permite a criação de *queries SQL* através da *Query builder*.

A camada de Visão compreende a interface do usuário e interatividade com o sistema. As partes de marcação, estilo e script ficam nos arquivos de *views*, utilizando a engine de templates *Blade* que facilita a utilização de códigos PHP na composição da interface.

Os controladores são responsáveis por validar e distribuir todas as informações, agindo como ligação entre as visões e os modelos. As validações de dados de requisições são implementadas pela classe *Controller* herdada por todos os controladores, onde o desenvolvedor só especifica os tipos de dados esperados.

O Laravel permite que se nomeie cada rota do sistema, definindo os tipos de requisições (GET, SET, POST) que esta rota pode receber. Outra funcionalidade importante são os *middlewares*, que permitem a execução de tarefas antes ou durante a execução de uma requisição. Neste sentido, a utilização de Laravel facilita a implementação de algumas funcionalidades como a tradução de páginas e verificação de login e permissões.

O *framework* permite a obtenção do local de onde o usuário está acessando durante a sessão (*locale*) ou a linguagem selecionada pelo usuário e carrega o conteúdo da página correspondente à aquela língua.

Para garantir que a segurança das funções que podem ser desempenhadas apenas por usuários com permissão de administrador, o sistema verifica à cada requisição qual o usuário está solicitando, e a permissão deste. Se o usuário não tem a permissão necessária, o middleware o redireciona para a página de login, e retorna para a página desejada depois da autenticação.

4.1.2. jQuery

A biblioteca jQuery é um conjunto de funções em JavaScript com o intuito de facilitar o desenvolvimento de scripts no lado cliente de aplicações web. É a biblioteca JavaScript mais utilizada no mundo, sendo utilizada em cerca de 65.9% dos websites. (W3TECHS, 2015)

As funções do jQuery permitem manipular tags e atributos das páginas, além de permitir a criação dinâmica de conteúdo. Além disso, é possível controlar atributos de estilo, com suporte a atributos de estilo CSS3.

Esta biblioteca trabalha em estreita colaboração com o Ajax (*Asynchronous JavaScript and XML*), cuja principal característica é o carregamento dinâmico da página, ou seja, carregar dados em uma página web sem recarregar a página inteira. As vantagens do Ajax incluem o uso de largura de banda, e separação de dados, formato, estilo e função. (GYÖRÖDI ET AL., 2009)

A implementação da fila, que informa ao usuário o tempo estimado de espera e o número de usuários aguardando, além de carregar dinamicamente o experimento remoto, faz uso de Ajax jQuery.

4.1.3. Bootstrap

O *framework front-end* Bootstrap foi criado em 2011 pelo desenvolvedor do Twitter, Mark Otto, com o objetivo de facilitar a criação de páginas responsivas, adaptáveis a diferentes tamanhos de tela.

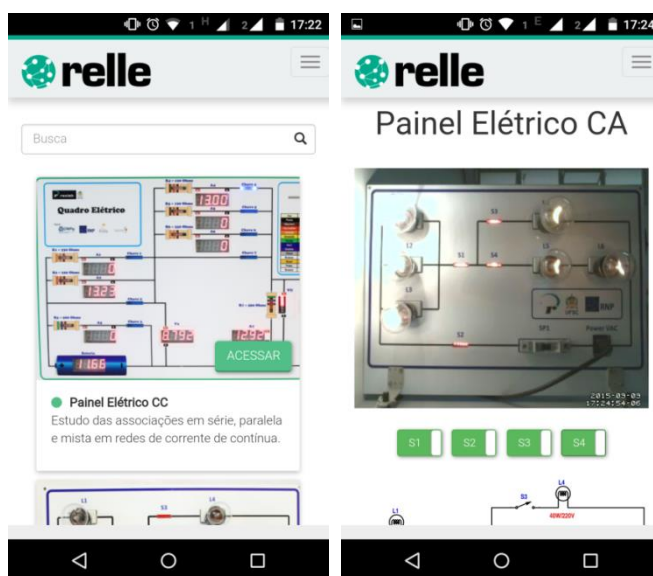
O Bootstrap conta com uma série de componentes prontos para a utilização do desenvolvedor, como ícones, formulários, tabelas, botões, listas, alertas, entre outros. Além disso, o *framework* traz algumas funções JavaScript que podem ser utilizadas na interface do sistema, como *sliders*, transições, abas e animações.

A utilização de Bootstrap torna menos complicado o tratamento de diferentes resoluções e posições de tela (no caso dos dispositivos móveis, retrato e paisagem), mantendo a mesma interface e experiência de uso.

4.1.4. Android WebView

A fim de facilitar o acesso por meio de dispositivos móveis, dispensando a utilização de um navegador para acesso ao sistema, foi desenvolvido um aplicativo Android utilizando a tecnologia WebView. O sistema operacional Android carrega a página do sistema dentro do container de um aplicativo, permitindo a utilização da mesma interface e códigos da página web (Figura 10).

Figura 10 – Sistema acessado através de um smartphone Android



Fonte: Autor

4.1.5. Node.js

Criado em 2009 por Ryan Dahl, Node.js é um *framework* orientado a eventos assíncronos, com o intuito de permitir o desenvolvimento de aplicações em rede escaláveis (TEIXEIRA, 2013, pág 16).

Ele é composto por um ambiente de execução multiplataforma e de código fonte aberto que interpreta códigos de aplicações escritas em JavaScript. Baseado no motor V8, presente também no navegador Google Chrome, se utiliza de um interpretador desenvolvido pela Google em C++ para ser usado no *backend*. O NodeJS usa um modelo orientado a eventos com operações de entrada e saída não bloqueantes. Por este motivo, ele é ideal para aplicações em tempo real com troca intensa de dados entre dispositivos distribuídos (ORNBO, 2012).

A API para acesso às funcionalidades do experimento contém funções vinculadas a *listeners*, comuns ao paradigma de orientação a eventos. Este módulo usa a biblioteca *Socket.io* e é o ponto de partida da aplicação, onde o servidor é iniciado e eventos são vinculados. O *Socket.io* é composto por dois componentes: servidor e cliente, ao qual usa principalmente o protocolo *WebSocket*, e *polling HTTP* como compatibilidade reversa.

4.1.6. WebSocket

O protocolo *WebSocket* provê um canal de comunicação bidirecional por canais full-duplex sobre um único socket TCP (Transmission Control Protocol), foi padronizado pela IETF⁴ como RFC 6455 em 2011 e é reconhecido pela W3C, logo suportado pela maioria dos navegadores atuais.

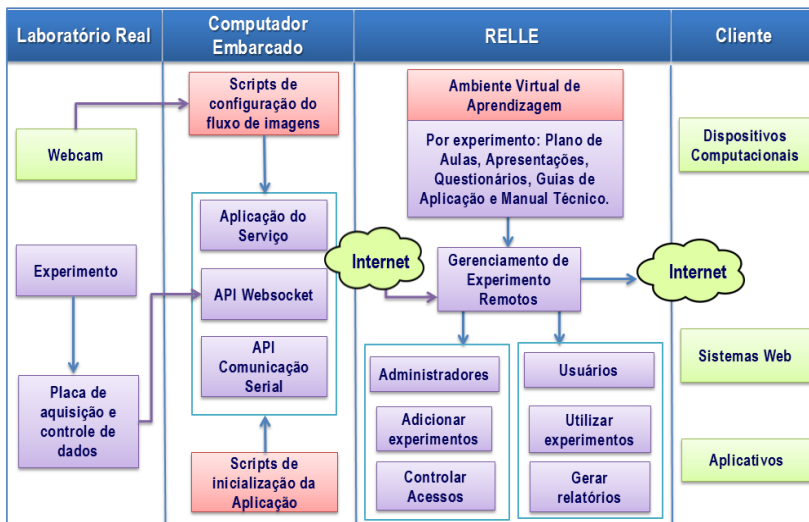
Com o *WebSocket* é possível implementar análises em “tempo real”, transmissão de dados binários, mensagens instantâneas e colaboração de documentos. O uso do protocolo *WebSocket* reduz o overhead na rede quando comparado com as técnicas de *polling* e *streaming* com o protocolo HTTP, resultando em menor latência e maior vazão de dados (PIMENTEL E NICKERSON, 2012).

3.3 ARQUITETURA

Todos os experimentos foram implementados a partir da arquitetura padronizada de hardware e software básico, conforme Figura 11. A diferenciação entre os experimentos será constatada nos tipos diferentes de sensores e atuadores, que serão instalados de acordo com as especificidades dos experimentos remotos disponibilizados.

⁴ Disponível em <https://tools.ietf.org/html/rfc6455>, acesso em 20 de novembro de 2015

Figura 11 - Arquitura do serviço



Fonte: Autor

A arquitetura foi dividida em 4 módulos: o Laboratório Real, que trata dos experimentos, Computador Embarcado, que compreende a parte de disponibilização do experimento na rede, RELLE, que é o sistema de gerenciamento, e Cliente, que trata do modelo de acesso utilizado para manipulação dos experimentos.

4.2.1. Módulo RELLE

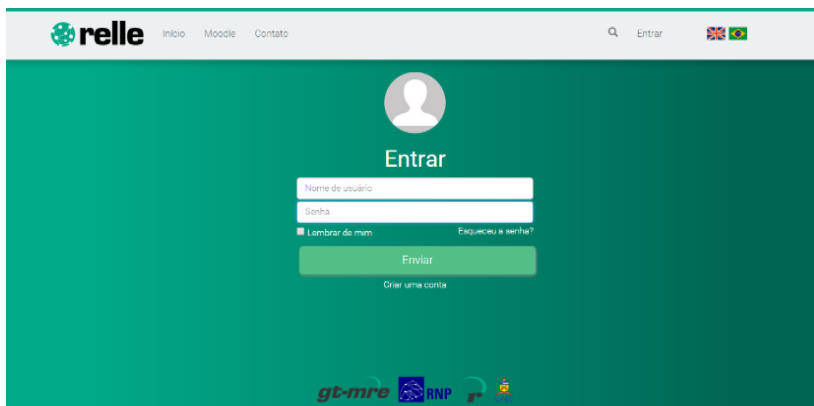
O módulo RELLE é responsável pelo gerenciamento de usuários e experimentos. Dentre as funções, estão a criação e edição de usuários e experimentos remotos, a emissão de relatórios e controle de acessos à plataforma.

Por mais que os experimentos estejam abertos a usuários visitantes, algumas das funções do RELLE são disponibilizadas apenas aos usuários registrados. Na página de autenticação do usuário (Figura 12) é realizado o acesso ao sistema utilizando um “nome de usuário” e “senha”, ou então acesso a página de cadastro, para obter uma conta.

