

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**UMA ARQUITETURA DE SENSORIAMENTO
REMOTO SENSÍVEL AO CONTEXTO PARA
IRRIGAÇÃO**

DISSERTAÇÃO DE MESTRADO

Rafael Boufleuer

Santa Maria, RS, Brasil

2015

UMA ARQUITETURA DE SENSORIAMENTO REMOTO SENSÍVEL AO CONTEXTO PARA IRRIGAÇÃO

Rafael Boufleuer

Dissertação apresentada ao Curso de Mestrado Programa de Pós-Graduação em Informática (PPGI), Área de Concentração em Computação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de
Mestre em Ciência da Computação

Orientadora: Prof^a. Dr^a. Iara Augustin

Santa Maria, RS, Brasil

2015

Boufleuer, Rafael

Uma arquitetura de sensoriamento remoto sensível ao contexto para irrigação / por Rafael Boufleuer. – 2015.

74 f.: il.; 30 cm.

Orientadora: Iara Augustin

Dissertação (Mestrado) - Universidade Federal de Santa Maria, Centro de Tecnologia, Programa de Pós-Graduação em Informática, RS, 2015.

1. Computação Pervasiva. 2. Computação Sensível ao Contexto. 3. Arquitetura. 4. Taxonomia de Contextos. 5. Medidor de Umidade e Pluviometria. 6. Open Hardware. 7. Irrigação. 8. Agricultura. I. Augustin, Iara. II. Título.

© 2015

Todos os direitos autorais reservados a Rafael Boufleuer. A reprodução de partes ou do todo deste trabalho só poderá ser feita mediante a citação da fonte.

E-mail: rafaboufler@redes.ufsm.br

**Universidade Federal de Santa Maria
Centro de Tecnologia
Programa de Pós-Graduação em Informática**

A Comissão Examinadora, abaixo assinada,
aprova a Dissertação de Mestrado

**UMA ARQUITETURA DE SENSORIAMENTO REMOTO SENSÍVEL AO
CONTEXTO PARA IRRIGAÇÃO**

elaborada por
Rafael Boufleuer

como requisito parcial para obtenção do grau de
Mestre em Ciência da Computação

COMISSÃO EXAMINADORA:

Iara Augustin, Dr^a.
(Presidente/Orientadora)

Benhur de Oliveira Stein, Dr. (UFSM)

Adenauer Correa Yamin, Dr. (UFPEL)

Santa Maria, 7 de Agosto de 2015.

AGRADECIMENTOS

Em primeiro lugar, gostaria de agradecer à minha família por todo o apoio que sempre foi demonstrado. Meus pais e irmãs incessantemente me deram forças e incentivo, servindo como base para a minha formação pessoal. Gostaria, também, de agradecer a minha orientadora Professora Iara Augustin e ao Professor João Carlos Damasceno Lima (famoso Caio), que me auxiliaram e incentivaram em vários momentos difíceis neste período. Além disso, quero agradecer aos Professores Celio e Benhur que também me auxiliaram muito.

Um forte abraço também para o pessoal do Sistema Irriga, especialmente ao Laudenir e ao Júnior, e aos Professores Reimar e Mirta. Agradeço ainda aos meus colegas Alfredo e Bruno, meu muito obrigado por todas as experiências compartilhadas, e aos meus colegas de apartamento: Tiago, Mateus, Rodrigo e Padilha, pelo grande apoio que me deram. Agradeço também ao pessoal do Núcleo de Ciência da Computação e do NTE da UFSM, pelas grandes amizades formadas, e pelos momentos de descontração que ficaram marcados. Enfim, agradeço a todos que de alguma forma me auxiliaram nesta caminhada. Por fim, agradeço ao Programa de Pós-Graduação em Informática e à Universidade Federal de Santa Maria pelo acolhimento e suporte, o que possibilitou o presente trabalho.

“O homem não é nada além daquilo que a educação faz dele.”
— IMMANUEL KANT

RESUMO

Dissertação de Mestrado
Programa de Pós-Graduação em Informática
Universidade Federal de Santa Maria

UMA ARQUITETURA DE SENSORIAMENTO REMOTO SENSÍVEL AO CONTEXTO PARA IRRIGAÇÃO

AUTOR: RAFAEL BOUFLEUER
ORIENTADORA: IARA AUGUSTIN

Local da Defesa e Data: Santa Maria, 7 de Agosto de 2015.

Considerando que a irrigação ocupa a maior porcentagem da água consumida na agricultura, e que o Brasil pretende expandir consideravelmente a sua área irrigada nas próximas décadas, há uma crescente necessidade de melhorar o manejo da água de irrigação, principalmente em regiões com reduzida disponibilidade de água ou ocorrência de períodos de deficiência hídrica. Por isso, o incremento de tecnologias na realização da irrigação utilizando-se de informações referentes ao ambiente, estão tornando-se importantes por permitirem a maximização do uso da água e energia, mantendo, ou mesmo melhorando, o rendimento e a qualidade da produção agrícola. Este trabalho propõe uma arquitetura de sensoriamento remoto sensível ao contexto para irrigação, e suas contribuições são: (a) o desenvolvimento de dois protótipos de um medidor de umidade e pluviometria baseados na arquitetura proposta utilizando tecnologias *open hardware*; (b) o desenvolvimento de uma taxonomia de contextos que define os tipos de informações que podem ser utilizadas na arquitetura dos protótipos; e (c) uma comparação entre dois tipos distintos de sensores de umidade do solo. Para a validação da arquitetura, foram realizados dois estudos de caso para verificar o correto funcionamento dos componentes da arquitetura, bem como a coleta de dados para efetuar a comparação entre um sensor resistivo de baixo custo produzido na Universidade Federal de Santa Maria, e um sensor de alta precisão e custo que utiliza a técnica de reflectometria no domínio da frequência (FDR) (CS616 - Campbell Scientific, Estados Unidos). Os resultados obtidos com os dados analisados foram satisfatórios, onde verificou-se que a arquitetura atende aos requisitos aos quais se propôs. Além disso, a comparação realizada apresentou coeficientes de determinação de até 95% entre os sensores resistivos de umidade do solo de baixo custo e os sensores de umidade do solo CS616 da Campbell®.

Palavras-chave: Computação Pervasiva. Computação Sensível ao Contexto. Arquitetura. Taxonomia de Contextos. Medidor de Umidade e Pluviometria. Open Hardware. Irrigação. Agricultura.

ABSTRACT

Master's Dissertation
Post-Graduate Program in Informatics
Federal University of Santa Maria

A CONTEXT-AWARE REMOTE SENSING ARCHITECTURE FOR IRRIGATION

AUTHOR: RAFAEL BOUFLEUER

ADVISOR: IARA AUGUSTIN

Defense Place and Date: Santa Maria, August 7st, 2015.

Considering that irrigation occupies the major percentage of consumed water in agriculture and that Brazil intends to expand considerably its irrigated area in the coming decades, there is a growing need in improving the irrigation water management, mostly in regions with limited availability of water or occurrences of hydric deficiency periods. Therefore, the increase of technologies for performing irrigation using environmental information is becoming important because they enable the maximization of water and energy consumption, maintaining or even improving the yield and quality of agricultural production. This work proposes a context-aware remote sensing architecture for irrigation and its contributions are: (a) the development of two prototypes of a moisture and precipitation meter based on the proposed architecture using open hardware technologies; (b) the development of a context taxonomy that defines the types of information that can be used in the prototype's architecture; and (c) a comparison between two different types of soil moisture sensors. For the validation of the architecture, two case studies were realized to verify the correct functioning of the architecture components, as well as a data collection was performed to make the comparison between a resistive low-cost sensor produced at the Federal University of Santa Maria, and a high-accuracy and high-cost frequency domain reflectometry (FDR) sensor (CS616 - Campbell Scientific, United States). The results obtained from the analyzed data were satisfactory, where it was verified that the architecture is viable, meeting the requirements to which it has proposed. In addition, the comparison performed showed a determination coefficient of up to 95 % between the low-cost resistive soil moisture sensors and the soil moisture sensors CS616 of Campbell®.

Keywords: Pervasive Computing. Context-aware Computing. Moisture and Pluviometry Meter. Architecture. Context Taxonomy. Open Hardware. Irrigation. Agriculture.

LISTA DE FIGURAS

Figura 2.1 – Artigos por ano referentes à sistemas sensíveis ao contexto. Adaptado de (HONG; SUH; KIM, 2009).	19
Figura 3.1 – Arquitetura proposta.	25
Figura 3.2 – Taxonomia de contextos.	27
Figura 3.3 – Distribuição dos componentes da arquitetura proposta.	28
Figura 4.1 – Protótipos do medidor de umidade e pluviometria.	29
Figura 4.2 – Sensores de umidade utilizados.	31
Figura 4.3 – Ligação dos sensores no Arduino.	31
Figura 4.4 – Casos de uso do sistema.	32
Figura 4.5 – Organização da EEPROM.	34
Figura 4.6 – Diagrama de transição de estados do protótipo.	35
Figura 4.7 – Diagrama de sequência do protótipo com a placa GSM/GPRS.	36
Figura 4.8 – Diagrama de transição de estados do servidor.	38
Figura 4.9 – Diagrama de sequência do servidor.	39
Figura 4.10 – Processo de comunicação entre o protótipo e o servidor.	40
Figura 4.11 – Coleta de dados na caixa de areia.	41
Figura 4.12 – Sensores de umidade instalados na camada de 10 centímetros.	43
Figura 4.13 – Sensores de umidade instalados na camada de 20 centímetros.	43
Figura 4.14 – Precipitações por dia.	44
Figura 4.15 – Comparação entre os pluviômetros.	44
Figura 4.16 – Comparação entre as temperaturas coletadas.	45
Figura 4.17 – Coeficientes de determinação dos sensores de umidade com regressão linear da camada de 10 centímetros.	47
Figura 4.18 – Coeficientes de determinação dos sensores de umidade com a rede Neural Multilayer Perceptron da camada de 10 centímetros.	48
Figura 4.19 – Ajuste da umidade do solo para o sensor SI da camada de 10 centímetros com regressão linear.	48
Figura 4.20 – Ajuste da umidade do solo para o sensor SI da camada de 10 centímetros utilizando informações das camadas de 10 e 20 centímetros com regressão linear.	49
Figura 4.21 – Coeficientes de determinação dos sensores de umidade com regressão linear da camada de 20 centímetros.	50
Figura 4.22 – Coeficientes de determinação dos sensores de umidade com a rede Neural Multilayer Perceptron da camada de 20 centímetros.	51
Figura 4.23 – Ajuste da umidade do solo para o sensor SI da camada de 20 centímetros com regressão linear.	51
Figura 4.24 – Ajuste da umidade do solo para o sensor SI da camada de 20 centímetros utilizando informações das camadas de 10 e 20 centímetros com regressão linear.	52
Figura 4.25 – Painel solar e bateria USB	54
Figura 4.26 – Painel solar para alimentação externa de energia.	55
Figura 4.27 – Regulador de tensão e bateria para alimentação externa de energia.	55
Figura 4.28 – Divisor de tensão utilizado para leitura da carga da bateria. Adaptado de (MARGOLIS, 2011).	56
Figura 4.29 – Monitoramento da carga da bateria do medidor.	57
Figura 5.1 – Visão conceitual do protótipo da aplicação Kilo.One. Adaptado de (KJAER, 2008).	59

Figura 5.2 – Arquitetura sensível ao contexto para a agricultura. Adaptado de (MOHAMMADRE-ZAEI; FATHI; ATTARZADEH, 2011)	60
Figura 5.3 – Interconexão entre os componentes utilizados no trabalho. Adaptado de (PRASAD et al., 2012).	61
Figura 5.4 – Interconexão entre os componentes do sistema utilizando o microcontrolador Arduino. Adaptado de (MILLS et al., 2013).	62
Figura 6.1 – Resultados do Alfa Seed Software (CERUTTI, 2014).	65

LISTA DE TABELAS

Tabela 4.1 – Versões testadas do Arduino.	30
Tabela 4.2 – Comparativo da corrente despendida do medidor de umidade e pluviometria.	53
Tabela 5.1 – Comparativo entre trabalhos relacionados e o presente trabalho.	62

LISTA DE QUADROS

4.1	Requisição GET enviada pelo medidor de umidade e pluviometria para o servidor.	40
4.2	Coeficientes de determinação dos sensores de umidade do solo da camada de 10 centímetros. . .	47
4.3	Coeficientes de determinação dos sensores de umidade do solo da camada de 20 centímetros. . .	50

LISTA DE ABREVIATURAS E SIGLAS

EEPROM	Electrically Erasable Programmable Read-Only Memory - Memória Somente de Leitura Programável Apagável Eletricamente
ETc	Crop Evapotranspiration - Evapotranspiração da Cultura
ETo	Reference Evapotranspiration - Evapotranspiração de Referência
FDR	Frequency Domain Reflectometry - Reflectometria no Domínio da Frequência
GPRS	General Packet Radio Service - Serviços Gerais de Pacote por Rádio
GSM	Global System for Mobile Communication - Sistema Global para Comunicações Móveis
HTTP	Hypertext Transfer Protocol - Protocolo de Transferência de Hipertexto
INMET	Instituto Nacional de Meteorologia
IP	Internet Protocol - Protocolo de Internet
Kc	Coefficient Crop - Coeficiente de Cultura
OMM	Organização Meteorológica Mundial
RTC	Real Time Clock - Relógio de tempo real
TCP	Transmission Control Protocol - Protocolo de controle de transmissão
TDR	Time Domain Reflectometry - Reflectometria no Domínio do Tempo

SUMÁRIO

1 INTRODUÇÃO	15
1.1 Objetivos	16
1.2 Estrutura do Trabalho	16
2 FUNDAMENTAÇÃO TEÓRICA	18
2.1 Computação Ubíqua e Pervasiva	18
2.1.1 Computação Sensível ao Contexto.....	18
2.2 Irrigação na Agricultura	19
2.2.1 Determinação de quando e quanto irrigar	20
2.2.1.1 Monitoramento da Planta	20
2.2.1.2 Monitoramento do Solo	21
2.2.1.3 Monitoramento do Clima	21
2.3 Sensoriamento Remoto	22
2.3.1 Sensores	22
2.3.2 Transmissão de dados.....	23
2.3.3 <i>Open Hardware</i>	23
2.4 Tecnologias <i>Open Source</i> envolvidas	24
3 ARQUITETURA PROPOSTA	25
3.1 Aquisição de dados	25
3.2 Normalização de dados	25
3.3 Informações de Contexto	26
3.3.1 Taxonomia de Contextos	26
3.4 Aplicação	27
3.5 Funcionamento da Arquitetura Proposta	27
4 DESENVOLVIMENTO E VALIDAÇÃO	29
4.1 Protótipos	29
4.2 Modelagem do sistema	31
4.2.1 Casos de uso do sistema	32
4.2.2 Coleta de dados e transmissão das informações do protótipo	33
4.2.3 Armazenamento das informações no servidor	37
4.2.4 Comunicação entre o protótipo e o servidor	39
4.3 Estudo de Caso 1	41
4.3.1 Umidade do solo com dados brutos	42
4.3.2 Fórmulas de ajuste da umidade do solo	45
4.3.3 Coeficientes de determinação do sensor da camada de 10 centímetros	46
4.3.4 Coeficientes de determinação do sensor da camada de 20 centímetros	49
4.4 Estudo de Caso 2	52
4.4.1 Painel solar e bateria USB	54
4.4.2 Painel solar e bateria para alimentação externa de energia	54
4.5 Problemas encontrados	57
5 TRABALHOS RELACIONADOS	59
6 CONSIDERAÇÕES FINAIS	63
6.1 Trabalhos futuros	64
6.2 Publicações	66
REFERÊNCIAS	69

1 INTRODUÇÃO

Os problemas relacionados aos recursos hídricos são uma das causas das limitações na agricultura, pois a mesma é a maior consumidora de água em nível mundial (mais de 70 %), fazendo com que o elemento água seja, ao mesmo tempo, o fator mais abundante e o mais limitante ao rendimento das culturas (RODRIGUES, 2004). Considerando que a irrigação ocupa a maior porcentagem da água consumida na agricultura, há um crescente interesse científico no potencial que a irrigação de precisão tem para melhorar a produtividade das culturas e aumentar a eficiência na utilização de recursos como a água e a energia na agricultura irrigada (DACCA-CHE et al., 2014). Esta irrigação deve ser determinada baseado em todas as informações que se tem do ambiente, como por exemplo as necessidades de água da planta cultivada, condições meteorológicas e características do solo, buscando determinar a quantidade e a hora certa para a irrigação, evitando desperdícios de água e/ou redução na produção por excesso de irrigação e perdas na produção devido à subirrigação (CARLESSO et al., 2007).

Hargreaves et al. (2000) destacam a irrigação como um dos principais instrumentos para manter o agricultor no campo, e possibilitar seu desenvolvimento econômico e social. Para que isso seja possível, é necessário que a pesquisa disponibilize novas e melhores opções de manejo que possam contribuir tanto para a conservação dos recursos naturais, quanto para o desenvolvimento sustentável (HOWELL et al., 2000). Entre as opções de manejo da irrigação, o monitoramento do solo é a metodologia utilizada neste trabalho, onde a quantidade de água e o momento da irrigação são determinados através da avaliação da umidade no solo. Neste sentido, diversos estudos têm sido realizados para avaliação e verificação de estimativas de umidade do solo à base do sensoriamento remoto (WAGNER et al., 2007; DE JEU et al., 2008; ALBERGEL et al., 2012; LAIOLO et al., 2014; HAULE; MICHAEL, 2014), que permite a obtenção e interpretação de dados à distância. Além disso, diversos trabalhos estão sendo realizados para a avaliação e comparação dos sensores resistivos e sensores FDR (SCUDIERO et al., 2012; BÖHME; BECKER; DIEKKRÜGER, 2013; ROTIMI OJO et al., 2015), mostrando em alguns casos uma boa correlação entre a resistividade do solo e medições de umidade realizadas (CALAMITA et al., 2012), demonstrando que os sensores resistivos de baixo custo podem ser uma alternativa viável para o monitoramento da umidade do solo.

Uma das alternativas para auxiliar no monitoramento de áreas irrigadas e no desenvolvimento das opções de manejo da irrigação com a utilização do sensoriamento remoto, é através

dos sistemas pervasivos ou sistemas sensíveis ao contexto, que possuem a capacidade de capturar e processar os dados de contexto disponíveis no ambiente, agindo e reagindo a estímulos dos mesmos (LIMA, 2013). Com a utilização destes sistemas na agricultura, é possível utilizar todas as informações disponíveis na lavoura para auxiliar no aprimoramento do manejo da água na irrigação, suprir as demandas das culturas e auxiliar no crescimento da produtividade no campo.

1.1 Objetivos

A computação ubíqua e pervasiva mostra-se como uma alternativa viável para fornecer uma nova abordagem no desenvolvimento de sistemas sensíveis ao contexto, utilizando todas as informações do ambiente para auxiliar no aumento da produtividade e na diminuição dos danos ao meio ambiente. Porém, dentro dos sistemas sensíveis ao contexto desenvolvidos, pode-se perceber uma lacuna a ser preenchida na agricultura (conforme seção 2.1.1), especialmente referente à irrigação na agricultura que é responsável por utilizar a maior quantidade de água consumida no setor agrícola. Por isso, faz-se necessário a utilização de novas tecnologias para tratar destes desafios relacionados a irrigação, tendo diversas culturas e climas envolvidos.

Com a finalidade de auxiliar na resolução destas questões, o presente trabalho tem por objetivo o desenvolvimento de um medidor de umidade e pluviometria baseado em uma arquitetura de sensoriamento remoto sensível ao contexto utilizando tecnologias *open hardware*, bem como de uma taxonomia de contextos que define os tipos de informações que podem ser utilizadas na arquitetura dos protótipos deste medidor. Além disso, são realizadas comparações entre os sensores utilizados para disponibilizar uma solução de sensoriamento remoto de custo reduzido, que possa auxiliar na realização de uma irrigação localizada e precisa.

Com isso, busca-se o aumento da produção, melhorias nas técnicas de manejo e racionalização do consumo de água e energia na agricultura.

1.2 Estrutura do Trabalho

O trabalho está estruturado da seguinte forma. No capítulo 2 é apresentada a fundamentação teórica do trabalho, que contem os principais conceitos referentes à computação ubíqua e pervasiva, bem como sobre os sistemas sensíveis ao contexto, explicando suas origens e diferenças. Além disso, são apresentados os principais conceitos relacionados à irrigação na

agricultura e ao sensoriamento remoto, demonstrando a importância de ambos para o aumento da produção agrícola. A arquitetura proposta no trabalho e o fluxo das informações entre os seus componentes são explicados no capítulo 3. Além disso, a taxonomia de contextos que é utilizada para a classificação das informações e o funcionamento da arquitetura proposta também são explicados nesse capítulo.

O capítulo 4 apresenta o desenvolvimento dos protótipos, além dos estudos de caso que foram realizados para verificar o funcionamento dos protótipos, e posterior comparação dos valores coletados pelos sensores para efetuar a sua validação. O capítulo 5 apresenta os trabalhos relacionados. Nesse capítulo é realizada uma análise comparativa com os trabalhos relacionados, analisando os aspectos tratados por cada um deles. O capítulo 6 tece as considerações finais, trabalhos futuros e publicações.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Computação Ubíqua e Pervasiva

Na percepção de Weiser (1991), a computação deveria se tornar integrada à vida do usuário, ajustando-se às necessidades do mesmo, fazendo parte de todos os objetos que nos cercam, tornando-se a "tecnologia que desaparece", diferentemente da situação atual, onde o usuário deve aprender a lidar com o computador e suas aplicações. O objetivo da computação ubíqua é tornar a interação homem-máquina transparente e integrada com diversos dispositivos computacionais interconectados, integrando a computação com as ações e comportamentos das pessoas (LIMA, 2013).

Na literatura existem os dois termos: Computação Ubíqua e Computação Pervasiva. Porém, segundo Augustin et al. (2004)

estes termos não são sinônimos, mas propostas de uma visão futura de computação, a segunda sendo mais próxima que a primeira considerando o momento de desenvolvimento tecnológico atual. Pode-se dizer que a computação pervasiva, associada a outras tecnologias como a computação de vestir e redes inteligentes, convergirá para a computação ubíqua.

Nesta concepção, a computação pervasiva fornece ao computador a capacidade de obter informações do ambiente no qual ele está inserido (BALDAUF; DUSTDAR; ROSENBERG, 2007), podendo utilizá-las para configurar ou ajustar as aplicações e melhor atender as necessidades dos dispositivos ou do usuário (LIMA, 2013).

2.1.1 Computação Sensível ao Contexto

O termo sensível ao contexto na computação pervasiva foi introduzido por Schilit et al. (1994). Porém, contexto é um termo amplo, e possui diversas definições. Para Augustin et al. (2004) contexto é "toda a informação relevante para a aplicação, que pode ser obtida por ela mesma". Para Chen et al. (2000) contexto é "um conjunto de configurações e estados que determinam o comportamento do ambiente e a importância que um evento tem para o usuário". Mas a definição mais utilizada para contexto é a de Dey et al. (2000) que consideram contexto como

qualquer informação que pode ser utilizada para caracterizar a situação de uma entidade. Uma entidade é uma pessoa, um lugar ou um objeto que é considerado relevante para a interação entre o usuário e a aplicação, incluindo o próprio usuário e a aplicação.

Na pesquisa realizada por Hong et al. (2009), foram analisados 237 trabalhos relaciona-

dos a sistemas sensíveis ao contexto, e a figura 2.1 ilustra o crescimento do número de trabalhos realizados no período analisado e o interesse científico pela área. Pode-se observar que o número de trabalhos em 2007 já era 7 vezes maior em comparação com os trabalhos encontrados no ano 2000. Porém, no trabalho de Hong et al. (2009), não foram citados trabalhos referentes à implementação de sistemas ou arquiteturas sensíveis ao contexto para a agricultura, demonstrando que, até o momento da pesquisa realizada por ele, a agricultura não recebia a devida atenção na área.

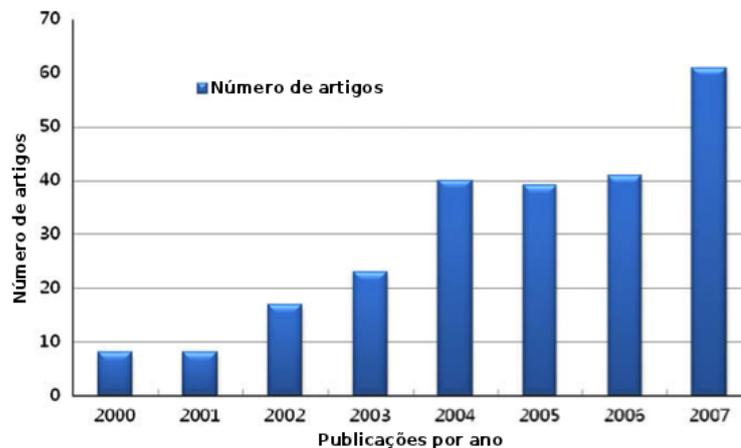


Figura 2.1: Artigos por ano referentes à sistemas sensíveis ao contexto. Adaptado de (HONG; SUH; KIM, 2009).

Entretanto, a agricultura pode ser considerada uma das áreas em que o contexto desempenha um papel importante, pois trata-se de transformar as preocupações agrícolas fundamentais que seguem abordagens tradicionais, para um modelo de contexto governado por um canal de entradas e saídas de sensores e atuadores (KJAER, 2008). Por isso, conhecer os contextos da propriedade monitorada, pode auxiliar na construção de aplicações de computação pervasiva para apoiar o trabalho diário, por exemplo, facilitando o acesso à informação para o agricultor. No capítulo 5 são analisados mais detalhadamente alguns trabalhos relacionados ao presente trabalho que foram produzidos recentemente, demonstrando o crescente interesse na agricultura nos últimos anos.

2.2 Irrigação na Agricultura

As atuais inovações tecnológicas têm favorecido a agricultura, tornando possível o monitoramento de variáveis meteorológicas em tempo real, com grande precisão e acurácia dos dados, possibilitando melhorias na eficiência dos sistemas de produção (CARLESSO et al.,

2007). Dentre os setores da agricultura, a irrigação é a principal consumidora de água em escala global. Porém, ela pode ser considerada como uma das principais ferramentas para aumentar a produção no campo (RODRIGUES, 2004). Segundo Allen et al (1998), sempre que a água proveniente das chuvas não for suficiente para atender a necessidade hídrica das plantas, será necessário realizar a irrigação para compensar as demandas destas culturas.

Na agricultura, a aquisição automática de dados tem favorecido o monitoramento de variáveis ligadas à planta, ao solo e ao planejamento da atividade agrícola, as quais estão direta ou indiretamente ligadas às melhorias na eficiência dos sistemas de produção como um todo. Uma importante ferramenta recentemente incorporada ao processo é a estação meteorológica automática, cuja principal vantagem é a frequência dos registros diários, o armazenamento dos dados medidos e a possibilidade de transmissão dos mesmos para visualização em tempo real. Além disso, a coleta, o armazenamento e transmissão automática de dados diminuem os erros oriundos de leitura, interpretação e digitação desses dados (CARLESSO; PETRY; TROIS, 2009). Essas estações meteorológicas podem ser encontradas dentro das propriedades como parte do sistema produtivo, ou fazendo parte de redes de estações públicas como o INMET (Instituto Nacional de Meteorologia), que é membro da Organização Meteorológica Mundial (OMM) que visa a interligação das redes meteorológicas para contribuições na climatologia (INMET, 2014), ou estações particulares como o Sistema Irriga, que oferece um conjunto de serviços de manejo e monitoramento para irrigação (IRRIGA, 2014).

2.2.1 Determinação de quando e quanto irrigar

A determinação de quando irrigar e da quantidade de água a ser aplicada na irrigação, pode ser realizada através do monitoramento da planta, do solo ou do clima (CARLESSO et al., 2007). Estas alternativas de manejo são apresentadas nas seções 2.2.1.1, 2.2.1.2 e 2.2.1.3.

2.2.1.1 Monitoramento da Planta

O monitoramento da planta, entre as três formas de monitoramento, pode ser considerado um dos mais eficientes no ponto de vista científico. Entre os diversos métodos de monitoramento das plantas têm-se as medições do potencial hídrico, da temperatura das folhas, entre outros. Porém, suas implicações operacionais e limitações de aplicabilidade dificultam a sua utilização no campo, uma vez que são poucas as informações disponíveis referentes aos limites e índices recomendáveis para a maioria das culturas (MANTOVANI; BERNARDO; PA-

LARETTI, 2007).

2.2.1.2 Monitoramento do Solo

O monitoramento do solo é uma metodologia usualmente utilizada para o manejo da irrigação, onde o momento da irrigação e a quantidade de água a ser aplicada são determinados através da avaliação do teor de água no solo. Esta determinação pode ser efetuada com a retirada de amostras do solo, para posteriormente efetuar a definição da umidade do mesmo. Porém, por questões operacionais, esta opção normalmente não é viável (MANTOVANI; BERNARDO; PALARETTI, 2007). Outra opção para a determinação do teor de água no solo é através da utilização de tensiômetros ou sensores para a medição da umidade solo. Entretanto, estes equipamentos utilizados devem ser capazes de capturar dados com precisão e com bom tempo de resposta, devido a necessidade de dados periódicos e representativos do teor de água no solo de uma determinada região (CARLESSO et al., 2007).

Esta é a opção de monitoramento utilizada no presente trabalho, pois apresenta uma boa relação custo-benefício entre o gasto com equipamentos e com a precisão em relação a seus métodos de medição da umidade do solo. A descrição dos equipamentos utilizados para a coleta e transmissão de dados neste trabalho são detalhados no capítulo 4.

2.2.1.3 Monitoramento do Clima

Atualmente, existe um crescente interesse no controle da irrigação através do uso de dados meteorológicos. Entretanto, o manejo da irrigação a partir de dados meteorológicos implica que estes sejam representativos de uma propriedade ou região, e que sejam diariamente coletados. De acordo com Elliott et al. (2000), uma importante aplicação dos dados meteorológicos em tempo real é a estimativa da evapotranspiração de referência (ET_o) para diferentes culturas ou regiões de abrangência. O processo da evapotranspiração é composto pelos fenômenos da evaporação e da transpiração, e geralmente é medida em milímetros por unidade de tempo. A evaporação consiste na passagem da água do estado líquido para o gasoso a partir de uma massa contígua de água (lagos, rios) ou de uma superfície úmida (solo, folhas da planta). A transpiração é a evaporação da água que foi utilizada nos processos necessários ao crescimento e desenvolvimento das plantas (CARLESSO et al., 2007).

A evapotranspiração de referência (ET_o) é definida como a evapotranspiração de uma cultura bem adaptada às condições específicas de uma região e com condições hídricas óti-

mas para o seu desenvolvimento (MANTOVANI; BERNARDO; PALARETTI, 2007). Este conceito foi introduzido para estudar a demanda evaporativa da atmosfera, independente de cultura, época do ano e características do solo, sendo utilizada para fins comparativos e, por isso, é denominada evapotranspiração de referência. Posteriormente, para calcular a estimativa da evapotranspiração das culturas (ETc), são utilizados os valores da evapotranspiração de uma cultura de referência (ETo) calculada por meio de dados meteorológicos da região, e de um coeficiente relativo à cultura, chamado de coeficiente de cultura (Kc) (ALLEN et al., 1998). Através do cálculo da ETo e posterior estimativa da ETc, é possível obter grande precisão com dados representativos de uma região.

Por isso, pretende-se utilizar esta opção de monitoramento nos trabalhos futuros do presente trabalho, para aprimorar os resultados apresentados ao produtor rural e auxiliar ainda mais na realização da irrigação (conforme apresentado na seção 6.1).

2.3 Sensoriamento Remoto

Diversas tecnologias estão sendo utilizadas para aumentar a produção agrícola e diminuir o impacto que a mesma causa no meio ambiente (ZHAO; LIU, 2012). A irrigação de precisão juntamente com as tecnologias de sensoriamento remoto, estão sendo utilizadas buscando estes objetivos. Através do sensoriamento remoto, é possível a obtenção e interpretação de dados a distância, possibilitando o aprimoramento do controle da produção agrícola (LIAGHAT; BALASUNDRAM, 2010).

2.3.1 Sensores

Um sensor é um dispositivo que responde a um estímulo físico/químico ou de algum processo tal como, calor, pressão ou movimento, de maneira específica e mensurável (AHMED, 2012). Nos medidores de umidade e pluviometria propostos, são utilizados sensores para medição da temperatura e umidade do solo e um pluviômetro. A temperatura do solo é uma informação importante pois influencia diretamente na germinação das sementes, na taxa de crescimento das plantas e nas condições para a colheita das culturas (PESSL, 2014), e o pluviômetro é importante por permitir o monitoramento das precipitações ocorridas.

Para mensurar a umidade do solo neste trabalho, são utilizados sensores resistivos e capacitivos. Os sensores resistivos medem os níveis de umidade baseados na resistência elétrica

do solo. A medida que o nível de água aumenta no solo, a resistência do solo diminui, ou seja, a condutância entre as hastes de metal aumenta (MA et al., 2011). No caso dos sensores capacitivos, a umidade do solo é determinada pela medição da constante dielétrica do solo que é ligada aos teores de umidade do mesmo. As técnicas mais utilizadas para esse tipo de medição são a reflectometria no domínio do tempo (TDR) e a reflectometria no domínio da frequência (FDR), e ambas utilizam a propriedade capacitiva para medir o nível de umidade no solo (AHMED, 2012). Os sensores de umidade do solo utilizados neste trabalho são apresentados na figura 4.2.

Normalmente considera-se na academia que os sensores resistivos possuem uma acurácia reduzida ou que tiveram seu interesse reduzido pelo avanço de outros métodos (ROBINSON et al., 2008). Entretanto, a diferença de preços também é significativa, sendo os sensores resistivos mais acessíveis. Se for realizada uma boa calibração destes sensores resistivos, é possível utilizá-los como uma alternativa para viabilizar um medidor de umidade e pluviometria de custo reduzido.

2.3.2 Transmissão de dados

Independentemente das técnicas de irrigação utilizadas, os sistemas de irrigação normalmente estão distantes das cidades, com infraestrutura de telecomunicação limitada e com serviços de alto custo. Isso dificulta a maioria dos agricultores a realizarem o monitoramento e controle remoto dos sistemas de irrigação (MILLS et al., 2013). Entretanto, com a utilização de tecnologias de transmissão de dados como a GPRS (*General Packet Radio Service* - Serviços Gerais de Pacote por Rádio), juntamente com a rede GSM (*Global System for Mobile Communication* - Sistema Global para Comunicações Móveis), é possível realizar a transmissão de dados no campo com um custo reduzido.

2.3.3 *Open Hardware*

É um conceito que vem sendo utilizado nos últimos anos, consistindo em compartilhar a estrutura de objetos físicos para a comunidade de uma forma semelhante ao software *open source* (HARNETT, 2011). O direito de usar, modificar, redistribuir e fabricar comercialmente ou como uma organização sem fins lucrativos, é concedido a todos, sem qualquer compromisso financeiro ou taxa (JUNIOR; NUNES; CELINSKI, 2013).

A solução *open hardware* utilizada no trabalho é a plataforma Arduino (ARDUINO, 2014). Entre algumas das características da plataforma, encontra-se a sua versatilidade, e o seu

microcontrolador que pode ser programado utilizando a linguagem de programação Arduino na sua plataforma de desenvolvimento *open source* (ZACHARIADIS; KASKALIS, 2012). Diversos sistemas já estão sendo desenvolvidos baseados nesta plataforma (SALAZAR et al., 2013; THALHEIMER, 2013; JUNIOR; NUNES; CELINSKI, 2013) pela necessidade da utilização da tecnologia no campo, buscando o desenvolvimento de novas técnicas que possibilitem o aumento da produção e a diminuição do uso de recursos do meio ambiente.

2.4 Tecnologias *Open Source* envolvidas

Esta seção apresenta as tecnologias utilizadas no presente trabalho para o desenvolvimento do servidor utilizado na camada de *Aplicação* da arquitetura proposta.

O software utilizado para criação do servidor foi o Oracle VM VirtualBox, que é um software *open source* de virtualização multiplataforma para computadores baseados na plataforma Intel e AMD, que possibilita a virtualização baseada em software através da instalação de diversos sistemas operacionais (VirtualBox, 2014). Dentre as vantagens do VirtualBox pode-se citar a criação de máquinas virtuais que não danificam ou modificam o sistema operacional utilizado na máquina real, além de permitir a restauração da máquina virtual para um momento estável na ocorrência de algum problema. O sistema operacional utilizado na máquina virtual do servidor é o Ubuntu, que é um sistema operacional *open source* baseado em Linux (Ubuntu, 2014).