Quando uma conta é cadastrada no sistema, é concedida a simples permissão de “usuário”, e cabe a um dos administradores alterarem o

tipo de permissão, caso necessário. O acesso do usuário no sistema garante a autenticação no RELLE e no Moodle simultaneamente, dispensando a utilização de um segundo “login”.

Figura 12 - Página de autenticação do usuário

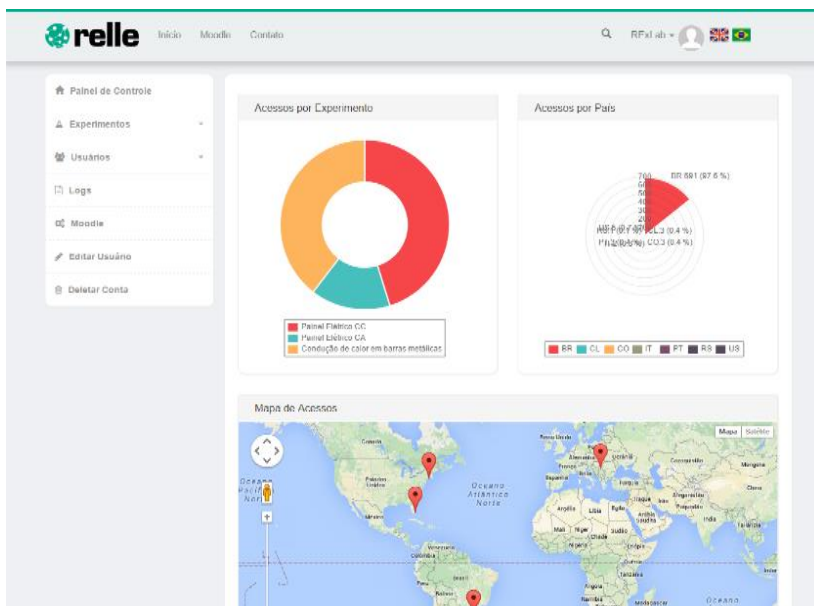


Fonte: Autor

Outra forma de acesso ao sistema é a opção de “visitante”, esse modo permite que qualquer indivíduo acesse os experimentos sem a necessidade de se cadastrar ou logar no RELLE. Esse modo tem por objetivo liberar o acesso aberto ao experimento de forma mais rápida e simplificada.

Após acessar o sistema, os usuários têm acesso ao painel de controle. Nesse painel são encontradas as principais funcionalidades do sistema exibidas em um menu disposto na lateral esquerda (Figura 13). As opções no painel de controle dependem do tipo de permissão do usuário conectado no sistema.

Figura 13 - Página do Painel de Controle



Fonte: Autor

Na página do painel de controle, o usuário visualiza informações de análise dos dados de utilização do sistema. O gráfico “Acessos por Experimento” apresenta a relação dos experimentos mais acessados; o gráfico “Acessos por País” mostra os números de acessos por país; além dos gráficos de acessos por tipo de dispositivo (mobile ou desktop) e navegador. O painel de controle conta ainda com um mapa de acessos, onde são representados os lugares de onde os experimentos foram acessados nos últimos sete dias.

O menu vertical na lateral esquerda da tela apresenta as funcionalidades de administração do sistema. Os usuários administradores são os únicos que têm acesso às funções de criar usuários (Figura 14), adicionar experimento (Figura 15), editar experimento, excluir experimento, e podem ainda editar informações de outros usuários e excluir usuários.

Figura 14 - Página 'Novo Usuário'

The screenshot shows the 'Novo Usuário' (New User) page in the RELLE system. The page has a header with the RELLE logo and navigation links (Início, Moodle, Contato). A sidebar on the left contains a 'Painel de Controle' (Control Panel) with options: Experimentos, Usuários, Logs, Moodle, Editar Usuário, and Deletar Conta. The main content area is titled 'Criar Usuário' (Create User) and contains a form with the following fields:

- Nome (Name)
- Sobrenome (Surname)
- Email
- Usuário (Username)
- Senha (Password)
- Repita a Senha (Repeat Password)
- País (Country)
- Organização (Organization)
- Tipo (Type): Por favor, selecione uma opção (Please select an option)
- Foto (Photo): Escolher arquivo (Choose file) | Nenhum arquivo selecionado (No file selected)

A green 'Enviar' (Send) button is located at the bottom right of the form. At the bottom of the page, there are logos for gt-mre, RNP, and UFSC.

Fonte: Autor

A fim de adicionar um novo experimento ao RELLE, o usuário precisa cadastrar no sistema algumas informações sobre o item como: nome, uma breve descrição (em português e inglês), conjunto de tags, o tempo de execução e uma imagem, além de classificá-lo quando ao público-alvo (ensino fundamental I, ensino fundamental II, ensino médio ou ensino superior), quanto à disciplina (física, química, biologia ou robótica), ao idioma (português, inglês ou espanhol), dificuldade e interatividade (baixa, média ou alta), conforme Figura 15.

Figura 15 - Página ‘Criar experimento’ - Aba Descrição

The screenshot shows the 'Criar experimento' page in the 'relle' system. The page is divided into a sidebar and a main content area. The sidebar contains a 'Painel de Controle' menu with options: 'Experimentos', 'Usuários', 'Logs', 'Moodle', 'Editar Usuário', and 'Deletar Conta'. The main content area is titled 'Criar experimento' and has three tabs: 'Descrição', 'Detalhes', and 'Arquivos'. The 'Descrição' tab is active and contains three sections: 'Português', 'Inglês', and 'Tags'. Each section has 'Nome' and 'Descrição' input fields. A 'Próximo' button is at the bottom right.

Fonte: Autor

A tela de cadastro de experimento está dividida em abas para melhor visualização. A descrição do experimento a ser cadastrado é feita nas duas primeiras abas, “Descrição” e “Detalhes”. Na aba “Arquivos”, o usuário carrega uma imagem para o experimento, além de um diretório comprimido contendo os arquivos para acesso ao experimento. Um modelo para estes arquivos está disponível no repositório do GitHub⁵.

⁵ https://github.com/RExLab/lab_package

Figura 16 – Página ‘Criar experimento’ - Aba Detalhes

Fonte: Autor

Depois de cadastrado no sistema, o experimento passa a ser listado na página inicial, onde o usuário pode acessar ou reservar um experimento, além de fazer buscas. A busca funciona com termos idênticos ou semelhantes aos presentes no título, descrição, tags e disciplinas de um experimento.

A partir do acesso ao experimento é gerado o registro de acesso, com informações como usuário, experimento, data e hora de início e fim, endereço IP, sistema operacional, navegador, língua, país e cidade e coordenadas que são armazenadas no banco de dados para serem usadas em “analytics”. Os registros podem ser acessados em uma página no painel de controle (Figura 17).

Figura 17- Página de logs

The screenshot shows the 'Logs' page of the RELLE system. The page has a sidebar on the left with a 'Panel de Controle' and several menu items: 'Experimentos', 'Usuários', 'Logs', 'Moodle', 'Editar Usuário', and 'Deletar Conta'. The main content area is titled 'Logs' and contains a table with the following columns: Experimento, Usuário, Início, Fim, IP, SO, Navegador, Mobile, Língua, País, and Cidade. The table lists 12 log entries, each representing a session. At the bottom of the page, there are logos for 'gt-mre', 'RNP', and 'UFSC'.

Experimento	Usuário	Início	Fim	IP	SO	Navegador	Mobile	Língua	País	Cidade
Condição de calor em barras metálicas	admin	2015-08-29 19:08:37	0000-00-00 00:00:00	150.162.232.29	Win10	Chrome	Não	pt	BR	Florianópolis
Panel Elétrico CC	admin	2015-08-29 19:02:22	0000-00-00 00:00:00	150.162.232.29	Win10	Chrome	Não	pt	BR	Florianópolis
Condição de calor em barras metálicas	admin	2015-08-29 18:30:31	0000-00-00 00:00:00	150.162.232.29	Win10	Chrome	Não	pt	BR	Florianópolis
Panel Elétrico CC	admin	2015-08-29 18:30:26	0000-00-00 00:00:00	150.162.232.29	Win10	Chrome	Não	pt	BR	Florianópolis
Panel Elétrico CA	guest	2015-08-29 17:20:10	0000-00-00 00:00:00	66.176.170.103	Win7	Firefox	Não	en	US	Fort Lauderdale
Panel Elétrico CA	guest	2015-08-29 17:19:20	0000-00-00 00:00:00	66.176.170.103	Win7	Firefox	Não	pt	US	Fort Lauderdale
Panel Elétrico CA	guest	2015-08-29 17:19:10	0000-00-00 00:00:00	66.176.170.103	Win7	Firefox	Não	en	US	Fort Lauderdale
Condição de calor em barras metálicas	guest	2015-08-29 17:15:27	0000-00-00 00:00:00	10.10.10.225	Android	Chrome	Não	pt	BR	Azarangá
Panel Elétrico CC	admin	2015-08-29 17:15:00	0000-00-00 00:00:00	10.10.10.225	Win7	Chrome	Não	pt	BR	Azarangá
Panel Elétrico CA	guest	2015-08-29 17:12:17	0000-00-00 00:00:00	10.10.10.225	Android	Chrome	Não	pt	BR	Azarangá

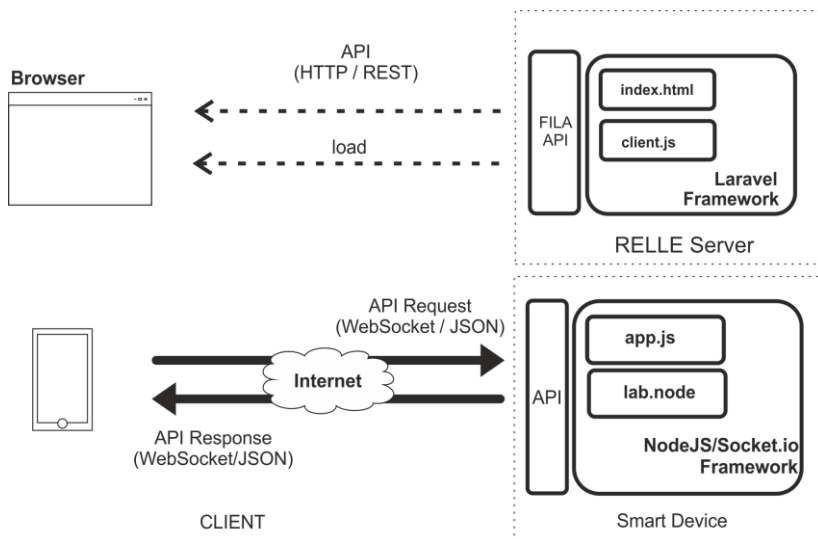
Fonte: Autor

4.2.2. Módulo Cliente

O cliente web disponibilizado pelo sistema RELLE é composto por um arquivo html, css e javascript diferentes para cada experimento. O RELLE provê uma página comum para cada experimento onde carrega os dados que foram inseridos no momento da publicação do experimento (armazenados numa base de dados). Por exemplo, o experimento de ID 1 é acessível pela URL “http://relle.ufsc.br/labs/1” pelo método GET e contém suas informações dentro do layout padrão do sistema. A partir do botão “Acessar” é possível disparar um evento para comunicação com a Web API FCFS (First-come first served).