O *framework* utilizado para projetar e desenvolver o servidor foi o Django (Django, 2014). Django é um projeto *open source* escrito em Python, que possui uma extensa comunidade, bem como uma estrutura base para criação e organização de projetos estruturados que auxilia na integração entre projetos e aplicações de terceiros. O *framework* foi utilizado pois facilita a integração com o banco de dados e com o servidor Apache. O servidor *web open source* utilizado foi o servidor Apache (Apache, 2014), pois é o servidor *web* mais utilizado na atualidade e possui integração com o *framework* Django, sendo responsável pelas requisições *web*. Já o banco de dados utilizado para o armazenamento das informações foi o PostgreSQL (PostgreSQL, 2014), que é um sistema de banco de dados objeto-relacional *open source*, e foi utilizado pois possui integração com o *framework* Django e pode ser executado na maioria dos sistemas operacionais, incluindo o Ubuntu que foi utilizado no servidor.

3 ARQUITETURA PROPOSTA

A arquitetura proposta neste trabalho se baseia nos conceitos referentes ao *middleware* sensível ao contexto proposto por Del Fabro Neto et al. (2013), e utiliza as duas camadas inferiores do mesmo respectivamente como as camadas de *Aquisição de dados* e *Normalização de dados* do presente trabalho. Porém, além de estender este *middleware*, são incorporadas mais duas camadas: *Informações de Contexto* e *Aplicação*. Nas próximas subseções, serão introduzidas as camadas da arquitetura proposta (Figura 3.1).

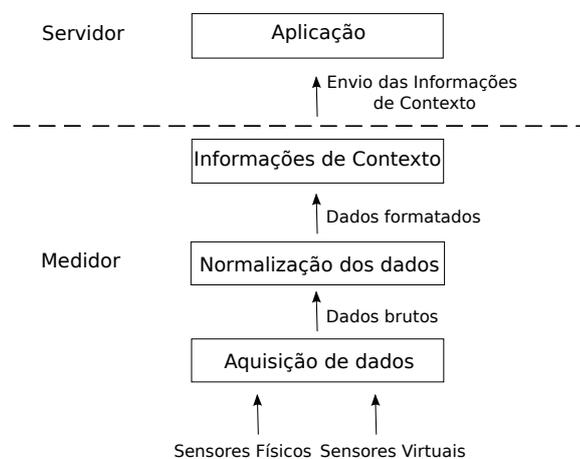


Figura 3.1: Arquitetura proposta.

3.1 Aquisição de dados

O principal objetivo da camada de *Aquisição de dados* é a coleta de dados, tanto dos sensores físicos, quanto dos sensores virtuais. Os dados referentes aos sensores virtuais podem ser coletados por alguma outra fonte de informação de contexto que não seja dos sensores físicos (por exemplo, informações de data e hora coletadas da rede GSM para atualizar o RTC (*real-time clock*)) sempre que o Arduino for reiniciado. Os dados coletados na camada de *Aquisição de dados* são repassados para a camada de *Normalização dos dados*.

3.2 Normalização de dados

A camada de *Normalização de dados* é responsável pela formatação dos dados brutos em um formato definido para a utilização nas camadas superiores. A camada normalizadora de dados necessita conhecer o formato das mensagens recebidas provenientes de sensores que

podem ser de diferentes fabricantes e tipos. Dessa forma, para acoplar um sensor de um novo tipo, fabricante ou modelo (ou seja, informações não reconhecíveis pela camada *Normalização de dados*), é necessário que seja desenvolvido um *parser* específico para obter as informações fornecidas pelo sensor. Após esta formatação, estes dados são repassados para a camada de *Informações de Contexto*.

3.3 Informações de Contexto

A camada *Informações de Contexto* é responsável por receber os dados formatados da camada de *Normalização de Dados*. Posteriormente, é realizada a classificação dos dados em informações de contexto, bem como o seu armazenamento (cartão de memória SD ou EEPROM) para manter um histórico das informações. Estas informações podem ser classificadas de diversas formas, sendo divididas em três categorias principais: Contextos de Monitoramento, Contextos Espaço-Temporais e Contextos de Auto-Monitoramento (Figura 3.2).

Após este processo de classificação das informações, estas são repassadas para a camada de *Aplicação*, e podem ser utilizadas para o monitoramento de diversas áreas da propriedade.

3.3.1 Taxonomia de Contextos

Na taxonomia de contextos proposta neste trabalho apresentada na figura 3.2, as informações de contexto são classificadas de diversas formas, sendo divididas em três categorias principais: *Contextos de Monitoramento*, *Contextos Espaço-Temporais* e *Contextos de Auto-Monitoramento*. O *Contexto de Monitoramento* é dividido em diversos *Contextos Ambientais*: *Contexto Hidrológico*, *Contexto do Solo* e *Contexto Climático*. No *Contexto Hidrológico*, as informações são referentes aos dados coletados pelos sensores de pluviometria, o *Contexto do Solo* é representado pelas informações de umidade e temperatura do solo e no *Contexto Climático* a temperatura do ambiente. No *Contexto de Monitoramento*, ainda pode ser inserido um componente para representação das informações de *Contexto da Planta* contendo as informações da cultura.

Nos *Contextos Espaço-Temporais* são classificadas as informações de data e tempo de transmissão de dados representadas pelo *Contexto de Transmissão*, que são importantes para a determinação de quando e onde um evento ocorreu. O *Contexto de Auto-Monitoramento* possui o *Contexto de Energia*, onde são classificadas todas as informações referentes à utilização de

energia de equipamentos, permitindo a detecção de problemas nos mesmos (carga da bateria, por exemplo). A taxonomia proposta foi criada de forma genérica, para que seja possível a inclusão de novos componentes.

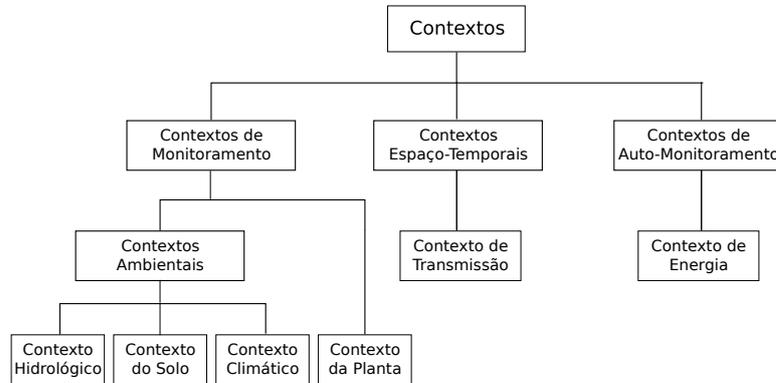


Figura 3.2: Taxonomia de contextos.

3.4 Aplicação

A camada de *Aplicação* localiza-se no servidor, e é responsável por receber as informações da camada *Informações de contexto* e efetuar o armazenamento destas informações no banco de dados do servidor. Nesta camada pode ser feita a disponibilização das informações através de gráficos para facilitar a compreensão e auxiliar o produtor a realizar a irrigação de forma localizada e precisa.

3.5 Funcionamento da Arquitetura Proposta

A figura 3.3 ilustra o funcionamento e a distribuição dos componentes da arquitetura proposta. No exemplo, os quatro medidores de umidade e pluviometria, que estão localizados em regiões distintas da propriedade, realizam a coleta dos dados referentes a umidade e temperatura do solo e da pluviometria de cada região.

A camada de *Aquisição de dados* da arquitetura é representada pela coleta de dados dos sensores realizada individualmente em cada medidor de umidade e pluviometria. Após a coleta, a normalização dos dados é realizada em cada medidor na segunda camada da arquitetura, e posteriormente estes dados são classificados em informações de contexto na terceira camada. Estas informações são armazenadas temporariamente nos protótipos para manter um histórico das mesmas, e são enviadas ao servidor, onde a camada de *Aplicação* é responsável

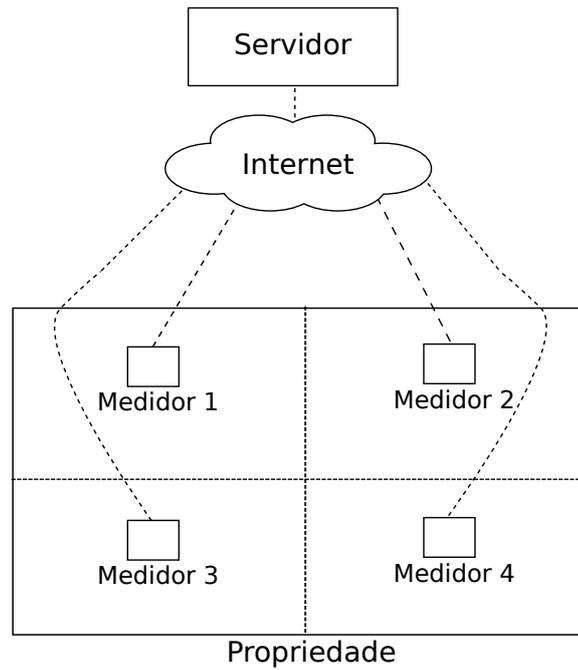


Figura 3.3: Distribuição dos componentes da arquitetura proposta.

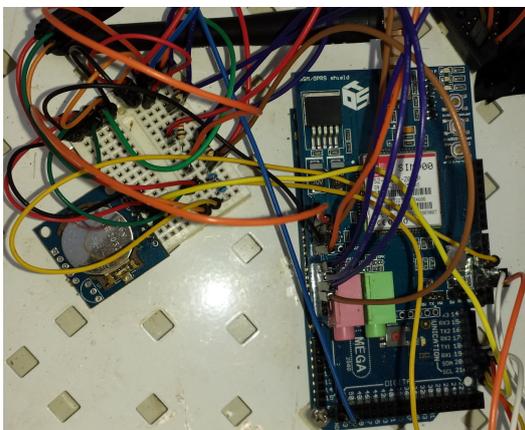
por receber estas informações e armazená-las no banco de dados. O tempo de coleta e o envio das informações pode variar de acordo com a necessidade.

4 DESENVOLVIMENTO E VALIDAÇÃO

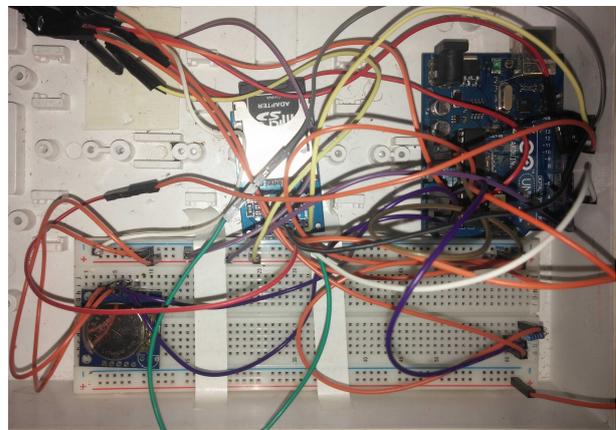
Este capítulo detalha o desenvolvimento dos protótipos do medidor de umidade e pluviometria desenvolvidos neste trabalho, suas características e componentes. Além disso, a modelagem do sistema é descrita, onde são explicados como ocorrem os processos de coleta de dados e transmissão das informações pelo protótipo com a placa GSM/GPRS, até o armazenamento das informações no servidor. Posteriormente, são apresentados os estudos de caso realizados para a validação da arquitetura proposta, bem como as comparações realizadas entre os dados coletados dos sensores utilizados.

4.1 Protótipos

Na figura 4.1 são ilustrados os protótipos do medidor de umidade e pluviometria propostos. Na figura 4.1 a) é apresentado o protótipo que contém o Arduino com a placa GSM/GPRS conectados (a direita na figura), juntamente com RTC DS1307 localizado a esquerda na figura (DALLAS, 2014). Para o protótipo proposto é utilizada uma placa GSM/GPRS SIM 900 para a transmissão dos dados (ICOMSAT, 2014). Na figura 4.1 b) é apresentado um protótipo alternativo que utiliza o Arduino (a direita na figura) juntamente com o RTC e uma placa para leitura e escrita em um cartão de memória SD (no centro na figura). Este protótipo foi projetado para locais de difícil acesso, onde a rede GSM não está disponível. Neste caso, é necessário que as informações sejam coletadas manualmente do cartão de memória SD para posteriormente armazená-las no servidor.



(a) Medidor com a placa GSM/GPRS.



(b) Medidor com o leitor SD.

Figura 4.1: Protótipos do medidor de umidade e pluviometria.

O protótipo com a placa GSM/GPRS que utiliza o Arduino Mega 2560 é ilustrado na figura 4.1a. Entre os modelos pesquisados, o Arduino Mega é o modelo mais extensível, pois possui mais entradas analógicas e digitais e foi utilizado no protótipo por permitir a utilização da placa GSM/GPRS, e ao mesmo tempo, permitir a utilização de diversos sensores (ARDUINO, 2014). Como a placa GSM/GPRS fica acoplada a diversos pinos do Arduino, ela ocupa diversas entradas com interrupções, e por isso, através do Arduino Mega, foi possível utilizar o pluviômetro diretamente em uma entrada que possibilita a utilização de interrupções, que não seria possível com os outros modelos testados. Além disso, possui uma memória flash superior aos outros modelos, permitindo programas maiores e mais elaborados.

O protótipo que realiza o armazenamento das informações no cartão SD é apresentado na figura 4.1b, e utiliza o Arduino Uno por possuir entradas analógicas e digitais suficientes. A tabela 4.1 ilustra as versões de microcontroladores Arduino testadas no escopo deste trabalho (ARDUINO, 2014).

Tabela 4.1: Versões testadas do Arduino.

	Arduino Uno	Arduino Leonardo	Arduino Mega 2560
Processador	ATmega328	ATmega32u4	ATmega2560
Voltagem de operação / voltagem de entrada	5 V/7-12 V	5 V/7-12 V	5 V/7-12 V
Flash [KB]	32	32	256
Entradas analógicas	6	12	16
In/Out digitais	14	20	54
EEPROM [KB]	1	1	4

Nos medidores de umidade e pluviometria propostos, são utilizados sensores para medição da temperatura e umidade do solo e um pluviômetro. O sensor utilizado para a medição da temperatura do solo é o Waterproof DS18B20, e o pluviômetro utilizado no trabalho é o modelo WS1080. Os sensores de umidade do solo utilizados na comparação realizada são apresentados na figura 4.2, sendo o sensor CS616 da Campbell o sensor branco na parte superior da figura, e o sensor resistivo de umidade do solo SI (Sistema Irriga) fabricado na Universidade Federal de Santa Maria, o sensor localizado na parte inferior da figura. O sensor capacitivo CS616 da CAMPBELL (CS616, 2014) foi utilizado como sensor de referência para efetuar a comparação com o sensor resistivo SI.

Os sensores de umidade do solo SI e o sensor de temperatura do solo são conectados no Arduino conforme ilustrado na figura 4.3, entretanto, o sensor de temperatura utiliza um

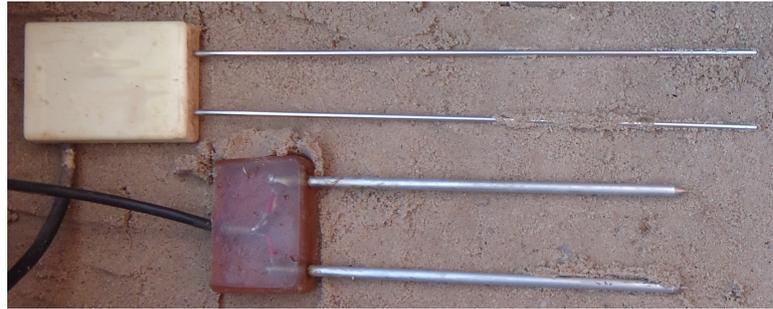


Figura 4.2: Sensores de umidade utilizados.

resistor de 120 Ohms. O pluviômetro é ligado diretamente em um pino digital do Arduino, e utiliza interrupções para contabilizar a ocorrência de chuva. Cada ocorrência detectada pelo pluviômetro contabiliza uma certa quantidade de água em milímetros (mm), totalizando ao final quanto choveu em um determinado período.

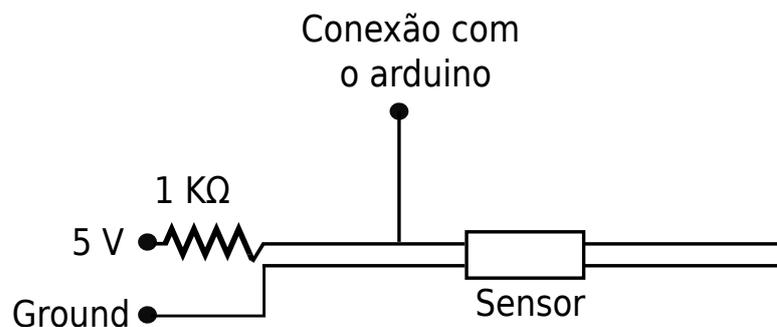


Figura 4.3: Ligação dos sensores no Arduino.

4.2 Modelagem do sistema

A modelagem de um sistema é a parte central das atividades que são relevantes para a sua implantação. Estes modelos são importantes para visualizar e controlar a sua arquitetura, possibilitando a melhor compreensão do sistema desenvolvido (BOOCH; RUMBAUGH; JACOBSON, 2006).

A modelagem do sistema para a arquitetura proposta é apresentada nesta seção, onde primeiramente, são discutidos quais são os casos de uso desenvolvidos, verificando a importância de cada um deles. Posteriormente, o processo de coleta de dados e transmissão das informações do protótipo que utiliza a placa GSM/GPRS é analisado. São apresentados o diagrama de transição de estados do protótipo, a organização do armazenamento das informações na EEPROM e o diagrama de sequência das informações. Por fim, no processo de armazenamento das informações no servidor, são apresentados o diagrama de transição de estados e o diagrama de

sequência das informações.

4.2.1 Casos de uso do sistema

Um caso de uso representa a interação dos atores com o próprio sistema e suas principais abstrações (BOOCH; RUMBAUGH; JACOBSON, 2006). Estas interações são funções que são utilizadas para visualizar, especificar e construir o comportamento pretendido do sistema. Na figura 4.4 são apresentados os casos de uso do sistema.

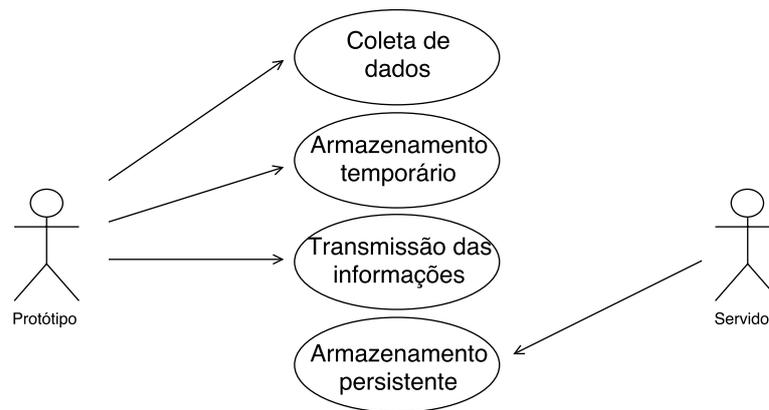


Figura 4.4: Casos de uso do sistema.

Com o protótipo como ator, têm-se três casos de uso:

- *Coleta de dados*: Neste caso de uso, o protótipo realiza a coleta de dados dos sensores físicos e virtuais, como por exemplo, umidade e temperatura do solo, pluviometria, entre outros;
- *Armazenamento temporário*: Neste caso de uso, as informações são armazenadas temporariamente no cartão SD, ou na EEPROM do RTC utilizado conforme descrito na seção 4.2.2;
- *Transmissão das informações*: Neste caso de uso, as informações devem ser enviadas através da placa GSM/GRPS para o servidor, conforme processo de comunicação descrito na seção 4.2.2.

Com o servidor como ator, têm-se o seguinte caso de uso:

- *Armazenamento persistente*: O servidor neste caso de uso, realiza o armazenamento das informações recebidas do protótipo no banco de dados conforme seção 4.2.3, para que posteriormente estas informações sejam utilizadas pela camada de aplicação.

4.2.2 Coleta de dados e transmissão das informações do protótipo

O código desenvolvido para a coleta de dados e transmissão das informações foi implementado na linguagem de programação do Arduino, que é baseada na linguagem C/C++ e preserva sua sintaxe na declaração de variáveis, operadores e estrutura. O código desenvolvido para o medidor que efetua o armazenamento das informações no cartão SD é uma versão simplificada do código do protótipo com a placa GSM/GPRS, pois não possui as funções referentes a gravação de dados na EEPROM, ligar/desligar a placa GSM/GPRS e o envio de informações para o servidor.

Primeiramente, um programa preliminar é executado para inicializar as variáveis da EEPROM do RTC que possui 32 Kilobytes (kB), onde é realizado o armazenamento das informações. A figura 4.5 ilustra a configuração da EEPROM, que é realizada por um programa preliminar responsável por: i) zerar o local onde é armazenada a posição inicial do último conjunto de informações coletadas (posições 0 e 1); ii) zerar a quantidade de informações ainda não enviadas para o servidor (posições 2 e 3); iii) e definir o endereço de início da gravação das informações na EEPROM (posição 4).

Como cada conjunto de informações possui um tamanho fixo, as informações são armazenadas a partir do endereço 4, e as próximas posições a serem gravadas são definidas pelo tamanho do conjunto de dados (como o tamanho é 20 bytes, o segundo conjunto de dados será gravado a partir da posição 24). No momento em que não houver mais espaço para a gravação das informações (quando o último endereço gravado somado com o tamanho das informações para a próxima gravação for maior que 32 Kb), as informações passam a ser gravadas no início da EEPROM novamente (posição 4), garantindo a circularidade de gravação na EEPROM. Esta característica da circularidade é importante, pois garante que, se houver algum problema com o fornecimento de energia do protótipo e o Arduino for reiniciado ou desligado por um tempo, assim que o protótipo for ligado, o mesmo irá seguir coletando informações, e enviando possíveis dados ainda não enviados ao servidor.

O tamanho de cada conjunto de dados é de 20 bytes:

- Temperatura do solo (4 bytes) e carga da bateria (4 bytes) do tipo *float*;
- Timestamp (log de tempo – data e hora em segundos - 4 bytes) e Umidade do solo (4 bytes) do tipo *int*. Os valores das entradas analógicas do Arduino variam na faixa de 0 a 1023, pois o conversor dos valores das entradas analógicas do Arduino é de 10 bits

(ARDUINO, 2014), onde cada sensor de umidade necessita de 10 bits ($2^{10} = 1024$) para o armazenamento de seus dados. Com três sensores de umidade do solo, soma-se 30 bits dos 32 que possui uma variável do tipo *int* com 4 bytes (1 byte = 8 bits). Isto é feito através do deslocamento de bits dentro dos 4 bytes, e se isto não fosse realizado, seriam necessários 12 bytes para armazenar as informações dos três sensores de umidade do solo (4 bytes cada sensor);

- Pluviômetro (4 bytes) do tipo *float*, pois contabiliza as batidas de uma gangorra que correspondem as ocorrências de chuva do pluviômetro. Cada ocorrência é referente a um determinado volume de água em milímetros, por exemplo, se cada ocorrência equivale a 0,4 milímetros, três ocorrências contabilizam 1,2 mm de chuva.

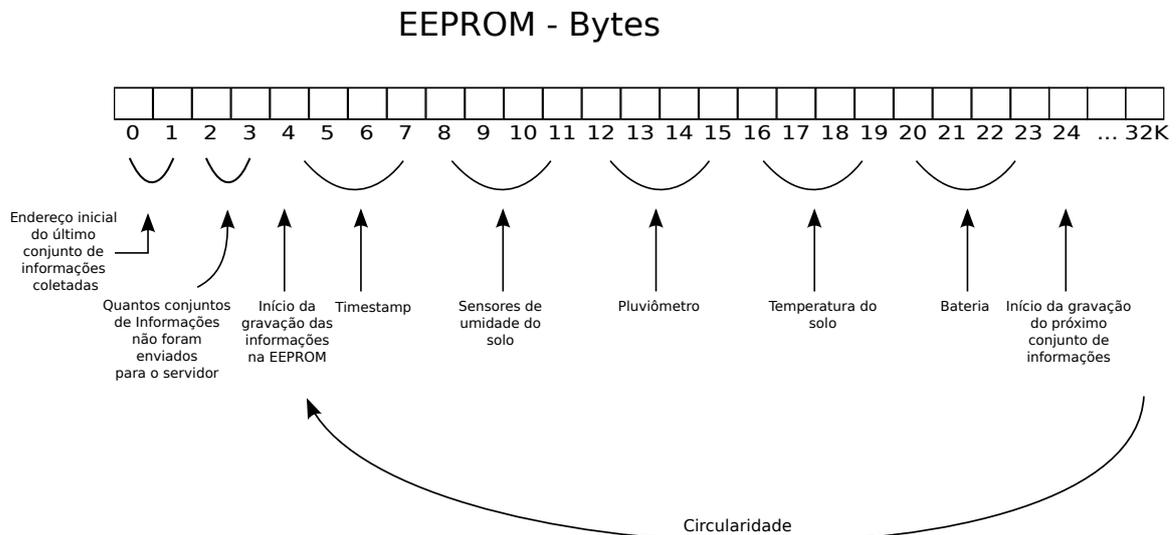


Figura 4.5: Organização da EEPROM.

Após o programa preliminar ter sido executado, o programa principal é inserido e executado no Arduino. A figura 4.6 ilustra o diagrama de transição de estados deste programa, que realiza a coleta dos dados e a transmissão das informações do medidor de umidade e pluviometria que utiliza a placa GSM/GPRS para o servidor. Nesta sequência, cada um dos estados representa uma condição ou situação de um evento sendo realizado. Após a realização dos eventos de cada estado, ocorre a transição (relacionamento entre dois estados) quando as condições especificadas para esta transição estão satisfeitas (BOOCH; RUMBAUGH; JACOBSON, 2006). A execução destes estados sempre se repete, com a exceção de que as interrupções ocasionadas pelas ocorrências do pluviômetro, podem acontecer em qualquer um dos estados para contabilizar a chuva ocorrida, independente do estado atual. Após a inicialização do Arduino, no estado

coletando dados é realizada a coleta de dados dos sensores físicos e virtuais, e posteriormente ocorre o armazenamento das informações na EEPROM no estado *armazenando informações*. Com o armazenamento concluído, são realizadas as tentativas de transmitir as informações no estado *transmitindo informações*, e após as tentativas, independente do sucesso ou falha no envio das informações, o protótipo entra no estado *esperando*, para aguardar o tempo de espera determinado, voltando ao início do ciclo para a coleta de dados.

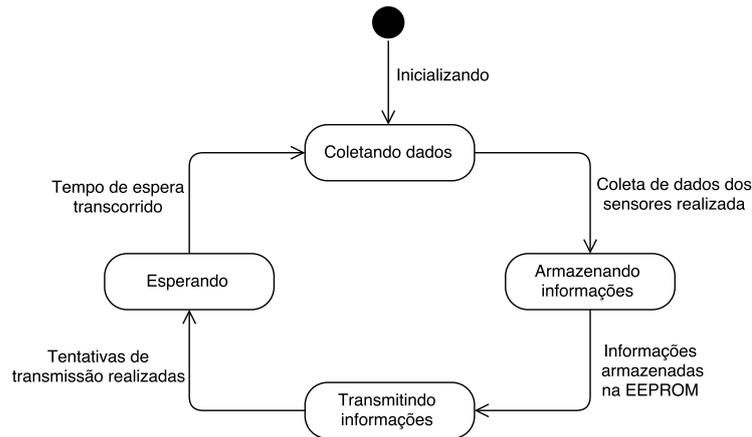


Figura 4.6: Diagrama de transição de estados do protótipo.

O diagrama de sequência das informações do programa principal é apresentado na figura 4.7. Um diagrama de sequência tem por objetivo ilustrar a ordenação temporal das mensagens na comunicação efetuada, em ordem crescente no tempo de cima para baixo (BOOCH; RUMBAUGH; JACOBSON, 2006). Após a inicialização do Arduino, o programa entra em um *loop* para a execução de diversas etapas que são executadas continuamente, formando um *loop* infinito da etapa final que é a *espera*, voltando para a etapa de *coleta de dados* que é a primeira etapa do *loop*. É nesta primeira etapa que é realizada a coleta de dados de todos os sensores, que posteriormente são normalizados e classificados em informações de contexto. Estas informações são *armazenadas* na EEPROM, para que as mesmas estejam disponíveis caso ocorra algum problema no envio das informações ao servidor.

Em seguida, a tentativa de *transmissão* das informações é iniciada, onde primeiramente a carga da bateria é verificada, demonstrado no diagrama através de um operador de sequência. Os operadores de sequência demonstram a execução concorrente de várias sequências de mensagens, e são apresentados como um retângulo no diagrama, possuindo uma *tag* (rótulo de texto no canto superior esquerdo) para informar o tipo de operador de controle (alt, loop, entre outros). A *tag alt* presente no operador de sequência para verificação da carga da bateria,

significa uma execução condicional. Esta execução é dividida em várias sub-regiões por linhas horizontais tracejadas, sendo que cada região representa um ramo de uma condição. Se a carga da bateria for menor que o valor limite estabelecido, as informações não são enviadas, e a placa GSM/GPRS permanece desligada pelo tempo determinado até a próxima leitura de dados, e com isso, o Arduino entra no estado de *espera*. Isto é efetuado para reduzir o gasto de energia quando a bateria estiver com níveis baixos de carga. Se o valor da carga da bateria for superior ao limite, o protótipo é autorizado a iniciar as tentativas de conexão. Este teste é realizado apenas quando o protótipo está utilizando uma bateria e um painel solar, conforme testes realizados e apresentados na seção 4.4.

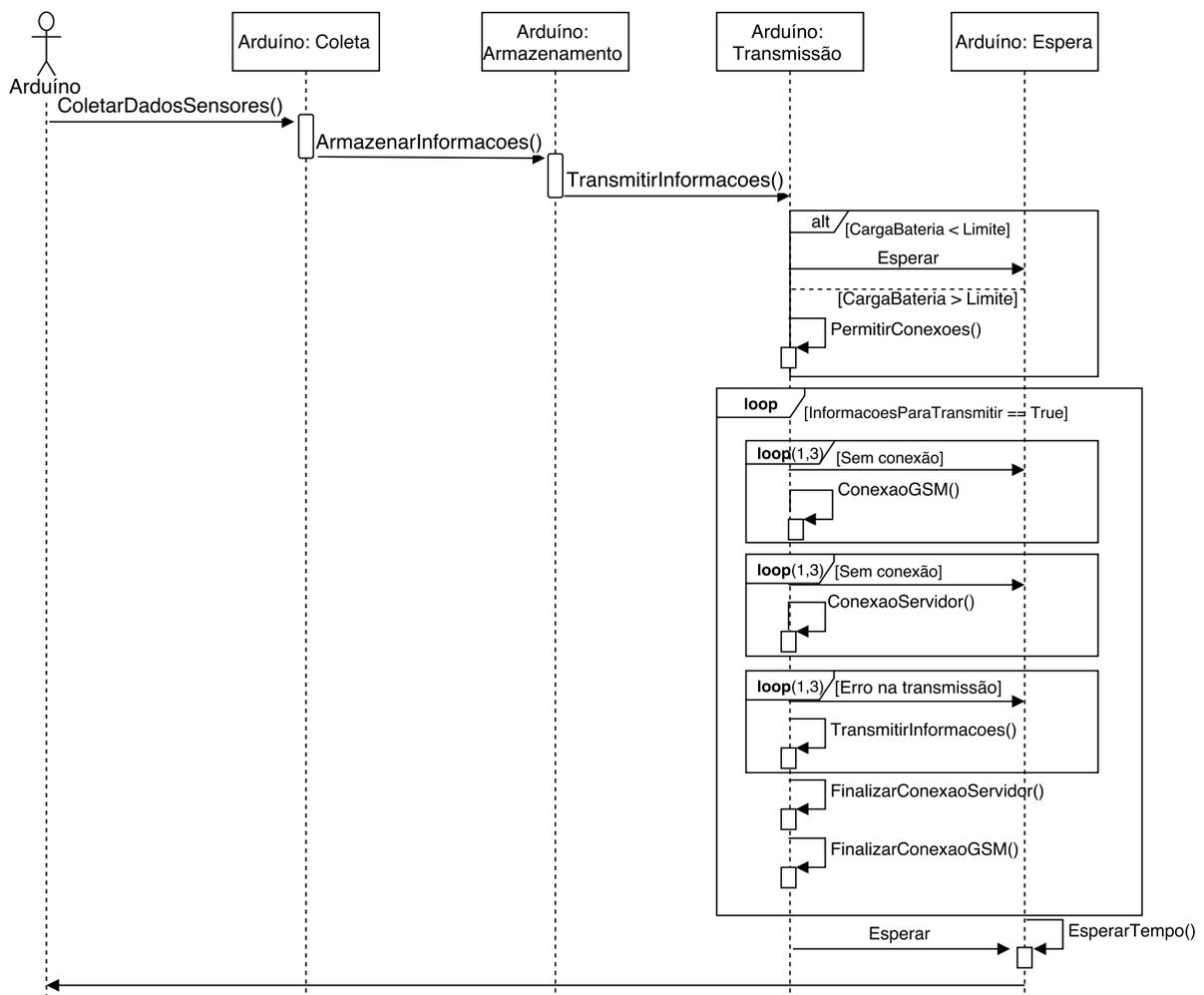


Figura 4.7: Diagrama de sequência do protótipo com a placa GSM/GPRS.

Com esta condição satisfeita, o primeiro *loop* é executado enquanto existirem informações a serem enviadas. Em seguida, a placa GSM/GPRS é ligada e o protótipo tenta efetuar a conexão com a rede GSM, conforme segundo *loop* ilustrado na figura 4.7. A *tag loop* é uma execução iterativa, e o corpo do *loop* é executado repetidamente enquanto a condição é verda-

deira (BOOCH; RUMBAUGH; JACOBSON, 2006). Nesta *tag*, o primeiro número é o mínimo de vezes que o *loop* será executado, e o segundo é o número máximo de tentativas (1,3). O protótipo tenta realizar a conexão com a rede GSM três vezes, e se não obtiver sucesso, a placa GSM/GPRS é desligada e o Arduino entra em *espera*, aguardando o tempo determinado para executar o *loop* inicial novamente, efetuando a coleta de dados. O mesmo acontece se o protótipo não conseguir efetuar a conexão com o servidor no terceiro *loop*, que é o passo seguinte a ser executado, caso a conexão com a rede GSM for efetuada com sucesso.

Se as conexões com a rede GSM e com o servidor forem realizadas, as informações referentes a uma coleta de dados (informações de todos os sensores em um determinado momento) são enviadas para o servidor no quarto *loop*. Se algum erro acontecer, a tentativa de envio é realizada novamente por até três vezes, e se em nenhuma das tentativas o envio for realizado, a placa GSM/GPRS é desligada, e o protótipo entra em estado de *espera*, aguardando a execução do *loop* inicial do programa a partir da coleta de dados, conforme apresentado no final do diagrama de sequência 4.7.

Com o envio de um conjunto de informações efetuado, as conexões com o servidor e com a rede GSM são finalizadas, e se houverem mais informações a serem enviadas para o servidor, ou seja, se a condição *InformacoesParaTransmitir==True* presente no *loop* inicial de *transmissão* for verdadeira, as conexões são efetuadas novamente para o envio das informações. Ao final, se todos os conjuntos de informações forem enviados com sucesso, a placa GSM/GPRS é desligada para diminuir o consumo de energia do medidor de umidade e pluviometria, e após o tempo determinado de *espera*, o *loop* inicial é executado novamente, a partir da coleta de dados.

4.2.3 Armazenamento das informações no servidor

A camada de Aplicação da arquitetura proposta é representada pelo servidor na figura 3.3, onde o servidor tem a responsabilidade de receber as informações enviadas pelos medidores de umidade e pluviometria, e armazená-las no banco de dados. A figura 4.8 apresenta o diagrama de transição de estados do servidor.

Após a inicialização, o primeiro estado em que o servidor fica é *aguardando conexão*. Este estado permanece constante até que alguma conexão seja estabelecida, e com isso, o servidor passa para o estado *aguardando requisição*, até o momento que o cliente (protótipo) envie uma requisição com as informações. Se nenhuma requisição for recebida dentro de um tempo

determinado, a conexão é *finalizada* e o servidor volta para o estado *aguardando conexão*. Se a requisição for recebida, as informações são verificadas, e se estiverem no formato certo, são armazenadas no estado *armazenando informações no BD* (banco de dados). Depois deste estado, o servidor volta a aguardar uma nova requisição por um tempo determinado, e se não receber nenhuma requisição com informações, ou se o protótipo solicitar o fim da conexão, o servidor passa para o estado *finalizando conexão*, e volta para o estado *aguardando conexão*.