Ao obter a permissão no navegador, o cliente navegador poderá carregar os arquivos (html, css e js), pois a API já tem o seu token de sessão como usuário sendo servido. Após carregar o cliente para o Smart Device (client.js), uma conexão WebSocket com este dispositivo é estabelecida.

Figura 18 - Esquema de comunicação cross domain no uso da API



Fonte: Autor

Quando o usuário acessa um experimento, uma página contendo a descrição do experimento é apresentada (Figura 19), e, ao clicar em acessar o usuário, é redirecionado ao experimento. O usuário não pode ter duas abas do experimento abertas no navegador, então ao clicar em acessar, porém estiver com o experimento aberto, ele recebe uma mensagem avisando que um experimento já se encontra aberto.

Figura 19 - Página do Experimento “Painel Elétrico CC”

A captura de tela mostra a interface de usuário para o experimento "Painel Elétrico CC". No topo, há o logo "relle" e links para "Inicio", "Moodle" e "Contato". À direita, há uma barra de busca e opções de idioma (BR, EN). O conteúdo principal apresenta uma imagem de um painel elétrico, o título "Painel Elétrico CC", a descrição "Estudo das associações em série, paralela e mista em redes DC.", a disciplina "Física", a duração "8 minutos" e um botão "Acessar". Na base da página, há logos de parceiros: "gt-mre", "RNP" e "USP".

Fonte: Autor

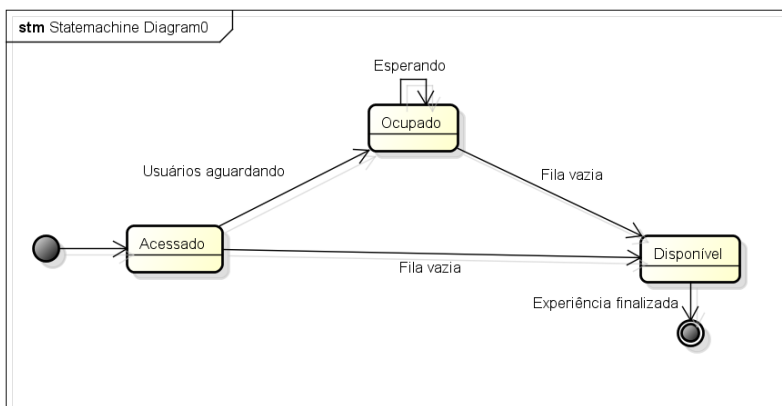
Contudo, se houver outro usuário utilizando o experimento, o usuário recebe uma mensagem avisando que ele está na fila, e qual o tempo restante para a liberação do acesso, conforme Figura 20. A fila garante o acesso ao experimento respeitando a ordem de chegada, onde o primeiro a chegar é o primeiro a ter acesso, e novos usuários vão para o final da fila, em um esquema FIFO (*First In, First Out*). O processo de controle de acesso ao experimento pode ser observado na Figura 21.

Figura 20- Fila para acesso ao experimento



Fonte: Autor

Figura 21 - Diagrama de estado de máquina da fila de acesso ao experimento



Fonte: Autor

A página do experimento é formada pela imagem da câmera e os controles de experimento, e pelas leituras provenientes dos sensores,

quando existentes, além dos blocos: “O quê?”, que descreve o que está acontecendo; e “Como?”, que explica como o usuário controla o experimento (Figura 22).

Figura 22 - Acesso ao Experimento Quadro Elétrico

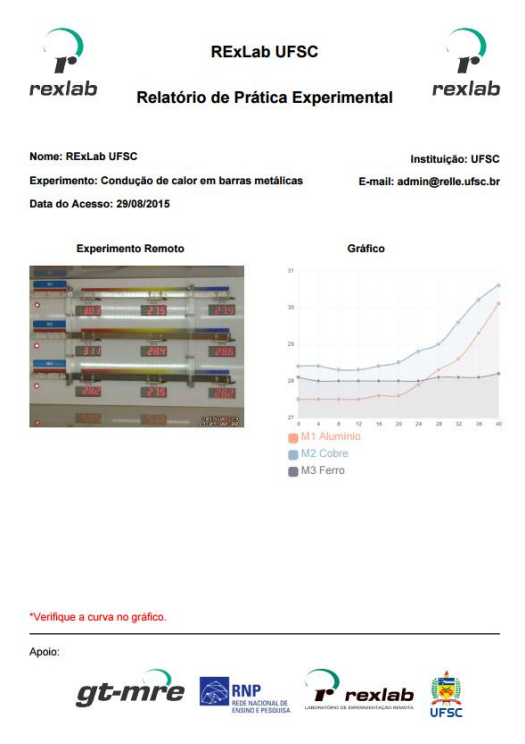
The screenshot displays the 'relle' web application interface for the 'Quadro Elétrico CC' experiment. At the top, there is a navigation bar with 'Início', 'Modulo', and 'Contato' links, a search bar, and a language selector. Below the navigation bar, a 'Sair' button is visible with a timer showing 'tempo restante: /:22'. The main content area is titled 'Painel Elétrico CC' and features a digital readout (DRO) of the circuit on the left and a schematic diagram on the right. The DRO shows four current meters (A1-A4) all at 0.000 and four voltage meters (V1-V4) with values 2.124, 1.155, 2.092, and 1.153. Below the DRO are eight red buttons labeled A1 through A8. The schematic diagram shows a circuit with a 5V source, resistors, and meters. At the bottom, there is a 'Gerar Relatório' button and two informational icons: a lightbulb and a question mark. The footer includes logos for 'gt-mre', 'RNP', and 'UFSC'.

Fonte: Autor

Na parte inferior da página, o usuário tem a informação de quanto tempo ainda resta, e ao botão “Sair” para finalizar sua sessão. Ao clicar nesse botão, ou quando o tempo limite é esgotado, o acesso é direcionado para uma página com opções de exportação dos dados. Os dados do experimento podem ser então exportados nos formatos de documento portátil (*.PDF) ou valores separados por vírgula (*.CSV).

O relatório em PDF (Figura 23) contém os dados do usuário, os resultados das leituras, gráficos, e uma imagem capturada pela câmera. Já o formato CSV permite a manipulação dos dados a partir dos principais softwares de leitura de dados como documentos de texto, planilhas eletrônicas (Microsoft Office Excel®, Calc, entre outros) e softwares voltados a cálculo numérico (MATLAB®, Scilab, GNU Octave, FreeMat, R, entre outros).

Figura 23- Relatório de prática experimental exportado em PDF



Fonte: Autor

Figura 24 - Ambiente Virtual de Aprendizagem no RELLE

Fonte: Autor

4.2.3. Módulo Computador Embarcado

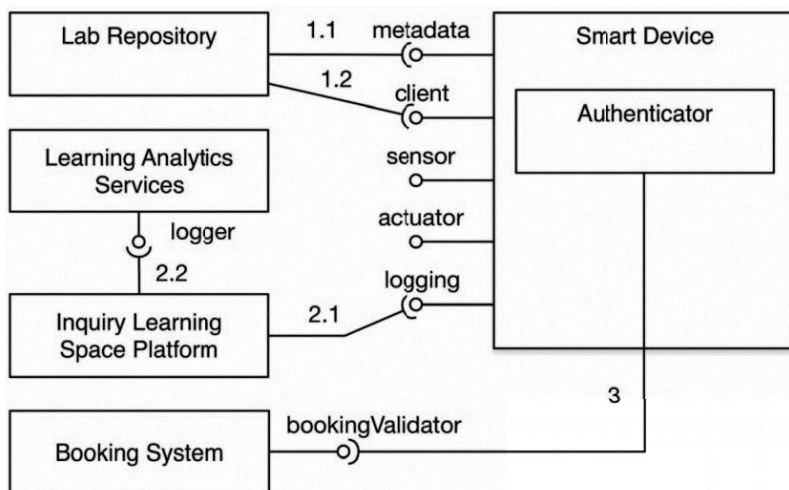
As funcionalidades para disponibilizar os experimentos na internet estão baseadas em placas “*Single board Computers*” (SBC). Dentro das perspectivas deste trabalho, foram utilizadas soluções de baixo custo, adotando o computador embarcado Raspberry Pi B+ para desenvolvimento dos protótipos. Essa escolha deve-se à presença de recursos similares aos computadores desktops, com suporte ao sistema operacional Linux e a facilidade de aquisição do dispositivo no mercado.

A aplicação de cada experimento não requer alto uso da memória e pode ser utilizada em qualquer sistema Linux. O desenvolvimento foi baseado no sistema Linux e, então, compilado para dispositivos Raspberry Pi (utilizando compilador cruzado).

O resultado é uma arquitetura fracamente acoplada, que habilita o compartilhamento dos experimentos em outras plataformas. Esse paradigma, chamado por Salzmann *et al.* (2015) de *Smart Devices*, utilizado no projeto Go-Lab⁶ (Global Online Science Labs for Inquiry Learning at School), no qual estão bem destacadas aplicações clientes e servidor, e fornecem interfaces bem definidas entre o usuário e o sistema, conforme Figura 25.

⁶ <http://www.go-lab-project.eu/>

Figura 25 - Diagrama de componentes das interações entre os serviços do Go-lab e o Smart Device



Fonte: Salzmann *et al.* (2015)

Considerando que laboratórios remotos geralmente necessitam de um protocolo assíncrono para entregar seus resultados ao cliente, técnicas de polling e long-polling com HTTP não são tão eficientes quando comparadas com protocolos assíncronos, como o WebSocket.

Os *Web services* permitem acessar entradas e saídas de dispositivos conectados à internet. Funcionalidades referem-se a comportamentos internos como validação da faixa de entrada de um atuador ou reconhecimento de situações críticas, como o superaquecimento. Para isso, foi elaborada uma especificação detalhada utilizando uma linguagem de descrição baseada em JSON (*Javascript Object Notation*). O compartilhamento dessa especificação permite um completo desacoplamento de cliente-servidor, e pretende garantir interoperabilidade e integração dos recursos do laboratório em qualquer software cliente.