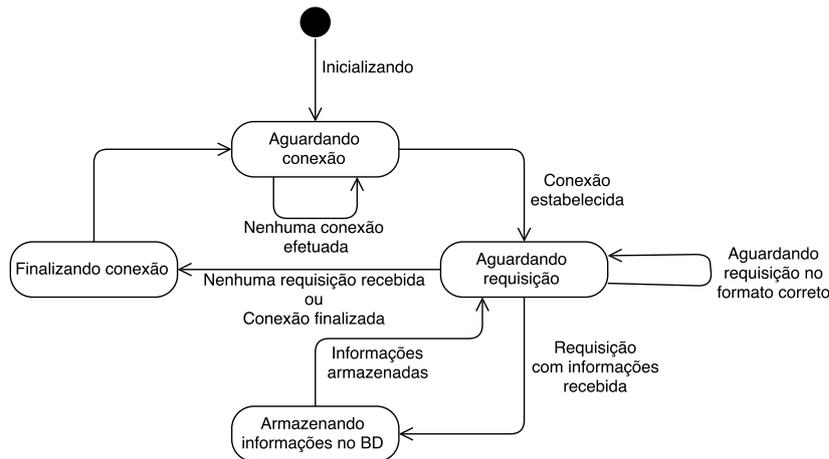


Figura 4.8: Diagrama de transição de estados do servidor.

O Diagrama de sequência das informações da figura 4.9 demonstra como é efetuado o armazenamento das informações no servidor. Primeiramente, o medidor de umidade e pluviometria estabelece a conexão com o servidor informando o IP e a porta do mesmo pela rede GSM, conforme descrito na seção 4.2.4. Se a conexão for estabelecida com sucesso, o servidor aguarda uma requisição enquanto o tempo de espera for menor que um *timeout* estabelecido, conforme apresentado no operador de sequência com um *loop* ($\text{TempoEspera} < \text{Timeout}$). Se a requisição for recebida, o servidor verifica se todas as informações foram recebidas corretamente e estão na formatação correta, e realiza armazenamento no banco de dados, enviando também um retorno para o medidor de umidade e pluviometria, informando que o recebimento das informações ocorreu com sucesso. Após o armazenamento, o *loop* é executado novamente com o tempo de espera zerado, e o servidor aguarda por novas requisições. Se o tempo de espera for maior que o *timeout* estabelecido, conforme segunda parte do operador de sequência com a tag *alt*, a conexão é finalizada e o servidor volta a aguardar novas conexões.

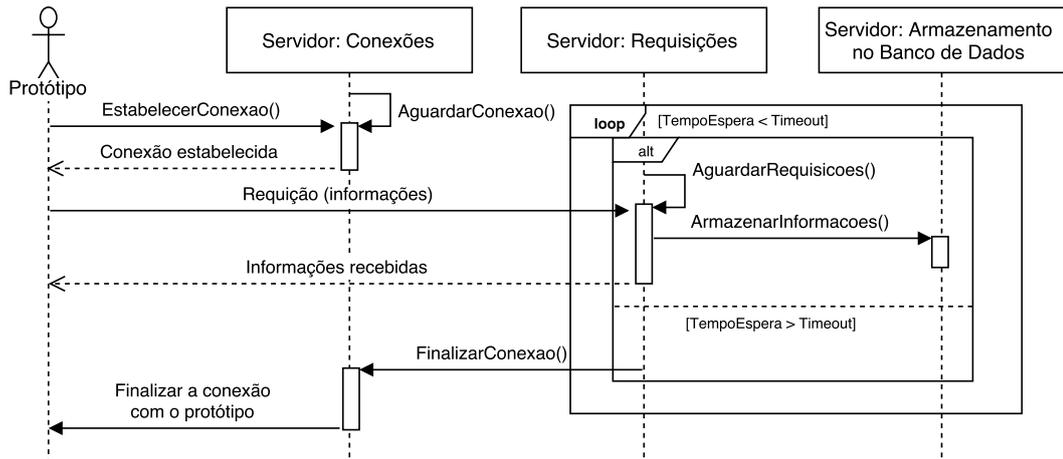


Figura 4.9: Diagrama de sequência do servidor.

4.2.4 Comunicação entre o protótipo e o servidor

A placa GSM/GPRS utilizada funciona através de comandos AT (SIM900, 2010), que é uma linguagem de comandos orientada por linhas que realiza a conexão da placa com a rede GSM. Primeiramente, é realizada uma conexão TCP (*Transmission Control Protocol* - Protocolo de controle de transmissão) que é um protocolo que possibilita a transmissão de dados pela rede utilizando-se uma conexão entre a origem e o destino das informações (KUROSE et al., 2010). Posteriormente, as informações são enviadas através método GET do protocolo HTTP (*Hypertext Transfer Protocol* - Protocolo de Transferência de Hipertexto), contendo as informações dos sensores de umidade do solo, do *timestamp* que representa o momento da coleta dos dados, pluviômetro, temperatura do solo e carga da bateria, conforme quadro 4.1. A comunicação efetuada entre o protótipo e o servidor para o envio das informações com sucesso, pode ser observada na figura 4.10.

Este processo de conexão do protocolo TCP é conhecido como *3 way handshake connection* (conexão com aperto de mão de três vias). O cliente (protótipo) envia um pacote (estrutura de transmissão de dados em uma rede) com a *flag SYN* marcada, e o servidor responde com um pacote com a *flag SYN/ACKnowledge* marcada. Então, o cliente recebe este pacote e envia uma resposta com a *flag ACK* marcada. Com isso, a conexão foi estabelecida entre o cliente e o servidor, conforme ilustrado nas três primeiras linhas da figura 4.10. Após a conexão estabelecida, o protótipo envia as informações através do método GET do protocolo HTTP, conforme a quarta linha da figura que representa o envio das informações. Após o ponto de interrogação (?) da quarta linha, são informados os parâmetros e os seus respectivos valores, conforme quadro

4.1, que representa o modelo de comunicação do protocolo HTTP. A quinta linha representa a resposta com o código padrão do protocolo HTTP, informando que as informações foram recebidas com sucesso pelo servidor. É também neste retorno que o servidor pode informar ao protótipo alguma falha ou erro ocorrido.

Depois que as informações foram enviadas, o cliente envia um pacote com a *flag FIN* marcada para finalizar a conexão TCP. O servidor ao receber esta mensagem, responde com um pacote com a *flag FIN/ACK* marcada e o cliente finaliza a conexão com um pacote com a *flag ACK* (KUROSE et al., 2010).

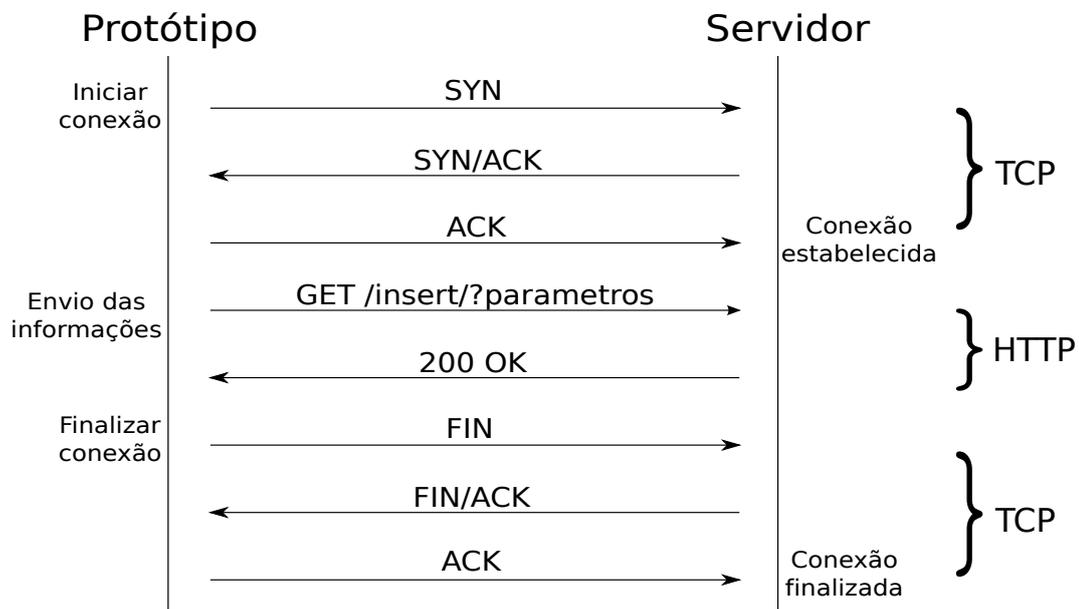


Figura 4.10: Processo de comunicação entre o protótipo e o servidor.

A seguir, são listadas as mensagens da comunicação:

- SYN - Inicia uma conexão;
- ACK - Certificação que recebeu o último pacote ou uma resposta;
- FIN - Termina uma conexão.

```
insert/?soil1=750&soil2=760&soil3=0&timestamp=1410275383&rain_gauge=0
&temperature=27.0&battery=11.5
```

Quadro 4.1: Requisição GET enviada pelo medidor de umidade e pluviometria para o servidor.

4.3 Estudo de Caso 1

Nesse estudo de caso, foi realizada a coleta de dados dos sensores conectados ao protótipo (sensores de temperatura e umidade do solo e pluviômetro) e de sensores de umidade do solo CS616 conectados a um *datalogger* CR1000 da Campbell, que foram utilizados para a comparação dos valores em relação aos sensores SI de umidade do solo. Além disso, foram utilizados dados da temperatura ambiental e da pluviometria coletados pela estação INMET localizada em Santa Maria, e de um pluviômetro do Sistema Irriga. Os sensores de temperatura e umidade do solo foram instalados dentro de uma caixa de areia com dois furos no fundo para a vazão da água. Os testes foram realizados na areia, pois a mesma possui uma boa infiltração, sendo uma boa alternativa para realização dos testes.

A caixa foi instalada em uma área rural na Universidade Federal de Santa Maria, onde foram efetuadas cerca de 2000 coletas de dados durante 3 semanas. O pluviômetro efetuou a coleta dos dados referente as chuvas ocorridas neste período, o sensor de temperatura do solo foi instalado horizontalmente em uma profundidade de cerca de 5 centímetros para coleta da temperatura do solo, e os sensores de umidade do solo ficaram dispostos em duas camadas, onde a primeira camada é mais superficial a cerca de 10 centímetros e a segunda camada a cerca de 20 centímetros da superfície da caixa de areia. Em cada uma das camadas, foram instalados um sensor CS616 e um sensor SI para efetuar a comparação entre os dois sensores de umidade. Na coleta de dados o protótipo foi ligado diretamente na rede de energia elétrica, para desconsiderar problemas com gasto de energia de bateria. O protótipo pode ser visualizado na figura 4.11.



Figura 4.11: Coleta de dados na caixa de areia.

4.3.1 Umidade do solo com dados brutos

Esta seção apresenta os gráficos gerados a partir dos dados brutos dos sensores SI normalizados, em comparação com os sensores CS616 da Campbell. Os valores dos sensores CS616 foram coletados com um *datalogger CR1000* da Campbell, que disponibiliza os dados referentes a umidade do solo na faixa de valores entre 0 e 1 (quanto mais próximo de 1, mais úmido está o solo). Os valores obtidos dos sensores de umidade SI através das saídas analógicas do Arduino, variam em uma faixa de valores de 0 a 1023 (por que o conversor dos valores das entradas analógicas do Arduino é de 10 bits), sendo necessário efetuar a normalização destes valores para a escala entre 0 e 1 para efetuar uma comparação direta com os sensores da Campbell. Os gráficos apresentaram variações similares, porém foi necessário efetuar uma inversão de 180 graus nos valores dos sensores SI para facilitar a comparação, pois inversamente aos sensores CS616, apresentam valores mais próximos de 0 quando o solo está úmido. A normalização e inversão dos dados dos sensores SI foi realizada conforme equação 4.1:

$$UF = 1 - \frac{U}{L} \quad (4.1)$$

Onde UF é a umidade do solo final, U é a umidade do solo coletada e L é o valor máximo que pode ser obtido para a umidade do solo (1023 como leitura da entrada analógica do Arduino). Os valores resultantes da equação variam entre [0, 1]: quanto mais próximo de 0, menos quantidade de água encontra-se no solo; quanto mais próximo de 1, maior a quantidade de água no solo. Os valores do pluviômetro do presente trabalho, foram normalizados entre 0,00 e 0,100 apenas para facilitar o entendimento do comportamento dos sensores nos gráficos com dados referentes a umidade do solo.

A coleta de dados foi realizada quatro vezes por hora e posteriormente foi efetuada uma média destes valores para geração dos gráficos de comparação. Pode-se observar nas figuras 4.12 e 4.13 que os sensores de umidade localizados na camada superior variaram mais que os sensores localizados na camada inferior. Isto acontece por que a água provinda das chuvas penetra com mais facilidade na parte superior do solo, e a medida que a profundidade aumenta, a retenção de água diminui.

A figura 4.14 apresenta os valores das precipitações por dia coletados pelo pluviômetro do presente trabalho, e a figura 4.15 ilustra a comparação entre as precipitações dos pluviômetros utilizados, todos localizados na Universidade Federal de Santa Maria. Para verificação dos

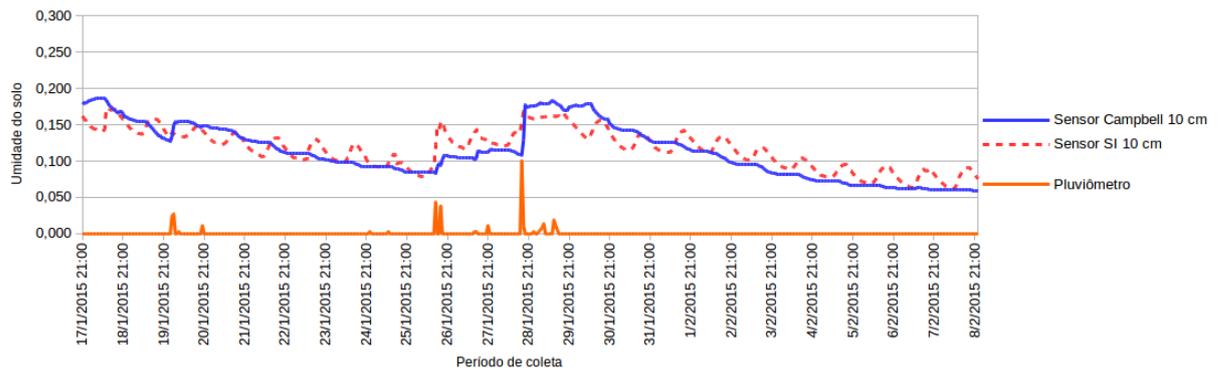


Figura 4.12: Sensores de umidade instalados na camada de 10 centímetros.

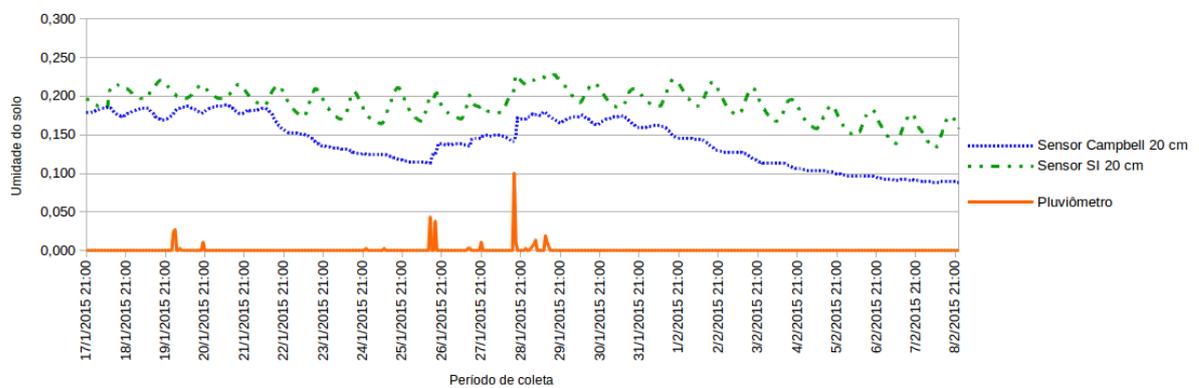


Figura 4.13: Sensores de umidade instalados na camada de 20 centímetros.

valores do pluviômetro, foram utilizados os valores de um pluviômetro do Sistema Irriga e de um pluviômetro da estação INMET. No gráfico da figura 4.15, pode-se verificar que os valores obtidos pelo pluviômetro do presente trabalho ficaram próximos aos coletados pelos outros dois pluviômetros utilizados para a comparação.

Entretanto, mesmo que o pluviômetro com custo reduzido utilizado funcione de forma aceitável, sendo capaz de demonstrar razoavelmente a quantidade de chuva ocorrida, pode-se notar que os valores coletados divergiram no dia 28, que foi o maior volume de chuva registrado durante o período da coleta de dados. Isso ocorre provavelmente por que a área de coleta do pluviômetro utilizado é de 55 cm^2 , sendo relativamente pequena em comparação, por exemplo, com o tamanho da área de coleta do pluviômetro do Sistema irriga que é de 186 cm^2 (TE525, 2015), principalmente para efetuar a coleta de chuvas muito intensas em curtos intervalos de tempo.

Na figura 4.16 são apresentados os valores médios referentes à temperatura do solo coletados pelo medidor de umidade e pluviometria, e a temperatura ambiente da cidade de

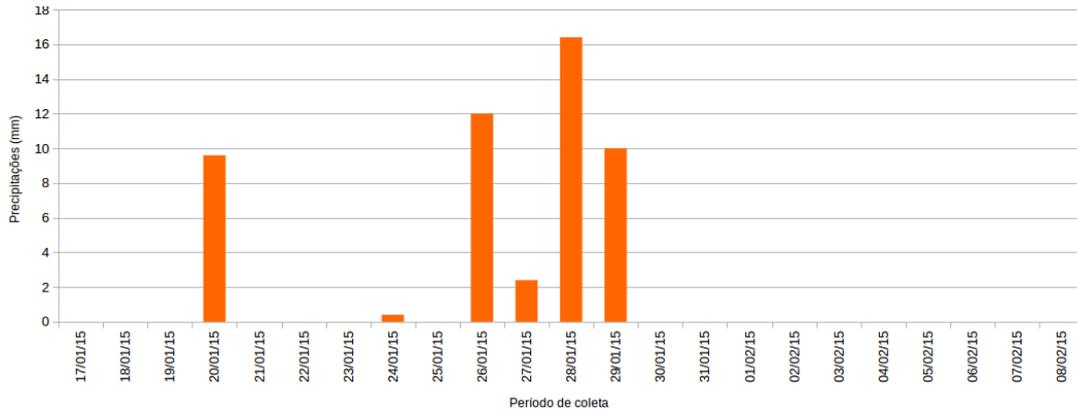


Figura 4.14: Precipitações por dia.