O formato JSON é utilizado para descrever e trocar conjuntos de dados entre aplicações, independente da linguagem em que cada uma foi desenvolvida. Esse padrão foi utilizado para representar entradas e saídas de dados. Trazemos o exemplo do laboratório de circuitos resistivos DC para ilustrar a descrição deste *Smart Device*.

A descrição do serviço por metadados permite a descoberta de serviços pelo cliente, incluindo tipos de parâmetros, valores aceitos, unidades de medidas, etc. Tão importante quanto descrever os recursos internos ao experimento é validar as entradas fornecidas pelo usuário e observar e agir sobre estados internos que representam situações de risco, tal como: verificar a faixa de intervalos de tempo para manter a fonte de calor ligada e temperatura máxima lida em qualquer termômetro, respectivamente.

O computador embarcado também é responsável pelo serviço de streaming. No GT-MRE foi optado pelo uso de câmeras web com conexão USB devido o baixo custo e a facilidade de aquisição. O mesmo computador embarcado utilizado para controle do experimento também é o responsável pelo gerenciamento e disponibilização do streaming no formato MJPEG (Motion JPEG). O MJPEG é um formato de compressão de vídeo na qual cada frame de vídeo é comprimido separadamente como uma imagem JPEG. No HTTP streaming, cada imagem é separada em respostas individuais por um marcador específico. O tipo de conteúdo (mime-type) *multipart/x-mixed-replace* informa ao cliente para esperar vários quadros delimitados por um identificador (*boundary-name*).

Visto que existem muitos servidores de streaming de código aberto, optou-se pelo *Motion* para explorar aspectos de leveza (utilização de poucos recursos) e configuração flexível. O Motion é um software escrito em C para sistemas Linux que usa a API *video4linux*, e é capaz de detectar se uma parte significativa da imagem tem mudado. Algumas variáveis são ajustadas através de seu arquivo de configuração principal para adequar-se aos requisitos de nossa aplicação.

Atualmente, os principais navegadores do mercado como Firefox, Google Chrome e Safari já possuem o suporte nativo para o streaming MJPEG. Outros navegadores, como o Internet Explorer podem apresentar o fluxo de imagens com o auxílio de plug-ins externos, como o Cambozola. Para clientes Android existem bibliotecas de código fonte aberto para incluir um visualizador MJPEG em aplicações de código nativo.

4.2.4. Módulo Laboratório Real

Todos os experimentos utilizam a mesma arquitetura básica. Em cada experimento, placas de aquisição e controle são responsáveis por receber sinais dos sensores e transmitir ao computador embarcado que gerencia o experimento pela aplicação desenvolvida.

Além disso, para facilitar a replicação e utilização da ferramenta, cada experimento conta com manual técnico, plano de aula, guia de aplicação e caderno didático, além de um curso modelo no Moodle para diferentes níveis de ensino.

Os experimentos disponíveis na plataforma são relacionados nas seções a seguir.

4.2.4.1. Painel Elétrico CC

Este experimento aborda as associações em série, paralela e mista em redes de corrente de contínua. O usuário controla sete chaves no painel configurando o circuito de diversas formas, e sete sensores de corrente e três de tensão mostram os dados em diferentes pontos.

Estes registros são apresentados na tela durante o experimento, conforme Figura 26, e no relatório ao final do experimento.

Figura 26- Experimento Painel Elétrico CC

rele Início Moodle Tutoriais GT-MRE Contato

Tempo restante: 7:55

Painel Elétrico CC

0.0 mA 12.0 mA 21.3 mA 0.0 mA 0.0 mA 0.0 mA 0.0 mA 11.78 V 0.00 V

O caminho que a eletricidade percorre altera as propriedades do circuito. Em caminhos contínuos o circuito está em série e em caminhos alternativos está em paralelo.

Ação: as chaves na imagem para configurar o circuito. Acesse o Moodle para o Ensino Médio e para o Ensino Superior. Baixe o Material Didático e o Manual Técnico.

RNP CAPEX gt-mre UFSC

Fonte: Autor

4.2.4.2. Painel Elétrico CA

Este experimento aborda as associações em série, paralela e mista, semelhantemente ao “Painel elétrico CC”, porém em redes de corrente de alternada. Neste experimento é possível observar a intensidade luminosa de seis lâmpadas variar de acordo com a configuração do circuito, para isso quatro chaves são dispostas em diferentes pontos e controladas pelo usuário, conforme Figura 27.

Figura 27 - Experimento Painel Elétrico CA

The screenshot displays the 'Painel Elétrico CA' experiment interface. At the top, there is a navigation bar with the 'relle' logo and links for 'Inicio', 'Moodle', 'Tutoriais', 'GT-MRE', and 'Contato'. A search bar and language selection (English and Portuguese) are also present. Below the navigation bar, a search input field shows 'Ser' and a timer indicating 'Tempo restante: 9:44'. The main title 'Painel Elétrico CA' is centered above the experiment content. The content is divided into two main sections: a photograph of the physical circuit board on the left and a schematic diagram on the right. The photograph shows six incandescent lamps labeled L1 through L6 and four switches labeled S1 through S4. The schematic diagram illustrates the electrical connections, including a power source (VAC 220V), a fuse (F1), and the arrangement of lamps and switches. Below the images, there are four interactive buttons for switches S1, S2, S3, and S4. At the bottom of the interface, there are two informational text boxes. The first box, with a lightbulb icon, states: 'As lâmpadas incandescentes substituem os resistores no circuito misto, e a intensidade luminosa emitida por cada lâmpada dá uma percepção visual da potência dissipada.' The second box, with a question mark icon, states: 'Ação: as chaves na imagem para configurar um novo circuito. Alterando a ligação, altera-se também as correntes e tensões nos terminais de cada lâmpada. Acesse o Moodle para o Ensino Médio e Baixe o Material Didático.' At the very bottom, there are logos for RNP, CAPES, gt-mre, and UFSC.

Fonte: Autor

4.2.4.3. Condução de calor em barras metálicas

Neste experimento, o estudo da propagação do calor por condução é feito através de três barras metálicas feitas de aço, cobre e alumínio, que são conectadas a uma fonte de calor, que às aquece em uma extremidade. Três sensores de temperatura em cada barra, indicam a temperatura em distâncias semelhantes, de modo que o aluno pode perceber como o calor se propaga.

Os registros da temperatura são indicados em um gráfico dinâmico na tela, conforme a Figura 28. Ao fim do experimento, o usuário tem acesso aos dados e ao relatório da experiência.

Figura 28 - Experimento Condução de Calor em Barras Metálicas

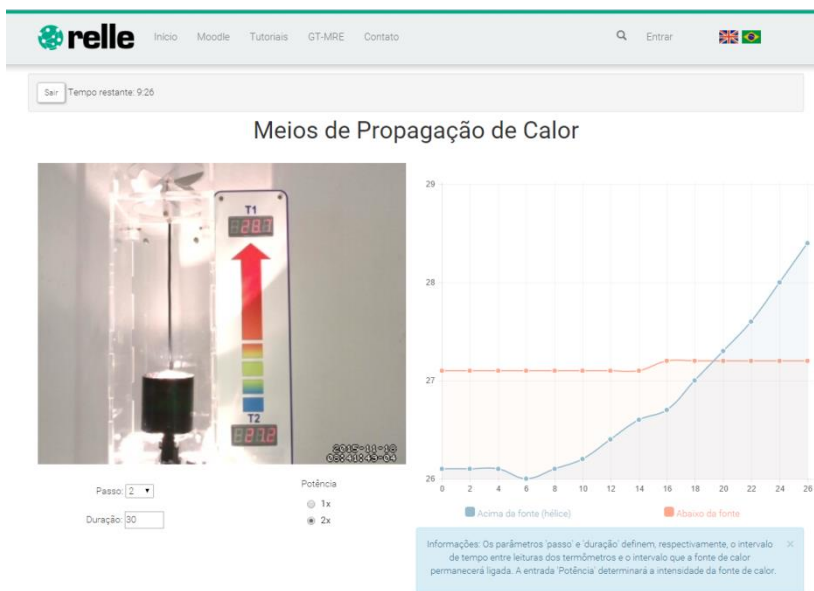


Fonte: Autor

4.2.4.4. Meios de Propagação de Calor

Estudo dos meios de propagação de calor por convecção e irradiação. Para isso o usuário escolhe a potência de uma lâmpada que aquece a parte inferior de uma tubulação de acrílico. A massa de ar quente passa por ser menos densa, e sobe pela tubulação, e a massa de ar frio das camadas superiores desce por ser mais densa. Dois sensores registram a temperatura, que é mostrada na tela (Figura 29), e também no relatório final.

Figura 29 - Experimento Meios de Propagação de Calor



Fonte: Autor

4.2.4.5. Microscópio Remoto

O experimento consiste em um microscópio, utilizado para estudo da morfologia das angiospermas, onde uma base giratória contendo as partes de uma planta (fruto, semente, flor, folha, caule e raiz) é controlada pelo usuário. A imagem do microscópio é apresentada na tela, junto às informações da parte que está sendo mostrada, como mostra a Figura 30.

Figura 30 - Experimento Microscópio Remoto

rele Início Moodle Tutoriais GT-MRE Contato

Tempo restante: 4:55

Microscópio Remoto

Amostra: Flor

Descrição: A flor é o órgão de reprodução das plantas angiospermas, plantas cuja semente é protegida por uma estrutura que chamamos de fruto.

tubo polínico
estame
antera
filete
pétala
sépala
receptáculo floral
pedicelo
ovário
ovulina
tubo gineceu

A base do microscópio contém amostras de partes de plantas que podem ser movidas para a direita e esquerda. Ao lado da imagem do microscópio são apresentados mais detalhes sobre a amostra.

Clique nas setas para para movimentar o microscópio. Acesse o material didático do experimento.

RNP CAPES gt-mre UFSC

Fonte: Autor

4.2.4.6. Plano Inclinado

O experimento consiste em uma base que pode ter seu ângulo de inclinação controlado pelo usuário, e uma esfera polimérica. A esfera percorre o plano com determinada velocidade que varia de acordo com a aceleração dada a partir da inclinação da base.

O tempo que a esfera levou para percorrer a base a cada 10 cm é retornado ao usuário. Além disso, são mostrados na tela o componente da força no eixo y e no eixo x , como mostra a Figura 31.

O plano inclinado pode ser utilizado no estudo da Segunda Lei de Newton, movimento retilíneo uniformemente variado (MRUV), decomposição de vetores, conceitos de aceleração, velocidade, além de conceitos de conservação da energia.

Figura 31 - Experimento Plano Inclinado


[Inicio](#)
[Moodle](#)
[Tutoriais](#)
[GT-MRE](#)
[Contato](#)







Sair Tempo restante: 9:47

Plano Inclinado





0°
Ângulo

$mg \sin(\theta)$ [N]:

$mg \cos(\theta)$ [N]:

d (cm)	10	20	30	40	50
t (ms)					

Se você estiver usando um smartphone ou tablet, poderá usar o acelerômetro do seu dispositivo para ajustar o ângulo. ✕

 O corpo percorre o plano com determinada velocidade que varia de acordo com a aceleração dada a partir da inclinação da base. Acesse o material didático para o Ensino Médio.

 Primeiramente, ajuste para "0°" e envie este ângulo para prender a esfera. Selecione um novo ângulo para realizar a experiência e o envie. Espere até chegar na posição e, então, clique em "Soltar" para que o corpo seja liberado.

Fonte: Autor

4.2.4.7 Disco de Newton

Este experimento é utilizado para demonstração dos conceitos de decomposição e sobreposição das cores. Ao girar, o disco mistura as cores resultando em um tom próximo de branco. Além disso, o usuário pode controlar uma simulação (Figura 32) que mostra como o fenômeno se dá em outros conjuntos de cores, como RGB e azul e amarelo, além de poder criar um disco personalizado com as cores que desejar.

Figura 32 - Experimento Disco de Newton

relle Inicio Moodle Tutoriais GT-MRE Contato Q Entrar

Sar Tempo restante: 4:52

Disco de Newton

Red-Green-Blue VIBGYOR Amarelo-Azul Novo disco

2015-11-10 05:49:14-13

OFF

- 0 + Nova velocidade

O disco de Newton contém setores pintados com as cores do arco-íris. Quando iluminado e girado velocemente, o disco adquire a cor branca se a proporção de cores for correta. Geralmente, a combinação de cores dos setores resultará num tom cinza bastante claro. Da mesma forma, é possível deduzir que a luz branca é proveniente da soma dos espectros luminosos.

Clique na chave para ligar o experimento. Ao lado da imagem que mostra o processo real, é possível interagir com uma simulação que rotaciona o disco a diferentes velocidades. Estão disponíveis as misturas das cores do arco-íris (VIBGYOR), o conjunto RGB (vermelho, verde e azul), o conjunto Amarelo e Azul, e também é possível criar um novo disco. Acesse o material didático para o Ensino Fundamental.

Fonte: Autor

4.2.4.7 Ambiente para Desenvolvimento em Arduino

Este experimento trabalha programação para o microcontrolador Arduino, e permite a verificação, carregamento de códigos e manipulação de um Arduino remotamente, além do controle de sensores e atuadores. É composto por um Arduino UNO, visor LCD 16x2, sensor de temperatura de umidade DHT11, LED, servo motor e uma *protoboard* de 840 furos.

A IDE web permite que o usuário faça upload e download de arquivos, e a conta com exemplos e código para utilização dos componentes. Após a compilação do código do usuário, o retorno do compilador é mostrado na tela. Além disso, após o upload do código, o usuário pode enviar dados via porta serial para o Arduino remoto.

O experimento é uma releitura do primeiro experimento do RexLab, um depurador remoto para o microcontrolador 8051.

Figura 33 - Experimento Ambiente de Desenvolvimento para Arduino

The screenshot displays the Arduino IDE web interface. At the top, there is a navigation bar with the 'relle' logo and links for 'Inicio', 'Moodle', 'Tutoriais', 'GT-MRE', and 'Contato'. A search bar and language flags (Brazilian and English) are also present. The main title is 'Ambiente de Desenvolvimento para Arduino'.

The central area is divided into three main sections:

- Code Editor:** Shows C++ code for an Arduino sketch. The code includes comments in Portuguese and functions for setting cursor positions, printing sensor data (temperature and humidity), and clearing the serial monitor. It uses the `Serial` library and `delay(4000)` for timing.
- Serial Monitor:** A window titled '/dev/ttyACM0' with a 'Send' button. It displays the output of the sketch: 'Umidade %: 40.00', 'Temperatura em Celsius: 24.00', and 'Umidade %: 40.00'. Below the monitor is a baud rate dropdown set to '9600 baud' and a 'Clear' button.
- Hardware:** Two images of the physical setup. The top image shows the Arduino board connected to a breadboard with various components. The bottom image is a schematic diagram of the circuit, showing the Arduino connected to a sensor module and a display screen. A note below the diagram reads '*Base o module para ampliar'.

At the bottom of the interface, there are logos for RNP, CAPES, GT-MRE, and UFSC.

Fonte: Autor

4.2.4.7. Efeito Fotoelétrico

Estudo das transformações energéticas, verificação da conversão da energia solar em elétrica e mecânica. A energia luminosa vinda da fonte (lâmpada) incide sobre a placa fotovoltaica feita de silício que converte a energia luminosa em energia elétrica. Os elétrons do material semiconductor entram em movimento quando sobre o mesmo incide a radiação eletromagnética vinda da fonte, (UV-VIS), e o movimento desses elétrons resulta em uma corrente elétrica contínua.

Figura 34 - Experimento Efeito Fotoelétrico


[Início](#)
[Moodle](#)
[Tutoriais](#)
[GT-MRE](#)
[Contato](#)




Conversão de energia luminosa em energia elétrica



2015-11-15 08:45:12-07

Rotações por minuto:
 Posição motor:

Fonte de luz: ON

A hélice é acoplada a um motor de corrente contínua alimentado pela célula solar. A frequência de giro da hélice é dada em rotações por minuto (RPM).


 A célula solar converte energia luminosa em energia elétrica. A quantidade de energia convertida está diretamente relacionada com a intensidade de luz que incidiu na célula. Confira o material didático para o Ensino Médio.


 Altere a chave de OFF para ON para ligar a fonte de luz. Usando o controle deslizante é possível cobrir a fonte de luz gradualmente. O valor varia de 0 (mais à esquerda ou totalmente aberto) a 800 (mais à direita ou fechado).






Fonte: Autor

5 TESTES E VALIDAÇÃO

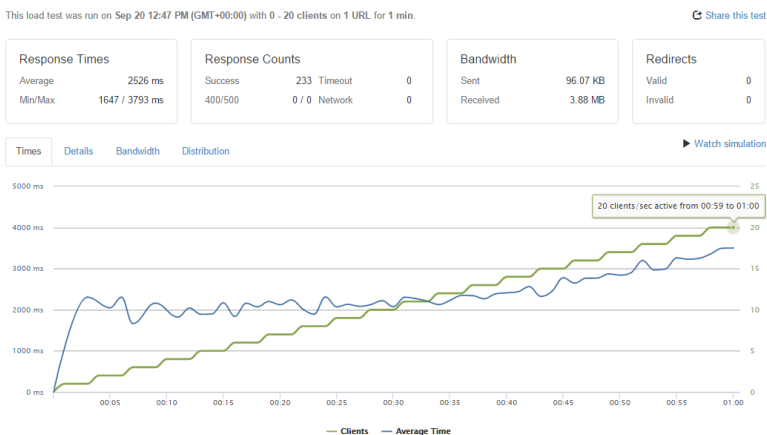
5.2 TESTES DE PERFORMANCE

O teste de performance considera os aspectos funcionais como disponibilidade, escalabilidade, tempo de resposta no servidor, no computador embarcado e no cliente. Para essa análise foi utilizado o serviço baseado em nuvens do “loader.io” que realiza testes de carga e escalabilidade em aplicações web e APIs permitindo configurar a quantidade de conexões simultâneas de três formas diferentes: manter clientes, clientes por teste e clientes por segundo.

Em cada um desses testes foram considerados os casos: a) o máximo de clientes sem erro; e b) o máximo de clientes com timeout menor que 40% em um minuto de testes. Cada teste foi realizado por cerca de 5 vezes e os valores médios são apresentados a seguir.

O primeiro teste analisa a carga de clientes que o servidor é capaz de suportar, “*Maintain client load*”. Durante todo o teste é mantida uma contagem de clientes. No primeiro caso, denominado “1.a”, foram inseridos 20 clientes a serem mantidos durante um minuto. Esses clientes realizaram 233 requisições, alcançando a taxa de 9 requisições por segundo. Durante o processo o servidor não sofreu falhas e nem perdas, com uma média de tempo de resposta em 2,5 segundos, conforme Figura 35.

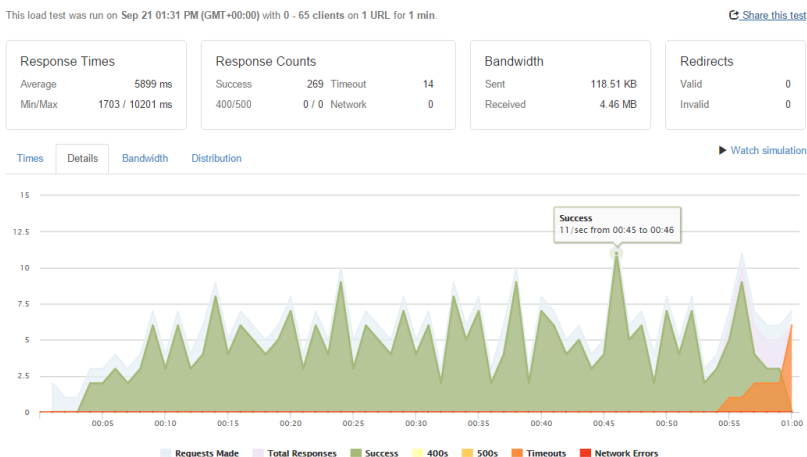
Figura 35 - Dados do teste “1.a”. Destaque para o tempo de resposta (linha azul) e a quantidade de clientes mantidos (linha verde)



Fonte: Autor

No segundo caso, denominado “1.b”, foi inserida uma carga de 65 clientes durante um minuto. Com essa configuração o servidor atingiu 11 requisições por segundo, totalizando com sucesso 269 requisições no período de um minuto. Já ao atingir 60 clientes por segundo, o tempo limite de resposta foi atingido em 14 requisições, que representaram cerca de 5% das requisições durante o teste. As falhas prejudicaram também o rendimento do tempo de resposta que teve média de 5,9 segundos, conforme Figura 36.