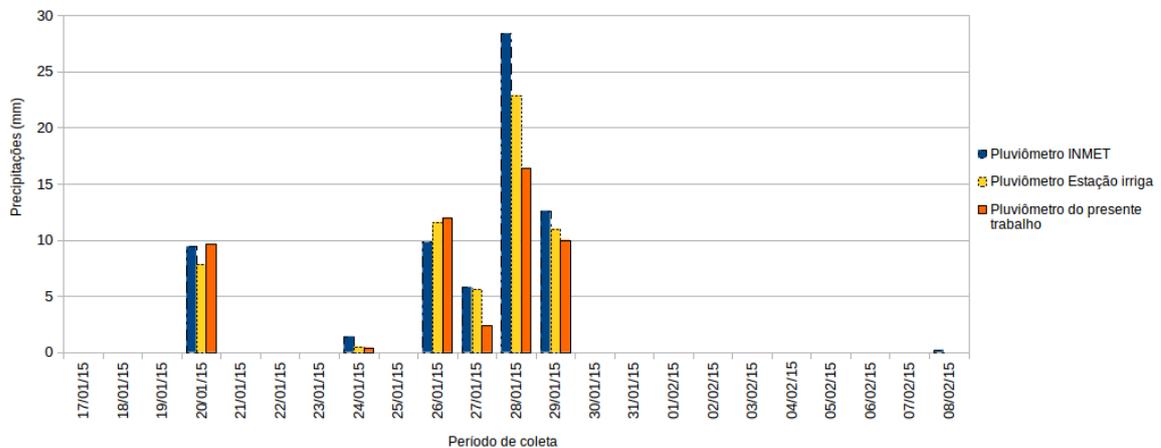


Figura 4.15: Comparação entre os pluviômetros.

Santa Maria coletados pela estação INMET. Pode-se verificar que os valores de temperatura do solo variam mais em comparação aos dados referentes a temperatura ambiente. Estes dados de temperatura do solo podem ser utilizados para efetuar o plantio em um período adequado, por exemplo, entre uma faixa específica de temperatura do solo para uma determinada cultura.

Com a análise dos dados coletados, percebeu-se que a correlação entre os valores dos sensores de umidade poderia ser melhorada. Por isso, foram estudadas duas alternativas apresentadas na seção 4.3.2: a) o ajuste dos valores da umidade do solo com a utilização de fórmulas que consideram a temperatura do solo; b) e a utilização de um programa de mineração de dados com algumas funções disponíveis (regressão linear e redes neurais). Com isso, buscou-se a geração de equações de ajuste dos valores de umidade dos sensores SI, utilizando como referência os sensores CS616 da Campbell, que são sensores amplamente utilizados no campo. As análises estatísticas são realizadas para determinar os coeficientes de determinação (correlação

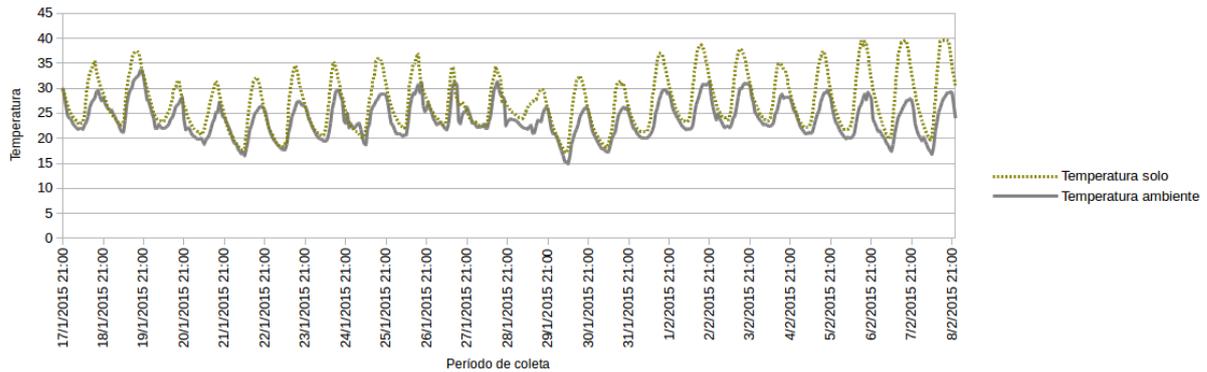


Figura 4.16: Comparação entre as temperaturas coletadas.

ao quadrado) dos sensores utilizando todas as informações disponíveis: dados dos sensores de umidade, temperatura do solo e ambiente (INMET) e pluviometria.

4.3.2 Fórmulas de ajuste da umidade do solo

Nesta seção são apresentadas as duas alternativas utilizadas para efetuar o ajuste da umidade do solo. A primeira leva em consideração a temperatura do solo para realizar estes ajustes e a segunda é através da utilização de um software de mineração de dados.

Na primeira alternativa, verificou-se que muitos fatores podem influenciar a condutividade elétrica do solo, como a textura do solo, quantidade de água, entre outros (BREVIK; FENTON; HORTON, 2004). E a temperatura é um fator importante que também afeta esta condutividade elétrica (conforme a condutividade aumenta, a resistência do material diminui conforme explicado na seção 2.3), e que eleva a mesma à cerca de 1,91% por °C na temperatura (CORWIN; LESCH, 2005). Por isso, para efetuar a correção do efeito da temperatura nas medições da umidade do solo, as medições são realizadas na temperatura atual do solo, e são ajustadas para uma temperatura particular. Para isso, foi utilizado o modelo de relação ilustrado na fórmula 4.2 que foi indicado no estudo sobre modelos de correção no trabalho de (MA et al., 2011), por permitir um ajuste com precisão satisfatória entre a faixa de 3-50 °C para medidas referentes a uma temperatura padrão de 25 °C.

$$U_{25} = \frac{U_t}{1 + \delta(T - 25)} \quad (4.2)$$

Onde U_{25} é a umidade do solo na temperatura base utilizada de 25 °C, δ é a compensação da temperatura informada no modelo (1,91), U_t é a umidade coletada na temperatura atual

e T é a temperatura do solo coletada.

A segunda alternativa encontrada foi através da utilização do software Weka (Waikato Environment for Knowledge Analysis) que possui um conjunto de algoritmos de aprendizado de máquina para tarefas de mineração de dados (WEKA, 2015), e foi utilizado para realizar a aferição dos dados dos sensores de umidade do solo SI considerando como referência os sensores de umidade do solo da Campbell. Os testes foram realizados utilizando a técnica de validação cruzada (*k-fold cross-validation*) disponível no software. Foram também efetuados testes utilizando fase de treinamento, e os resultados foram semelhantes ao da validação cruzada.

Nas seções 4.3.3 e 4.3.4 serão apresentados os coeficientes de determinação das comparações realizadas entre os sensores de umidade do solo SI e CS616 da Campbell.

4.3.3 Coeficientes de determinação do sensor da camada de 10 centímetros

Nesta seção, os coeficientes de determinação do sensor de umidade do solo da camada de 10 centímetros apresentados no quadro 4.2 são analisados. Neste quadro são realizadas três comparações entre o sensor SI e o sensor CS616 da Campbell:

- *Sem outras Inf. Cont*: Utilizando apenas o sensor da camada de 10 cm sem outras informações de contexto;
- *Com Inf. Contexto e S. Umidade 10 cm*: Utilizando os dados do sensor da camada de 10 cm com outras informações de contexto (temperatura do solo e do ambiente e pluviometria);
- *Com Inf. Contexto e S. Umidade 10/20 cm*: Utilizando os dados coletados dos sensores de umidade SI das duas camadas (10 e 20 cm) com outras informações de contexto (temperatura do solo e do ambiente e pluviometria).

A *Regressão Linear* e as *Redes Neurais Multilayer Perceptron* presentes no programa Weka, foram as funções utilizadas para a comparação dos *dados originais* e dos dados com *ajustes de temperatura* apresentados nas seções 4.3.1 e 4.3.2, pois estão entre as funções utilizadas para este fim (VILLORDON et al., 2010; GONZALEZ-SANCHEZ; FRAUSTO-SOLIS; OJEDA-BUSTAMANTE, 2014). Já as figuras 4.17 e 4.18, ilustram mais claramente os coeficientes de determinação referentes aos *dados originais - DO* e aos dados dos *ajustes de temperatura - AT* do quadro 4.2. Nos testes realizados, o sensor CS616 da Campbell da camada correspondente (10 cm) foi utilizado como referência para realizar todas as comparações.

Camada de 10 cm	Sem outras Inf. Cont.	Com Inf. Contexto e S. Umidade 10 cm	Com Inf. Contexto e S. Umidade 10/20 cm
Regressão Linear Dados originais	R ² : 0.78	R ² :0.93	R ² : 0.93
Regressão Linear Ajuste temperatura	R ² : 0.90	R ² : 0.93	R ² : 0.93
Rede Neural Dados originais	R ² : 0.72	R ² : 0.94	R ² : 0.94
Rede Neural Ajuste temperatura	R ² : 0.91	R ² : 0.95	R ² : 0.94

Quadro 4.2: Coeficientes de determinação dos sensores de umidade do solo da camada de 10 centímetros.

Tanto os valores gerados através do ajuste de temperatura, quanto os valores gerados com as fórmulas de correção indicadas pelo software de mineração de dados, tiveram resultados satisfatórios. Entretanto, os resultados com o ajuste da umidade levando em consideração a temperatura do solo ficaram com resultados mais satisfatórios comparados com as fórmulas que utilizaram os dados originais pelo software Weka. Isso pode ser observado na comparação na coluna *Sem outras Inf. Cont.*, onde tanto para a função de regressão linear quanto para a rede neural, o aumento do coeficiente foi significativo com o ajuste da temperatura do solo:

- *Regressão linear - DO*: De R²: 0.78 para R²: 0.90;
- *Rede neural - DO*: De R²: 0.72 para R²: 0.91;

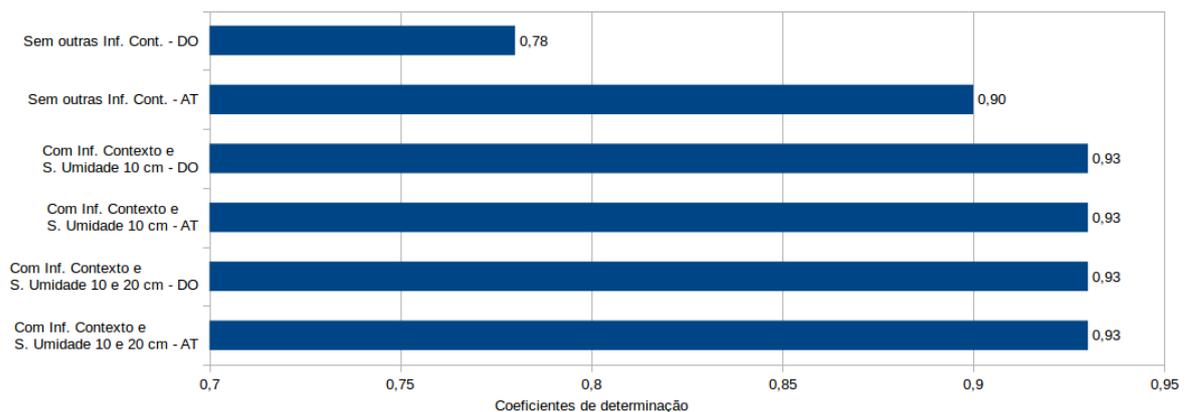


Figura 4.17: Coeficientes de determinação dos sensores de umidade com regressão linear da camada de 10 centímetros.

As figuras 4.19 e 4.20 ilustram a comparação entre os dados do sensor CS616 e do sensor SI instalados a 10 centímetros na caixa de areia. As fórmulas 4.3 e 4.4 que efetuam os ajustes realizados nos valores da umidade do solo, foram geradas através da função de regressão

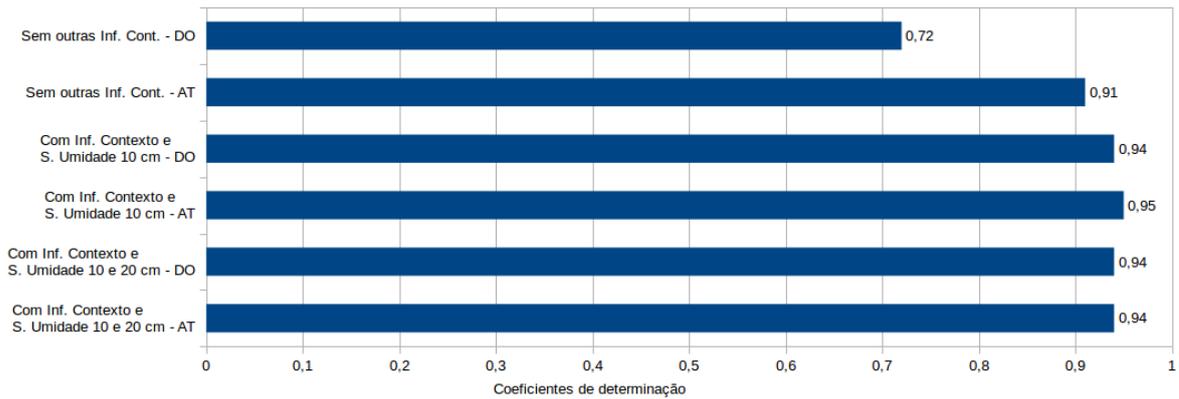


Figura 4.18: Coeficientes de determinação dos sensores de umidade com a rede Neural Multilayer Perceptron da camada de 10 centímetros.

linear pelo software Weka. Pode-se perceber que estas fórmulas podem ser implementadas diretamente na camada de Normalização de dados da arquitetura (Arduino), pois não requerem grande processamento de dados.

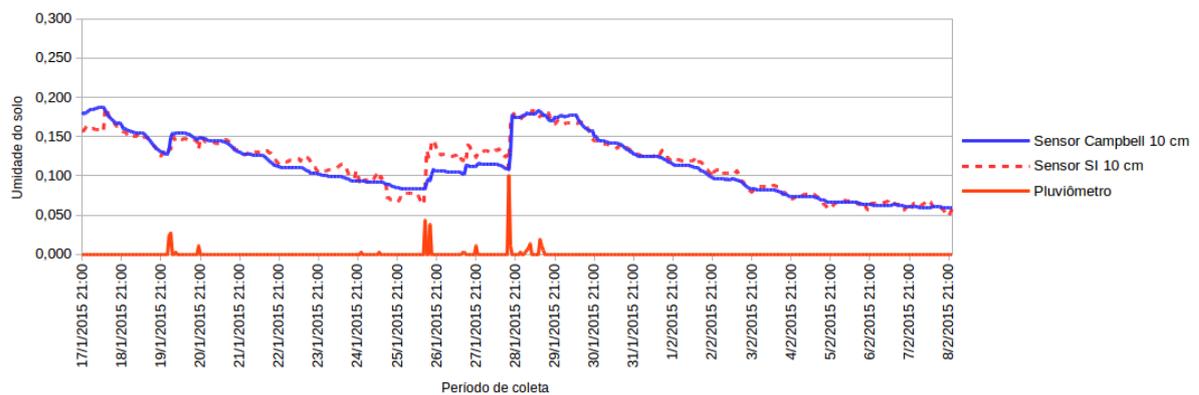


Figura 4.19: Ajuste da umidade do solo para o sensor SI da camada de 10 centímetros com regressão linear.

Fórmula para ajuste da umidade do solo do sensor SI à 10 centímetros utilizando dados da camada de 10 centímetros (regressão linear):

$$US10 = (-0.0013 * SI10) + (-0.0009 * TS) + (-0.0027 * TA) + (-0.003 * P) + 1.3798 \quad (4.3)$$

Onde US10 é o valor final da umidade do solo para a camada de 10 centímetros, SI10 é o sensor resistivo à 10 cm, SI20 é o sensor resistivo à 20 cm, TS é a temperatura do solo, TA é a temperatura ambiental da estação INMET e P são as precipitações.

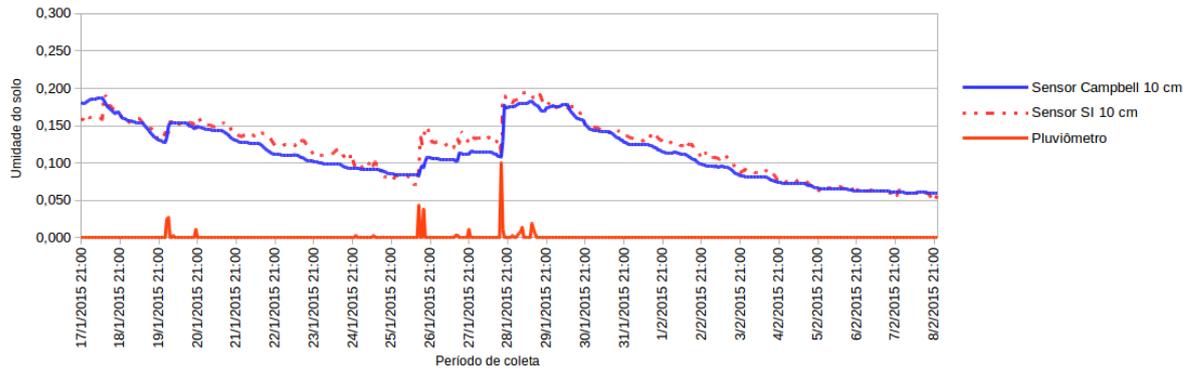


Figura 4.20: Ajuste da umidade do solo para o sensor SI da camada de 10 centímetros utilizando informações das camadas de 10 e 20 centímetros com regressão linear.

Formula para ajuste da umidade do solo do sensor SI à 10 centímetros utilizando dados das camadas de 10 e 20 centímetros (regressão linear):

$$US10 = (-0.0003 * SI20) + (-0.0012 * SI10) + (-0.0009 * TS) + (-0.003 * TA) + (-0.0029 * P) + 1.55 \quad (4.4)$$

Com a utilização de todas as informações de contexto disponíveis (temperatura do solo e do ambiente e pluviometria) e mais os sensores de umidade do solo SI (10 cm ou 10 e 20 cm), os resultados variaram entre R^2 : 0.93 e R^2 : 0.95. Com isso, pode-se notar que para a camada de 10 centímetros, o ajuste da temperatura é muito importante para elevar os coeficientes de determinação dos sensores de umidade do solo, conforme coluna *Sem outras Inf. Cont* da tabela 4.2. As informações de contexto como temperatura ambiental e pluviometria auxiliam no complemento deste aumento, conforme as colunas *Com Inf. Contexto e S.Umidade 10 cm* e *Com Inf. Contexto e S. Umidade 10/20cm*.

4.3.4 Coeficientes de determinação do sensor da camada de 20 centímetros

Nesta seção, os coeficientes de determinação do sensor de umidade do solo da camada de 20 centímetros, apresentados no quadro 4.3, são analisados. Neste quadro são realizadas três comparações entre o sensor SI e o sensor CS616 da Campbell, já apresentados na seção 4.3.3. Da mesma forma, as funções utilizadas para efetuar as comparação foram a regressão linear e as redes neurais multilayer perceptron. As figuras 4.21 e 4.22, apresentam os coeficientes de determinação referentes aos *dados originais - DO* e aos dados dos *ajustes de temperatura - AT* do quadro 4.3, utilizando o sensor CS616 da Campbell da camada correspondente (20 cm)

como referência.

Camada de 20 cm	Sem outras Inf. Cont.	Com Inf. Contexto e S. Umidade 20 cm	Com Inf. Contexto e S. Umidade 10/20 cm
Regressão Linear Dados originais	R ² : 0.51	R ² : 0.78	R ² : 0.88
Regressão Linear Ajuste temperatura	R ² : 0.75	R ² : 0.78	R ² : 0.88
Rede Neural Dados originais	R ² : 0.42	R ² : 0.78	R ² : 0.94
Rede Neural Ajuste temperatura	R ² : 0.74	R ² : 0.75	R ² : 0.94

Quadro 4.3: Coeficientes de determinação dos sensores de umidade do solo da camada de 20 centímetros.

Na camada de 20 centímetros, os dados originais do sensor SI coletados não foram muito satisfatórios em relação a comparação com o sensor da Campbell, com coeficientes de determinação de R²:0.51 para regressão linear e R²:0.42 para a rede neural. Entretanto, os ganhos obtidos com o ajuste da temperatura foram ainda maiores, se comparados com a camada de 10 centímetros. Na função de regressão linear, o coeficiente de determinação com o ajuste da temperatura do solo aumentou:

- *Regressão linear - DO*: De R²:0.51 para R²: R²:0.75;
- *Rede neural - DO*: De R²: 0.42 para R²: 0.74;

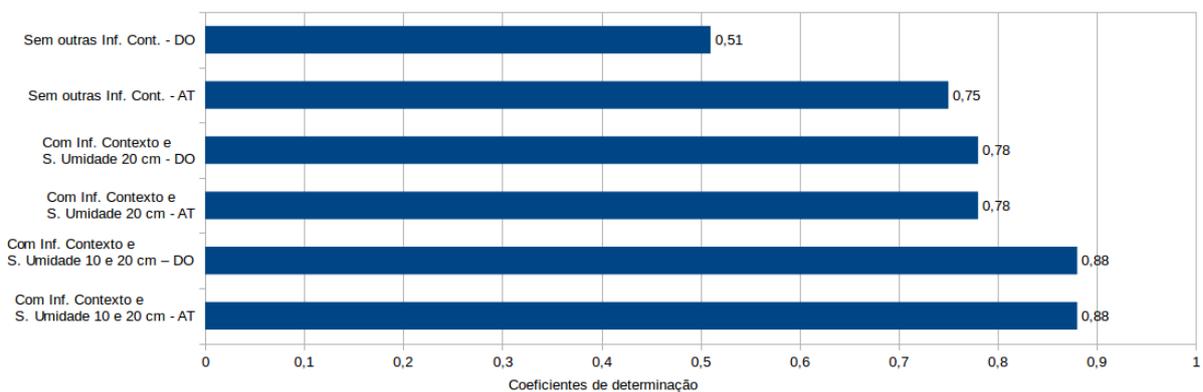


Figura 4.21: Coeficientes de determinação dos sensores de umidade com regressão linear da camada de 20 centímetros.

Na comparação *Com Inf. Contexto e S. Umidade 20 cm* da tabela, que utiliza a temperatura do solo e também as informações de temperatura ambiental e pluviometria, nota-se que

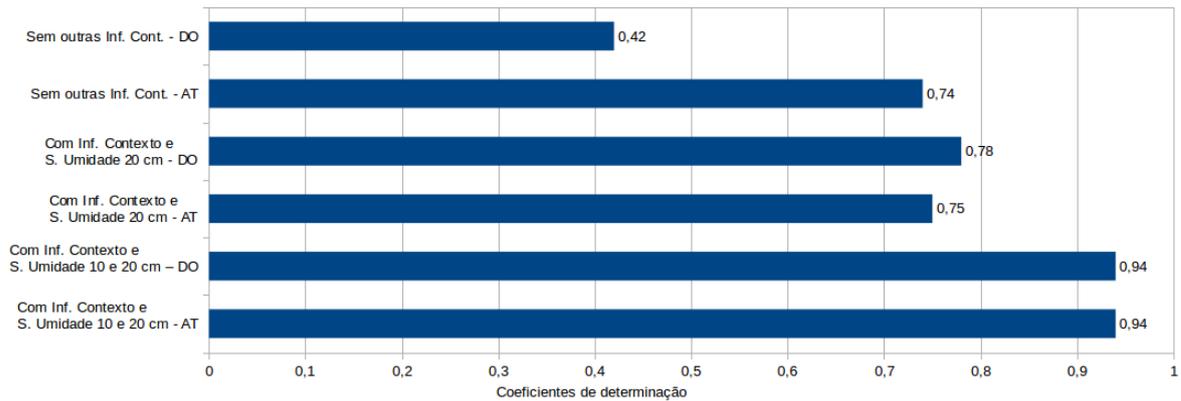


Figura 4.22: Coeficientes de determinação dos sensores de umidade com a rede Neural Multilayer Perceptron da camada de 20 centímetros.

o ganho foi pequeno em comparação com os valores que utilizaram apenas o ajuste da temperatura do solo, ficando entre R^2 : 75 e R^2 : 78. Porém, na comparação *Com Inf. Contexto e S. Umidade 10/20 cm*, percebe-se como é importante para a camada de 20 centímetros, utilizar todas as informações de contexto disponíveis, não sendo suficiente o ajuste com a temperatura do solo, para que os resultados fossem satisfatórios da mesma forma que foram na camada de 10 centímetros. Nesta comparação, os resultados ficaram entre R^2 : 88 e R^2 : 94.

As figuras 4.23 e 4.24 ilustram a comparação dos valores dos sensores SI e CS616 instalados à 20 centímetros na caixa de areia. As fórmulas 4.5 e 4.6 que efetuam os ajustes realizados nos valores da umidade do solo, foram geradas através da função de regressão linear pelo software Weka.

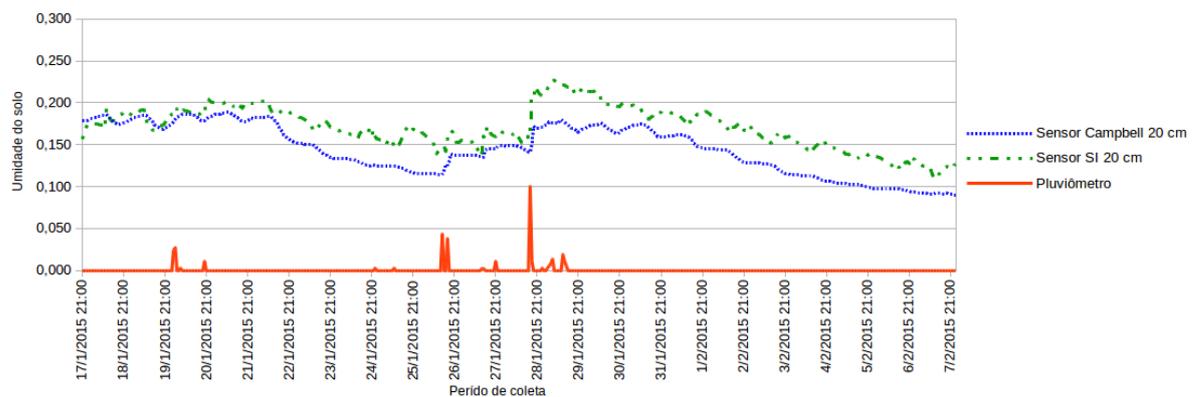


Figura 4.23: Ajuste da umidade do solo para o sensor SI da camada de 20 centímetros com regressão linear.

Fórmula para ajuste da umidade do solo para o sensor SI à 20 centímetros utilizando os

dados dos sensores da camada de 20 cm (regressão linear)::

$$US20 = (-0.0014 * SI20) + (-0.0015 * TS) + (-0.0029 * TA) + (-0.0013 * P) + 1.4393 \quad (4.5)$$

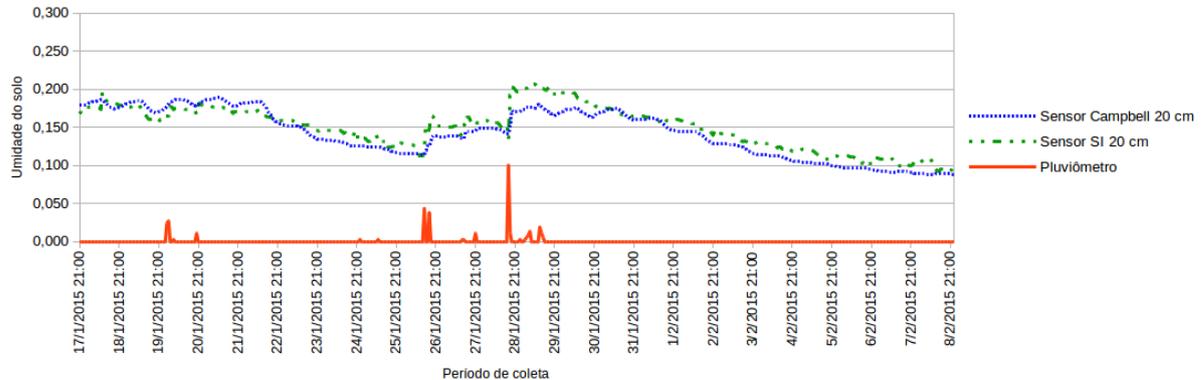


Figura 4.24: Ajuste da umidade do solo para o sensor SI da camada de 20 centímetros utilizando informações das camadas de 10 e 20 centímetros com regressão linear.

Fórmula para ajuste da umidade do solo para o sensor SI à 20 centímetros utilizando dados dos sensores das camadas de 10 e 20 cm (regressão linear):

$$US20 = (-0.0005 * SI20) + (-0.0007 * SI10) + (-0.0013 * TS) + (-0.0024 * TA) + (-0.0024 * P) + 1.290 \quad (4.6)$$

Com esta análise, pode-se perceber que para esta camada de 20 centímetros, os dados referentes aos sensor SI da camada de 10 centímetros são importantes para possibilitar uma maior proximidade nos valores do sensores CS616 da Campbell. É interessante também, para trabalhos futuros, realizar testes com mais camadas de sensores dispostos no solo, para verificar se esta dependência de mais informações de contexto (ou seja, dos sensores das camadas superiores) aumenta conforme a profundidade da camada.

4.4 Estudo de Caso 2

Este estudo de caso foi realizado com o objetivo de testar alternativas para o sistema de fornecimento de energia através de uma bateria e de um painel solar para os protótipos do medidor de umidade e pluviometria. Uma vez que foi verificado o correto funcionamento da leitura dos dados e o envio das informações do medidor com a placa GSM/GPRS para o servidor, bem como o funcionamento dos sensores utilizados, é importante efetuar o estudo de

possibilidades para a autossuficiência dos medidores, pois a necessidade de cabos para ligá-los na rede de energia elétrica em áreas irrigadas normalmente não é uma opção viável. Estas alternativas de fornecimento de energia não foram utilizadas no estudo de caso 4.3 para que não fossem considerados problemas relacionados a quantidade de energia despendida e a capacidade das baterias.

As medições referentes à corrente despendida do medidor de umidade e pluviometria são apresentadas na tabela 4.2, e foram realizadas com um multímetro que foi inserido em um circuito de carga para ligar o medidor. Para efetuar a comparação, foram realizadas algumas medições com a utilização do programa *Blink* que está disponível na plataforma de programação do Arduino, e que apenas liga e desliga um led do Arduino, consumindo cerca de 70 mA. As outras comparações são referentes a três estágios do programa que é responsável pela coleta dos dados e envio das informações. A coluna *Tempo de espera* representa o medidor no momento em que o programa está em estado de espera, aguardando o momento para realizar a coleta de dados e o envio das informações, consumindo cerca de 69 mA, e a terceira coluna *Coleta de dados* com 71 mA de corrente despendida no momento em que o medidor realiza a coleta de dados. A quarta coluna de *Envio das informações* representa o momento do envio das mesmas para o servidor, que leva de 1 a 3 minutos, dependendo da conexão com a rede GSM e com o servidor, variando entre 100 mA e 160 mA, porém com picos de 260 mA de corrente.

Tabela 4.2: Comparativo da corrente despendida do medidor de umidade e pluviometria.

	Blink	Tempo de espera	Coleta de dados	Envio das informações
Consumo de energia	70 mA	69 mA	71 mA	260 mA

Com isso, nota-se que o maior consumo é referente à placa de transmissão GSM/GPRS, e por isso uma das opções para minimizar o consumo do medidor de umidade e pluviometria é diminuir a quantidade de informações enviadas. Outra alternativa é a realização do envio das informações apenas durante o dia, evitando gastos excessivos da carga da bateria durante a noite. Neste sentido, os dados coletados a noite podem ser enviados, por exemplo, no início da manhã quando a bateria já está sendo carregada com a incidência do sol. Outra alternativa seria efetuar a coleta dos dados, por exemplo à cada hora, e o envio das informações em um período mais estendido, por exemplo à cada 4 horas, encontrando uma maneira de efetuar o envio de todas estas informações apenas com uma conexão na rede GSM e com uma conexão

no servidor para diminuir o consumo de energia. Porém, conforme relatado na seção 4.5, até a versão atual do protótipo é necessário efetuar uma nova conexão na rede GSM a cada envio de informações.

A disponibilização de energia para o Arduino pode ser realizada através de uma conexão USB (5 V) ou com uma fonte de alimentação externa (6-20 V) (ARDUINO, 2014). Estas duas alternativas serão apresentadas nas seções 4.4.1 e 4.4.2.

4.4.1 Painel solar e bateria USB

A figura 4.25 apresenta o painel solar de 3 Watts e a bateria de 5V x 6000 mAh para ligar o medidor através da conexão USB. Porém, pelas pesquisas realizadas, não foi encontrada nenhuma forma de monitorar a carga da bateria através da conexão USB.



(a) Painel solar (PORTAPOW, 2014).



(b) Bateria USB (PORTAPOW, 2014).

Figura 4.25: Painel solar e bateria USB

Nos testes realizados em um período de 3 dias com forte incidência de sol, a bateria foi carregada normalmente com o medidor, enviando informações apenas de 4 em 4 horas. Porém, com subsequentes dias de pouca incidência de sol ou com chuva, o painel solar não teve capacidade suficiente para carregar a bateria. Nestas condições, o envio das informações só acontece quando a bateria possui energia suficiente, dependendo de uma nova recarga através da incidência do sol.

4.4.2 Painel solar e bateria para alimentação externa de energia

Nesta alternativa foram efetuados testes para verificar a alimentação externa de energia para o Arduino, bem como realizar a leitura da carga desta bateria. A tensão de entrada para a placa Arduino, quando a mesmo está utilizando uma fonte de alimentação externa, é efetuada

pelo pino *VIN*, quando não é utilizada a alimentação de 5 V a partir da conexão USB ou de outra fonte de alimentação regulada (ARDUINO, 2014). Neste estudo de caso, apenas o protótipo principal que utiliza a placa GSM/GPRS foi testado com o sistema de fornecimento de energia por bateria.

O painel solar utilizado para carregar a bateria foi o modelo MSX10 da Campbell (CAMPBELL-SCIENTIFIC, 2014) apresentado na figura 4.26, e o regulador de tensão utilizado foi o modelo PS12 da Campbell, que é responsável por controlar a carga que é repassada do painel solar para carregar a bateria, bloqueando qualquer corrente reversa da bateria para o painel solar. O regulador de tensão e a bateria utilizados são apresentados na figura 4.27.



Figura 4.26: Painel solar para alimentação externa de energia.

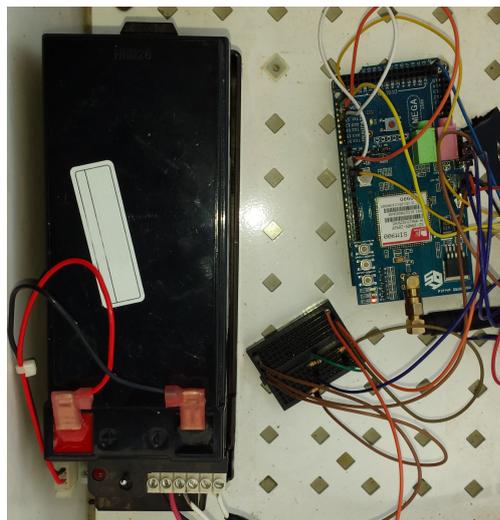


Figura 4.27: Regulador de tensão e bateria para alimentação externa de energia.

O Arduino pode operar com uma fonte externa de 6 a 20 V através do pino *VIN*, porém é aconselhável evitar valores de tensão que fiquem próximos a estes limites (ARDUINO, 2014). Por isso, conforme testes realizados, quando a bateria encontra-se abaixo de um nível de 7.5

V, o protótipo não possui energia suficiente para o envio das informações, pois o valor já se aproxima dos limites estabelecidos pelo fabricante.

Já o circuito com divisores de tensão apresentado na figura 4.28, é utilizado para efetuar medições referentes a voltagens maiores que 5 V nas saídas analógicas do Arduino (voltagens aceitas entre 0-5 V). Para efetuar o cálculo do fator de resistência que é utilizado para realizar a queda de tensão, foram utilizados os resistores $R1 = 2k\Omega$ e $R2 = 1k\Omega$ para uma voltagem máxima de 15 V, onde o fator de resistência é calculado através da equação 4.7 conforme (MARGOLIS, 2011):

$$Fator = R2 / (R2 + R1) \quad (4.7)$$

Com isso, têm-se a voltagem de saída do circuito para o Arduino de no máximo 5 V, evitando danos nos pinos analógicos do mesmo conforme indicação no site do fabricante, pois as entradas analógicas realizam apenas o mapeamento das tensões de entrada entre 0 e 5 V em valores inteiros entre 0 e 1023. (ARDUINO, 2014). Posteriormente, para se ter o valor real da voltagem da bateria, utiliza-se a equação 4.8:

$$Voltagem = (v/f) * vr \quad (4.8)$$

Onde: v é o valor coletado na entrada analógica do Arduino (valores entre 0 e 1023), divide-se este valor pelo fator de resistência (f) calculado na equação 4.7, e multiplica-se o resultado pela voltagem de referência (vr) que neste caso é 5 V, para obtenção da carga real da bateria entre 0 e 15 V.