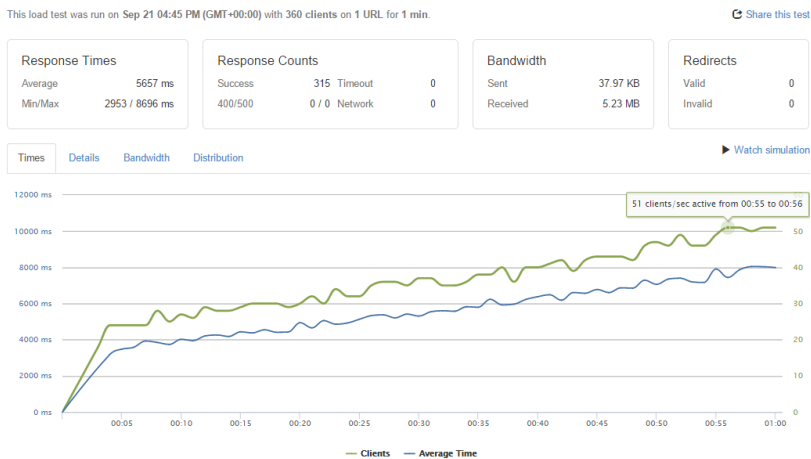
Figura 36 - Dados do teste “1.b”. Demonstrando as requisições por segundo respondidas com sucesso (em verde) e timeouts (em laranja)



Fonte: Autor

Seguindo os testes, a segunda análise executa uma verificação de clientes por teste (*Clients per test*). Nessa análise, uma quantidade de clientes é distribuída uniformemente durante a execução, o que possibilita analisar a escalabilidade do servidor. O serviço conseguiu responder com sucesso 360 clientes distribuídos em um minuto de testes, alcançando até 51 clientes e 10 requisições por segundo, mantendo o tempo médio de resposta em 5,6 segundos e sem timeout, conforme Figura 37.

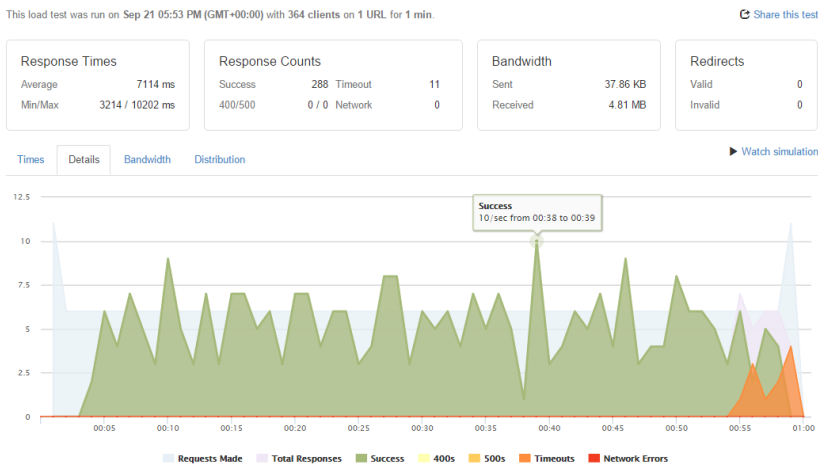
Figura 37 - Dados do teste “2.a”. Destaque para o tempo de resposta (linha azul) e a quantidade de clientes mantidos (linha verde).



Fonte: Autor

Em um segundo caso, o teste denominado “2.b”, foram inseridos 364 clientes que realizaram 288 requisições ao qual todas foram respondidas com sucesso, em uma média de 7,1 segundos. Porém, ao alcançar 68 usuários por segundo, o servidor sofreu com requisições de tempo esgotado num total de 11 ao final do teste (Figura 38), reforçando os dados verificados durante o teste “1.b” de limite de usuários.

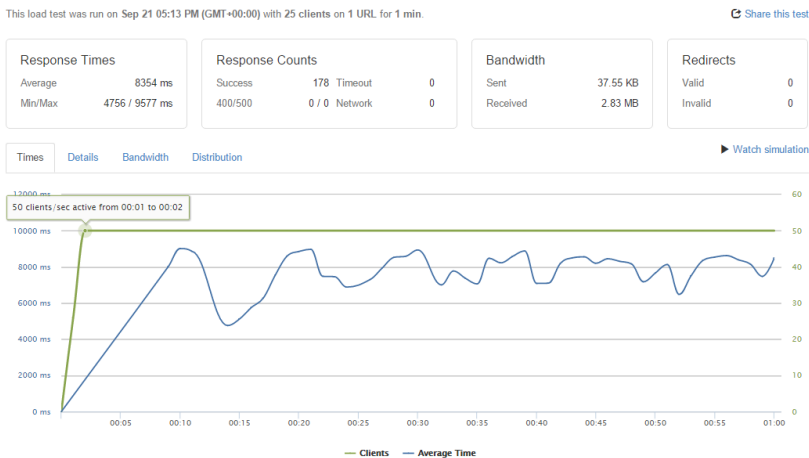
Figura 38 - Dados do teste “2.b”. Demonstrando as requisições por segundo respondidas com sucesso (em verde) e timeouts (em laranja)



Fonte: Autor

Por fim, o teste clientes por segundo (*Clients per second*) é executado com uma carga de clientes que é iniciada logo no primeiro segundo. Nesse, a melhor resposta foi o acréscimo de 25 clientes por segundo, chegando a 50 clientes aos dois segundos, conforme Figura 39. Esses clientes foram mantidos durante um minuto com o tempo de resposta médio de 8,3 segundos o que resultou em uma taxa de sucesso de 100% das 178 requisições, com picos de 14, 15, 16 e até 19 requisições por segundo.

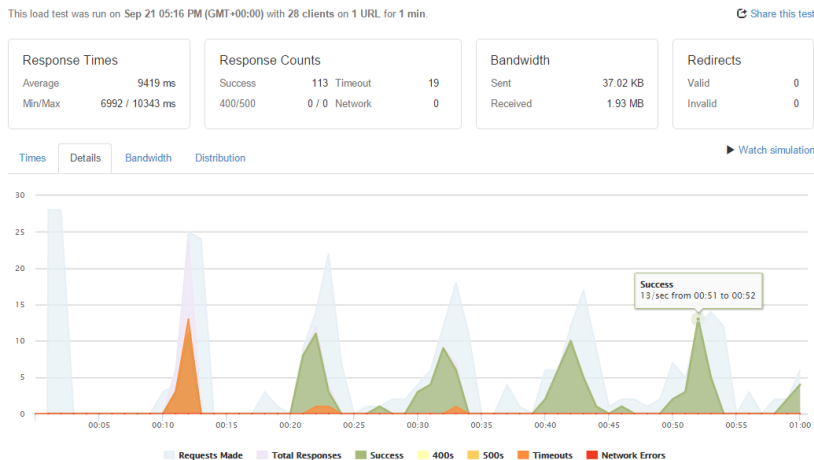
Figura 39 - Dados do teste “3.a”. Destaque para o tempo de resposta (linha azul) e a quantidade de clientes mantidos (linha verde)



Fonte: Autor

No último caso, denominado “3.b”, foram acrescentados 28 clientes por segundo, chegando a 56 clientes logo aos dois segundos de teste. Mantendo esses usuários, o servidor chegou a responder 13 requisições por segundo, o que resultou em 113 requisições com sucesso e uma falha por tempo esgotado de 19 requisições.

Figura 40 - Dados do teste “3.b”. Demonstrando as requisições por segundo respondidas com sucesso (em verde) e timeouts (em laranja)



Fonte: Autor

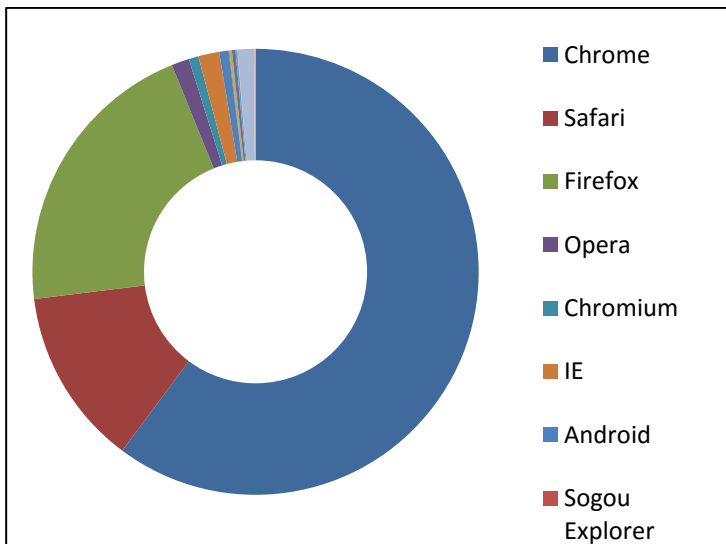
5.3 ACESSOS

A partir das informações providas pelos logs do sistema, é possível determinar um perfil dos usuários do sistema. Os dados foram coletados à partir da função de exportação de logs do RELLE, e são de um período entre os dias 11 de setembro e 30 de setembro de 2015.

Quanto ao número de acessos por tipo de dispositivo, pôde-se observar que a maioria dos acessos foi feito a partir de dispositivos convencionais (93,54%), e não de dispositivos móveis (6,45%).

Os navegadores mais utilizados pelos usuários (Gráfico 2) são o Google Chrome (60,2%), Mozilla Firefox (20,8%) e Safari (12,9%), e estes números estão diretamente ligados à estatística geral de uso de navegadores web. Segundo a W3C (2015), o navegador mais utilizado no mundo é o Google Chrome (64%), o segundo é Mozilla Firefox (21,12%) e o Safari (4,5%).

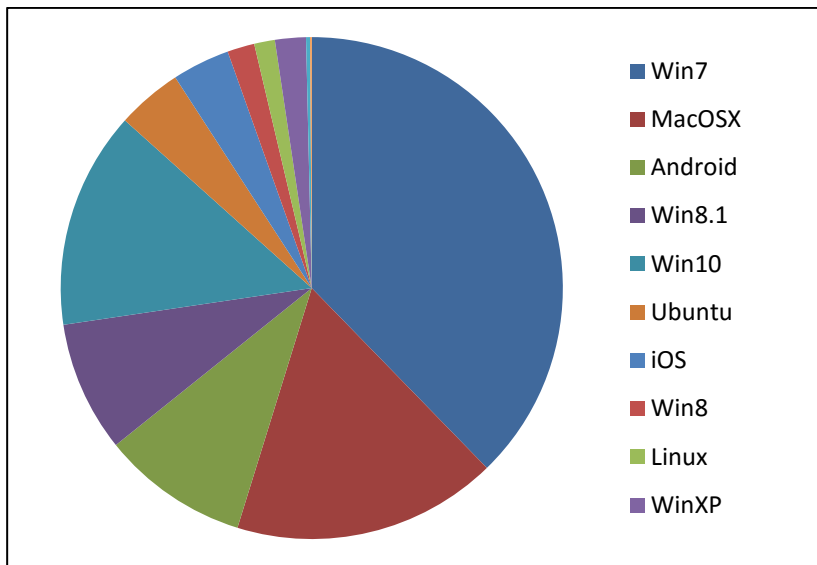
Gráfico 2 - Navegadores utilizados no acesso ao sistema



Fonte: Autor

Os sistemas operacionais mais utilizados pelos usuários (Gráfico 3) são as versões do Microsoft Windows (60,2%) e MacOS X (17%). Distribuições Linux são utilizadas por apenas 5,54% dos usuários.