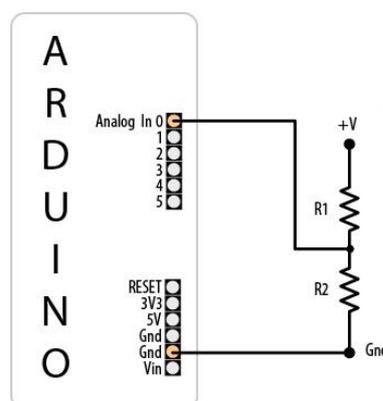


Figura 4.28: Divisor de tensão utilizado para leitura da carga da bateria. Adaptado de (MARGOLIS, 2011).

A figura 4.29 apresenta o nível de carga da bateria (mV) durante alguns dias de coleta de

dados. Com o monitoramento do comportamento da bateria dependendo do ambiente em que o medidor está instalado (por exemplo, verão no Brasil, ou inverno na Suíça), pode-se estabelecer valores limites de carga para o envio de informações, visando a diminuição do consumo quando a carga está em níveis críticos. Também é importante evitar o descarregamento completo da bateria para aumentar a sua vida útil.

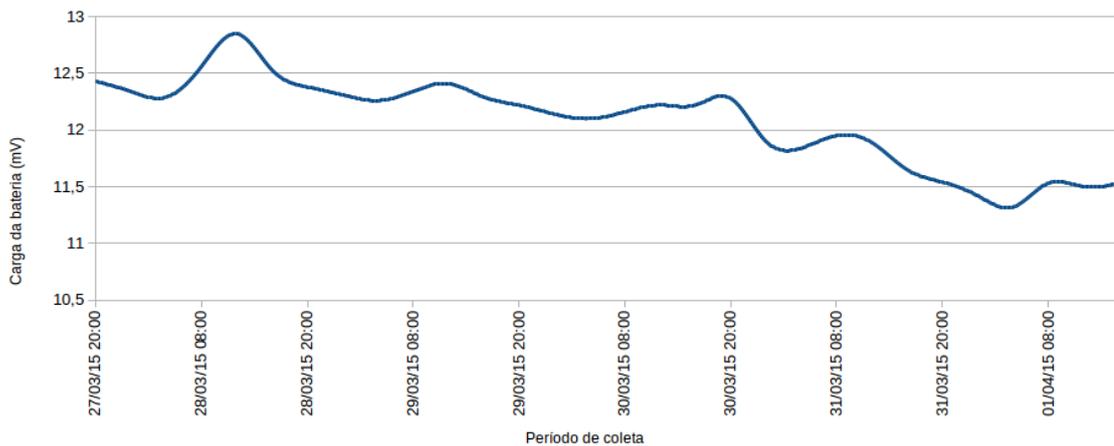


Figura 4.29: Monitoramento da carga da bateria do medidor.

Os testes realizados demonstraram que as duas alternativas para o sistema de fornecimento de energia são viáveis. Entretanto, pela pesquisa realizada, para a alternativa que utiliza a entrada USB, não foi encontrada uma alternativa para realizar o monitoramento da carga da bateria, sendo também necessária a diminuição do consumo de energia do medidor ou de um painel solar de maior potência para o seu correto funcionamento. Já a alternativa que utiliza uma fonte de alimentação externa apresenta uma boa opção para o monitoramento da bateria através do divisor de tensão apresentado, embora ainda não represente uma solução completa para que o medidor de umidade e pluviometria seja autossuficiente. Para isso, é necessário a utilização de um regulador de tensão que realize o bloqueio de qualquer corrente reversa da bateria para o painel solar, e que realize o controle da carga repassada do painel solar para a bateria para efetuar a carga completa da bateria, aumentando a sua durabilidade e eficiência.

4.5 Problemas encontrados

Durante o desenvolvimento dos protótipos, e da realização da coleta dos dados e envio das informações dos medidores de umidade e pluviometria, foram encontrados alguns problemas que serão brevemente apresentados.

Os testes de coleta de dados e envio das informações dos sensores foram realizados com o medidor de umidade e pluviometria ligado na rede de energia elétrica, para desconsiderar problemas com gasto de energia e potência das baterias utilizadas, bem como para evitar problemas de não funcionamento do medidor com a ocorrência de vários dias sem incidência de sol.

Foi necessário também, que o IP utilizado a cada envio de informações pela placa GSM/GPRS fosse alterado, pois depois de realizar o envio de diversos conjuntos de informações com o mesmo IP, o medidor não consegue mais efetuar o envio de informações. Além disso, a conexão com o servidor é realizada novamente a cada envio para evitar problemas de comunicação entre a troca de mensagens HTTP que ocorreram em testes realizados. Também não foi encontrada uma maneira de efetuar a transmissão das informações em horários específicos do dia através de comandos AT. Nesta fase de desenvolvimento, o protótipo espera um tempo determinado para a transmissão do próximo conjunto das informações.

A proposta de resolução das limitações do trabalho é apresentada na seção 6.1 dos trabalhos futuros.

5 TRABALHOS RELACIONADOS

No trabalho de Kjaer et al. (2008), foi desenvolvido um *middleware* com suporte a coleta e compartilhamento de contextos a partir de fontes e ambientes heterogêneos encontrados em uma fazenda, onde os usuários finais auxiliaram nas etapas de levantamento de requisitos e de projeto do *middleware* proposto. O objetivo principal do *middleware* é realizar um filtro nas informações de contexto disponíveis, e apresentar as informações relevantes dependendo da tarefa que o agricultor está realizando em uma determinada localização da propriedade. O *middleware* foi desenvolvido para ter suporte a dois tipos de cenários distintos: i) um envolvendo atividades ou tarefas internas (dentro de prédios, por exemplo), possibilitando que os agricultores insiram informações diretamente em um sistema através de dispositivos móveis; ii) e outro voltado para atividades realizadas no campo (sem localização fixa para a realização da tarefa, por exemplo), podendo envolver um veículo utilizado para alguma tarefa específica.

A aplicação desenvolvida para o primeiro cenário chama-se Kilo.One, apresentada na figura 5.1. Na aplicação do protótipo desenvolvido, diversas tarefas sendo realizadas são reconhecidas através de áreas geográficas da propriedade, porém ainda não levando em consideração os diversos contextos que podem caracterizar uma determinada tarefa (por exemplo, tipos de equipamentos utilizados). Isto poderia auxiliar no reconhecimento, por exemplo, da execução de tarefas diferentes na mesma localização geográfica. A aplicação possui também um serviço de registros, que permite que o agricultor insira comentários relacionados a tarefa realizada. Além disso, um serviço de alarmes permite o acesso a sistemas de monitoramento já existentes na fazenda, que disponibilizam alertas/alarmes para o agricultor. O protótipo para cenários externos não foi implementado neste trabalho.

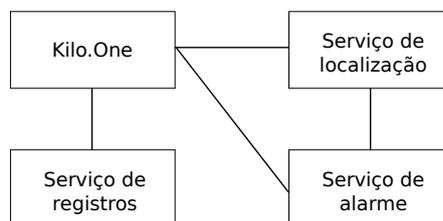


Figura 5.1: Visão conceitual do protótipo da aplicação Kilo.One. Adaptado de (KJAER, 2008).

No trabalho de Mohammadrezaei et al. (2011), é apresentada uma arquitetura sensível ao contexto para a agricultura que utiliza diversos sensores para a coleta de dados referentes ao solo, as plantas e ao clima. A arquitetura do trabalho possui quatro camadas conforme figura

5.2, com destaque para a camada superior, chamada de camada de atuação, que utiliza as informações de contexto providas das camadas inferiores para disponibilizar serviços, como por exemplo, sistemas de pulverização ou de inserção de fertilizantes químicos no solo. Nesta arquitetura, primeiramente são coletados os dados das condições do solo, clima e planta, para posteriormente efetuar a criação e classificação das informações de contexto. Na próxima camada, as taxas de materiais injetáveis no solo são calculadas para os testes realizados em laboratório, para que posteriormente os atuadores realizem as ações necessárias.

Os dados coletados em laboratório foram utilizados para efetuar uma simulação de crescimento da cultura do milho, onde através dos dados coletados do solo, são realizados os ajustes referentes ao tipo e a quantidade de fertilizante utilizado, buscando as condições ideais de solo para o crescimento da planta.

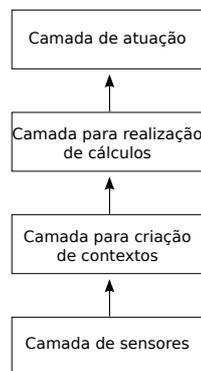


Figura 5.2: Arquitetura sensível ao contexto para a agricultura. Adaptado de (MOHAMMADREZAEI; FATHI; ATTARZADEH, 2011)

No trabalho de Prasad et al. (2012), foi desenvolvido um sistema para efetuar o controle da irrigação, que tem como principal objetivo a redução do consumo de água na agricultura. Quando o sistema implementado detecta que os níveis de água ou temperatura do solo monitorados estão fora do padrão estabelecido, um motor é automaticamente ligado para efetuar a irrigação, até que os níveis da temperatura e da umidade do solo estejam dentro dos valores determinados, desligando o motor responsável pela irrigação. Os dados são coletados através de uma estação base que utiliza um microcontrolador ARM 7/LPC212, e são enviados para o servidor através da rede GSM.

O trabalho demonstra o desenvolvimento de controladores que podem ser utilizados para realizar a irrigação por microclima, visando oferecer uma solução simples e de baixo custo. Para testar o protótipo implementado, foram realizados testes em laboratório verificando as condições dos sensores de temperatura e umidade, e ativando o motor quando notadas as condições

estabelecidas pelo sistema. A figura 5.3 ilustra a interconexão entre os componentes que foram utilizados no trabalho.

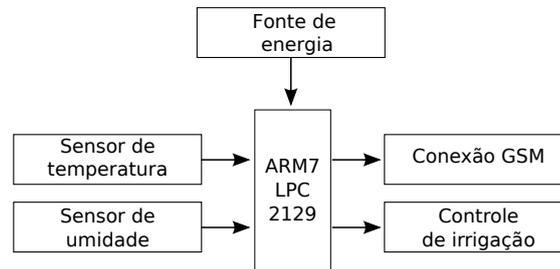


Figura 5.3: Interconexão entre os componentes utilizados no trabalho. Adaptado de (PRASAD et al., 2012).

Já no trabalho de Mills et al. (2013), foi implementado um sistema de controle e monitoramento da irrigação baseado na rede GSM utilizando serviços de SMS. Este sistema possibilita aos usuários a utilização de dispositivos móveis e comandos SMS para gerenciar o sistema de irrigação, sendo possível verificar os níveis de umidade do solo, efetuar a irrigação por um determinado período de tempo, entre outros. Na figura 5.4 apresentada, o microcontrolador é responsável por receber as instruções do sistema, decodificar estas instruções, e enviar os comandos correspondentes a cada interface de operação dos componentes conectados a ele. Com isso, existem duas formas de comunicação: i) comunicação interna, onde o microcontrolador se comunica com as interfaces dos dispositivos do sistema, independentemente das requisições externas do usuário; ii) e comunicação externa, que é uma resposta a uma requisição realizada pelo usuário através de comandos SMS.

Os resultados demonstraram que é possível para os usuários a utilização do SMS para monitorar diretamente as condições de sua propriedade, e controlar a irrigação de acordo com as necessidades de água das culturas. O protótipo apresentado utiliza algumas tecnologias que também foram utilizadas no presente trabalho, como a plataforma Arduino e a rede GSM. Entretanto, os autores ainda estudam a possibilidade de adicionar uma alternativa de fornecimento de energia sustentável para o sistema, que utilize um painel solar para carregar uma bateria. Com isso, seria possível manter o protótipo em pleno funcionamento mesmo em situações onde ocorre a queda no fornecimento de energia ou ainda para áreas sem um fornecimento regular de energia.

Na pesquisa realizada, não foram encontrados trabalhos relacionados a irrigação de precisão com uma arquitetura sensível ao contexto que possui uma abordagem formal para a representação das informações de contexto bem definida. Além disso, nem todos os trabalhos possuem um protótipo implementado que utiliza-se de tecnologias *open hardware*, e que dis-

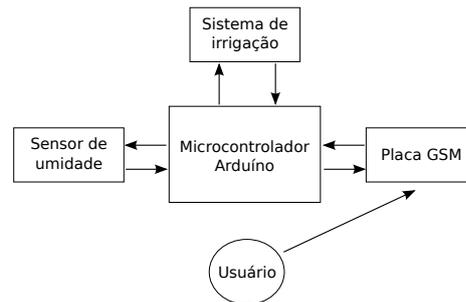


Figura 5.4: Interconexão entre os componentes do sistema utilizando o microcontrolador Arduino. Adaptado de (MILLS et al., 2013).

ponibilize opções para o fornecimento de energia utilizando um painel solar e uma bateria, tornando o protótipo autossuficiente. A tabela 5.1 ilustra a comparação entre os trabalhos relacionados e o presente trabalho, bem como as características tratadas por cada um deles.

Tabela 5.1: Comparativo entre trabalhos relacionados e o presente trabalho.

	Agricultura de Precisão	Irrigação de Precisão	Arquitetura Sensível ao Contexto	Modelo de Representação de Contexto	Protótipo desenvolvido
Kjaer et al.(2008)	x	-	x	-	x
Mohammadrezaei et al. (2011)	x	-	x	-	-
Prasad et al. (2012)	x	x	-	-	x
Mills et al.(2013)	x	x	-	-	x
Trabalho Proposto	x	x	x	x	x

Os critérios apresentados na tabela 5.1, foram escolhidos através da análise de quais pontos são relevantes para a realização da comparação entre os trabalhos. Segundo Kjaer et al. (2008), diversos trabalhos realizados são voltados para arquiteturas e aplicações sensíveis ao contexto destinados ao uso em ambientes fechados, como escritórios ou laboratórios. Com isso, não conseguem abranger os requisitos necessários ao domínio agrícola (KJAER, 2008). Por isso, é importante a análise de trabalhos que sejam voltados para a *Agricultura de Precisão*, e principalmente para a *Irrigação de Precisão*. Ainda segundo Kjaer et al. (2008), a implementação de sistemas ou *arquiteturas sensíveis ao contexto* são importantes por permitirem a coleta de diversas informações de contexto que podem auxiliar o agricultor de diversas formas, como por exemplo, facilitando o acesso a informação independentemente do lugar em que o mesmo se encontra. Para isso, um *modelo de representação de contextos* auxilia no desenvolvimento e implementação destas arquiteturas. Além disso, uma das alternativas para a validação destas arquiteturas, é a implementação de *protótipos*.

6 CONSIDERAÇÕES FINAIS

A agricultura irrigada tem se tornado uma importante estratégia para a otimização da produção de alimentos no mundo, auxiliando no desenvolvimento sustentável no campo e possibilitando a geração de empregos (MANTOVANI; BERNARDO; PALARETTI, 2007). Entre as alternativas para determinação do momento e da quantidade de água aplicada na irrigação, a realização do manejo com base no monitoramento da umidade do solo, é uma importante opção para auxiliar na racionalização da água e no aumento da produtividade no campo. Com este objetivo, cada vez mais as tecnologias de sensoriamento remoto são utilizadas na irrigação de precisão para realizar a obtenção e interpretação de dados à distância (LIAGHAT; BALASUNDRAM, 2010).

Dentro deste escopo, esse trabalho propôs e desenvolveu um medidor de umidade e pluviometria baseado em uma arquitetura de sensoriamento remoto sensível ao contexto utilizando tecnologias *open hardware*, que possui uma taxonomia de contextos que define os tipos de informações utilizadas na arquitetura. Para a validação da arquitetura proposta, dois protótipos do medidor foram desenvolvidos conforme seções 4.1 e 4.2. Além disso, conforme seções 4.3 e 4.4, dois estudos de caso foram realizados, e através dos testes apresentados, verificou-se que a arquitetura é viável, atendendo aos requisitos aos quais se propôs, através do funcionamento dos protótipos com a utilização da rede de energia elétrica e também com a utilização de alternativas de alimentação externa de energia (painel solar e bateria).

Foram apresentados os fatores de determinação entre os sensores de umidade do solo, com a utilização das funções de regressão linear e das redes neurais multilayer perceptron presentes no programa Weka, efetuando a comparação dos dados originais coletados e dos dados com ajustes de temperatura. Conclui-se que ambas as alternativas utilizadas para o ajuste das informações dos sensores resistivos SI de umidade do solo tiveram resultados satisfatórios, com valores de até 95% para os coeficiente de determinação. Na camada de 10 centímetros, notou-se a importância do ajuste de temperatura para elevar os coeficientes de determinação. O mesmo acontece na camada de 20 centímetros, entretanto, verificou-se também que para esta camada, é interessante a utilização de todas as informações de contexto disponíveis, incluindo as informações da camada superior (10 centímetros), para elevar os resultados obtidos. Através destas análises, foram geradas fórmulas para a adequação dos valores coletados para um solo específico, levando em consideração os sensores CS616 da Campbell que são amplamente utilizados

no campo.

Com isso, demonstra-se que é viável o desenvolvimento e utilização dos protótipos desenvolvidos no presente trabalho, por utilizar tecnologias *open hardware* que possibilitam a academia a sua utilização e reprodução. Além disso, são utilizados sensores e equipamentos de custo reduzido, possibilitando uma alternativa viável para o monitoramento das áreas cultivadas com um protótipo autossuficiente que pode utilizar energia solar para o seu abastecimento.

6.1 Trabalhos futuros

Sobre os aspectos discutidos nos capítulos 4, 5, e 6, identificam-se diversos aperfeiçoamentos que podem ser realizados, sugerindo-se como trabalhos futuros:

1. Primeiramente, podem ser pesquisadas alternativas para diminuir o consumo de energia para que os medidores possam permanecer ligados em pleno funcionamento, permitindo sua utilização principalmente em regiões com pouca incidência de sol por mais tempo. Além disso, o código pode ser melhorado para permitir que algumas configuração dos protótipos sejam feitas à distância.
2. Implementar na camada de aplicação os gráficos que apresentam as informações coletadas, disponibilizando o acesso as informações ao produtor rural em qualquer lugar, auxiliando na tomada de decisões referente a irrigação.
3. Efetuar testes com mais tipos de solos e condições climáticas adversas, bem como colocar mais camadas de sensores no solo, para verificar o comportamento dos mesmos em diversas condições.
4. No presente trabalho, o medidor de umidade e pluviometria auxilia no manejo da irrigação através do monitoramento do solo conforme descrito na seção 2.2.1.2. Porém, uma alternativa para aprimorar os resultados mostrados ao produtor rural na camada de aplicação, é através da utilização das informações de pluviometria coletados pelo medidor do presente trabalho, para aprimorar os cálculos realizados pelo sistema