Gráfico 3 - Sistemas operacionais utilizados no acesso ao sistema



Fonte: Autor

Quanto à localização geográfica dos usuários, a Figura 41 mostra que o sistema recebeu acessos de todos os continentes. Notou-se certa concentração de acessos não só no Brasil, mas também na Europa e costa-leste dos Estados Unidos.

Figura 41 - Mapa de acessos ao RELLE



Fonte: Autor

Estes resultados refletem o período de validação dos experimentos (Seção 5.3), onde especialistas do mundo inteiro foram convidados a acessar o sistema e avaliar os experimentos. A Tabela 1 mostra os 10 países que mais acessaram o sistema, e o número de acessos de cada um.

Tabela 1 - Número de acessos por país

País	Acessos
Brasil	3365
Ucrânia	119
Estados Unidos	51
Espanha	33
Reino Unido	28
Portugal	28
Austrália	20
Argentina	15
Chile	15
Uruguai	14

Fonte: Autor

5.4 VALIDAÇÃO POR ESPECIALISTAS

Foram convidados para avaliar o sistema cerca de 300 especialistas dos 5 continentes, das áreas de experimentação remota, educação e engenharias, e tecnologias da informação e comunicação. Dentre os convidados estão coordenadores de laboratórios de experimentação remota, organizadores de eventos na área e membros do grupo de trabalho da IEEE que trata de padronização de experimentação remota (IEEE-SA P1876⁷ WG).

Foi enviado por e-mail um questionário (Apêndice I) com perguntas sobre o perfil do profissional, sobre sua experiência na utilização de experimentos reais e remotos, e sobre os experimentos disponíveis na plataforma. O questionário era formado por perguntas

⁷ <https://ieee-sa.imeetcentral.com/1876public/>

abertas e de múltipla escolha, além de um campo para que o avaliador atribuisse uma nota para os experimentos analisados.

O questionário foi respondido por 56 especialistas de 26 países, conforme mostra a Tabela 2.

Tabela 2 - Número de Avaliadores por País

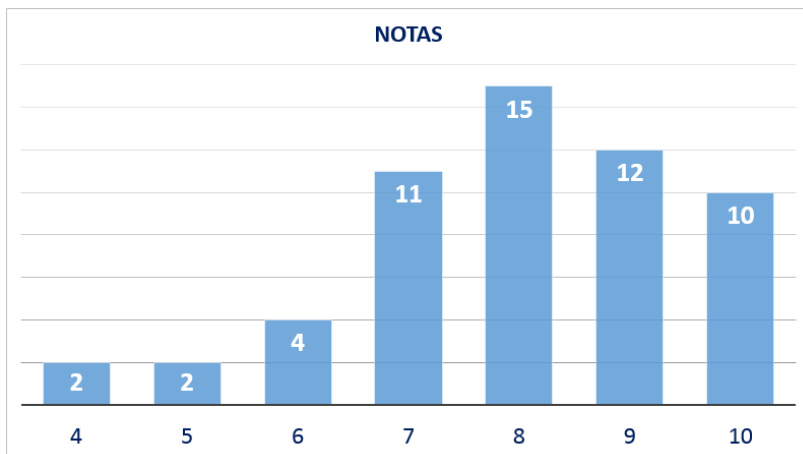
Número de Avaliadores por País			
<i>País</i>	<i>Especialistas</i>	<i>País</i>	<i>Especialistas</i>
Argentina	1	Itália	1
Austrália	4	Japão	1
Áustria	1	Ilhas Maurício	1
Brasil	9	Paraguai	2
Chile	1	Portugal	8
Chipre	1	Rússia	1
Colômbia	2	Sérvia	2
Costa Rica	2	Sri Lanka	1
Egito	1	Tunísia	1
Espanha	6	Turquia	1
Finlândia	2	Uruguai	1
Grécia	1	USA	3
Índia	1	Zimbábue	1

Fonte: Autor

Dentre os avaliadores, a maioria (52,9%) utiliza práticas em laboratório real, e 64,7% já utilizou experimentação remota antes de avaliar os experimentos. Quando questionados sobre se recomendariam o uso do ambiente avaliado à outras pessoas, quase a totalidade dos especialistas (95%) respondeu que sim.

No que diz respeito à nota dada ao ambiente, a média aritmética das notas atribuídas pelos avaliadores foi de 7,98, com mediana e moda 8, e desvio padrão de 1,53. No Gráfico 4 pode-se notar o agrupamento das notas atribuídas.

Gráfico 4 - Notas atribuídas pelos avaliadores



Fonte: Autor

Dentre os comentários feitos pelos avaliadores quando o questionário pedia para que se justificasse a nota dada, puderam ser destacados como pontos positivos:

- Acesso remoto de boa qualidade (não existem atrasos nas respostas às ações realizadas remotamente).
- A interface é prática e intuitiva.
- Permite autonomia do estudante.
- Não foram detectados erros durante a utilização.

Por outro lado, quando pedido que fossem levantados aspectos a serem melhorados, foram levantados os seguintes:

- Melhorar a interatividade da Interface do site.
- Incompatibilidade com alguns navegadores (Internet Explorer).
- Melhorar os tutoriais sobre os experimentos.
- Melhorar a disponibilidade dos conteúdos didáticos, não apresentando apenas em PDF.

5.5 VALIDAÇÃO EM CLASSES EXPERIMENTAIS

As avaliações de usabilidade foram realizadas em três escolas e três instituições de ensino superior. Participaram diretamente das aplicações 9 professores em torno de 300 estudantes.

5.4.1. Ensino Médio

Nas escolas de educação básica, as intervenções foram acompanhadas pela equipe do RExLab e o próprio professor da disciplina, que, após capacitações, realizou a aplicação com os estudantes em sala de aula ou como atividade extraclasse. Dentre as escolas que utilizaram os experimentos, estão:

- **Escola de Educação Básica Apolônio Ireno Cardoso:** um docente aplicou os experimentos Painel Elétrico CC e Painel Elétrico CA com 60 alunos em 3 turmas de física no 3º ano, e Condução de Calor em Barras Metálicas e Meios de Propagação de Calor com 95 alunos em 4 turmas de física no 2º ano.
- **Escola de Educação Básica Maria Garcia Pessi:** um professor utilizou o experimento Painel Elétrico CA com 324 alunos em turmas de física do 3º ano.

5.4.2. Ensino Superior

Nas instituições de ensino superior de Criciúma, Araranguá e São Paulo, as atividades foram assistidas apenas a distância e reportados os feedbacks dos docentes e estudantes em formulários no ambiente virtual de aprendizagem Moodle. Dentre as instituições que utilizaram os experimentos, estão:

- **Universidade Federal de Santa Catarina:** dois docentes utilizaram os experimentos Painel Elétrico CC e Painel Elétrico CA com 33 alunos na disciplina de Circuitos Elétricos nos cursos de Engenharia de Computação e Engenharia de Energia.
- **Pontifícia Universidade Católica de São Paulo:** um docente aplicou os experimentos Painel Elétrico CC e Painel Elétrico CA com 37 alunos em turmas de Laboratório de Física 3 e Circuitos Elétricos 2, nos cursos de Engenharia Elétrica e Engenharia de Produção.
- **Universidade Federal do Pará:** um docente utilizou os experimentos Painel Elétrico CC e Painel Elétrico CA com 16

alunos junto ao Programa de Educação Tutorial de Engenharia Elétrica.

- **Faculdade SATC:** dois docentes utilizaram os experimentos Painel Elétrico CC e Painel Elétrico CA com 58 alunos na disciplina de Medidas Elétricas no Curso Superior de Tecnologia em Automação Industrial.

6 CONSIDERAÇÕES FINAIS

Este trabalho apresentou uma iniciativa de desenvolvimento de um sistema de gerenciamento de experimentos remotos baseado em módulos, com garantia de acesso exclusivo. No decorrer do desenvolvimento desta pesquisa, pôde-se perceber a ausência de ferramentas flexíveis para o gerenciamento de experimentos remotos.

Assim, foi desenvolvida uma interface de criação e utilização de experimentos remotos e análise dos dados de utilização, com funcionalidades de criação e gerenciamento de usuários. Também foi criada uma interface de análise dos dados de utilização do sistema, e após a realização testes de performance e validação da plataforma, o código-fonte foi disponibilizado sob licença MIT em repositório público na internet.

A solução aqui descrita pode melhorar a experiência de uso de experimentos remotos, uma vez que há um maior controle sobre os usuários e o tempo utilizado para cada experimento. Além disso, a fila garante que um usuário não possa interferir em uma experiência em andamento, o que prejudicaria os resultados.

A utilização desta ferramenta vem atender a demanda crescente de usuários dos recursos disponíveis no REXLab, tornando o procedimento de utilização dos experimentos mais seguro e escalável.

O desenvolvimento de soluções de baixo custo, fazendo uso de ferramentas de hardware e de software aberto é importante quando consideramos as diferentes realidades na educação. A solução apresentada pode ser facilmente replicada ou personalizada por outros desenvolvedores ou instituições de ensino.

A partir dessas possibilidades, como trabalhos futuros se destacam a implementação de um sistema de filas utilizando *Websocket*, bem como uma API REST e *WebSocket* webservices. Estes aspectos poderiam tornar os experimentos disponíveis para um maior número de clientes, e consequentemente mais alunos.

Além disso, uma análise nos termos de usabilidade e experiência do usuário são necessárias, assim como melhorias de acessibilidade e criação de conteúdo multimídia integrado à página do experimento. Nesse sentido, como trabalho futuro pode ser implementado um estudo com os professores e alunos que utilizaram o sistema nas turmas experimentais.

REFERÊNCIAS

ACOLA. **STEM: Country Comparisons**. Australian Council of Learned Academies. Australia. 2013

AUER, M. et al. Distributed virtual and remote labs in engineering. Industrial Technology, 2003 IEEE International Conference on, 2003, 10-12 Dec. 2003. p.1208-1213 Vol.2.

BRASSCOM. **Estratégia TIC Brasil 2022**. Associação Brasileira de Empresas de Tecnologia da Informação e Comunicação. São Paulo. 2013

CONFEA, Conselho Federal de Engenheiros. A falta de engenheiros. 2015. Disponível em: <http://www.confea.org.br/cgi/cgilua.exe/sys/start.htm?infoid=15360&sid=1206>. Acesso em: 16 nov. .

COSTA, R. J.; ALVES, G. R. **Mobile Experimentation Innovating Education to the ‘Mobile Phone’ Generation**. European Conference on the Use of Modern Information and Communication Technologies. Gent 2006.

COSTA, R. J.; ALVES, G. R.; ZENHA-RELA, M. Reconfigurable IEEE1451-FPGA based weblab infrastructure. Remote Engineering and Virtual Instrumentation (REV), 2012 9th International Conference on, 2012, 4-6 July 2012. p.1-9.

FARIAS FILHO, M. C.; ARRUDA FILHO, E. J. M. **Planejamento da pesquisa científica**. São Paulo: Atlas, 2013. 154 ISBN 978-85-224-7626-8.

FIGUEIREDO, N. M. A. D. **Método e Metodologia na Pesquisa Científica**. 3 ed. São Caetano do Sul: Yendis Editora, 2009. ISBN 978-85-7728-085-8.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4. Atlas, 2002. 175 ISBN 85-224-3169-8.

GYÖRÖDI, C. et al. Web 2.0 Technologies with jQuery and Ajax. **Journal of Computer Science & Control Systems**, v. 2, n. 2, p. 11-15, 2009. ISSN 18446043. Disponível em: <

<http://search.ebscohost.com/login.aspx?direct=true&db=iih&AN=51003345&lang=pt-br&site=ehost-live> >.

HARWARD, V. J. et al. The iLab Shared Architecture: A Web Services Infrastructure to Build Communities of Internet Accessible Laboratories. **Proceedings of the IEEE**, n. June 2008, p. 19, 2008. Disponível em: < <http://ieeexplore.ieee.org/ielx5/5/4527074/04527087.pdf?tp=&arnumber=4527087&isnumber=4527074> >.

HE, R. Y. **Design and implementation of web based on Laravel framework 2014** International Conference on Computer Science and Electronic Technology (ICCSET 2014). ShenZhen: Atlantis Press 2014.

INEP, I. N. D. E. E. P. E. A. T. **Censo da Educação Superior 2012: Resumo Técnico**. INEP. Brasília. 2014

_____. **Censo Escolar 2014**. Brasília 2015.

KRASILCHIK, M. Reformas e realidade: o caso do ensino das ciências. **São Paulo em Perspectiva**, v. 14, p. 85-93, 2000. ISSN 0102-8839. Disponível em: < http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0102-8839200000100010&nrm=iso >.

LESCOURRET, F. et al. Modelling an information system using the MERISE method for agricultural research: The example of a database for a study on performances in dairy cows. **Agricultural Systems**, v. 38, n. 2, p. 149-173, 1992/01/01 1992. ISSN 0308-521X. Disponível em: < <http://www.sciencedirect.com/science/article/pii/0308521X9290038P> >.

LOWE, D.; MACHET, T.; KOSTULSKI, T. UTS Remote Labs, Labshare, and the Sahara Architecture. In: ZUBÍA, J. G. e ALVES, G. R. (Ed.). **Using Remote Labs in Education: Two Little Ducks in Remote Experimentation** Bilbao: University of Deusto, 2011. cap. 403, p.21. ISBN 978-84-9830-398-8.

LOWE, D. et al. Evolving Remote Laboratory Architectures to Leverage Emerging Internet Technologies. **Learning Technologies, IEEE Transactions on**, v. 2, n. 4, p. 289-294, 2009. ISSN 1939-1382. Disponível em: < <http://ieeexplore.ieee.org/ielx5/4620076/5370793/05210092.pdf?tp=&arnumber=5210092&isnumber=5370793> >.

LOWE, D.; OROU, N. Interdependence of booking and queuing in remote laboratory scheduling. *Remote Engineering and Virtual Instrumentation (REV)*, 2012 9th International Conference on, 2012, 4-6 July 2012. p.1-6.

MARCELINO, R. **Ambiente virtual de aprendizagem integrado a mundo virtual 3D e a experimento remoto aplicados ao tema resistência dos materiais**. 2010. (Doutorado). Programa de Pós-Graduação em Engenharia de Minas, Metalúrgica e de Materiais, Universidade Federal do Rio Grande do Sul, Porto Alegre.

NYSTRÖM, A. **Agile Solo: Defining and Evaluating an Agile Software Development Process for a Single Software Developer**. 2011. 37 (MSc.). Department of Computer Science and Engineering, University of Gothenburg, Göteborg, Sweden.

ODEH, S. et al. Experiences with deploying VISIR at Al-Quds University in Jerusalem. *Global Engineering Education Conference (EDUCON)*, 2014 IEEE, 2014, 3-5 April 2014. p.273-279.

ORDUÑA, P. et al. Adding New Features to New and Existing Remote Experiments through their Integration in WebLab-Deusto. **International Journal of Computer Engineering**, v. 7, p. 6, 2011. ISSN 1861-2121.

ORDUÑA, P. et al. Towards a microRLMS approach for shared development of remote laboratories. *Remote Engineering and Virtual Instrumentation (REV)*, 2014 11th International Conference on, 2014, 26-28 Feb. 2014. p.375-381.

ORNBO, G. **Node.js in 24 Hours**. Indianapolis: Sams, 2012.

PIMENTEL, V.; NICKERSON, B. G. Communicating and Displaying Real-Time Data with WebSocket. **Internet Computing, IEEE**, v. 16, n. 4, p. 45-53, 2012. ISSN 1089-7801. Disponível em: <<http://ieeexplore.ieee.org/ielx5/4236/6257353/06197172.pdf?tp=&arnumber=6197172&isnumber=6257353>>.

PREDKO, M. **Programming and customizing the 8051 microcontroller**. New York: McGraw-Hill, 1999.

ROCHADEL, W. **RExMobile: Integrando Experimentação Remota na Educação Básica**. 2013. 140 (Bacharel em Tecnologias da Informação e Comunicação). Universidade Federal de Santa Catarina, Araranguá.

SALZMANN, C. et al. The Smart Device specification for remote labs. Remote Engineering and Virtual Instrumentation (REV), 2015 12th International Conference on, 2015, 25-27 Feb. 2015. p.199-208.

SCHAF, F. M. **Arquitetura para ambiente de ensino de controle e automação utilizando experimentos remotos de realidade mista**. 2006. Universidade Federal do Rio Grande do Sul

SCHECKLER, R. K. Virtual labs: a substitute for traditional labs? **The International Journal of Developmental Biology**, v. 47, p. 5, 2003.

SCHWABER, K.; SUTHERLAND, J. The Definitive Guide to Scrum: The Rules of the Game. 2013. Disponível em: < <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf> >. Acesso em: 04/05.

SILVA, J. B. et al. Adaptation Model of Mobile Remote Experimentation for Elementary Schools. **Tecnologias del Aprendizaje, IEEE Revista Iberoamericana de**, v. 9, n. 1, p. 28-32, 2014. ISSN 1932-8540. Disponível em: < <http://ieeexplore.ieee.org/ielx7/6245520/6746056/06719587.pdf?tp=&number=6719587&isnumber=6746056> >.

SILVA, J. B. D. **A Utilização da Experimentação Remota como Suporte para Ambientes Colaborativos de Aprendizagem**. 2006. 196 (Doutorado em Engenharia e Gestão do Conhecimento). Programa de Pós-Graduação em Engenharia de Gestão do Conhecimento, Universidade Federal de Santa Catarina, Florianópolis.

SURGUY, M. History of Laravel PHP framework, Eloquence emerging. 2013. Disponível em: < <http://maxoffsky.com/code-blog/history-of-laravel-php-framework-eloquence-emerging/> >. Acesso em: 28 Set. 2015.

TEIXEIRA, P. **Professional Node.js: Building Javascript Based Scalable Software**. Indianapolis: John Wiley & Sons, 2013.

W3C. Browser Statistics and Trends. 2015. Disponível em: < http://www.w3schools.com/browsers/browsers_stats.asp >. Acesso em: 01 Out. 2015.

W3TECHS. Usage statistics and market share of JQuery for websites. 2015. Disponível em: < <http://w3techs.com/technologies/details/js-jquery/all/all> >. Acesso em: 29 Set. 2015.

YAOYE, L.; ESCHE, S. K.; CHASSAPIS, C. A scheduling system for shared online laboratory resources. Frontiers in Education Conference, 2008. FIE 2008. 38th Annual, 2008, 22-25 Oct. 2008. p.T2B-1-T2B-6.

APÊNDICE A – Formulário de Validação



LABORATÓRIO DE EXPERIMENTAÇÃO REMOTA
 Universidade Federal de Santa Catarina - UFSC
 Rua Pedro João Pereira, 150
 CEP 88905-120 - Araranguá/SC – Brasil
 +55 48 3721-4194 | rexlab@contato.ufsc.br
<http://rexlab.ufsc.br/>

Avaliação dos Experimentos

DADOS PESSOAIS

Nome: _____

Formação: _____

Instituição: _____ País: _____

Linhas de pesquisa: _____

Atua (podem ser assinaladas mais de uma opção):

Ensino Fundamental Ensino Médio Ensino Técnico Ensino Superior Pós-Graduação

Disciplina(s) que ministra: _____

Utiliza práticas em laboratório real? Sim Não.

Por que? _____

Já utilizou Experimentação Remota anteriormente a esta experiência?

Sim Não

AVALIAÇÃO

Quais experimentos remotos está avaliando?

Você recomendaria a utilização do Experimento Remoto para outros docentes?

Sim Não

Considerando uma escala de 0 a 10, qual nota você atribuiria ao Experimento Avaliado?

NOTA: _____

Caso a nota atribuída ao Experimento Remoto:

- a) Se encontre de 0 a 6, por favor apresente os aspectos principais que você identificou na avaliação. Sua resposta é muito importante para melhorarmos nosso serviço.

- b) Se encontre de 7 a 9, por favor indique recomendações para que possamos chegar a 10.

- c) Caso tenha identificado erros ou equívocos que podem influenciar o uso do experimento remoto ou favor apresentar sugestões ou críticas: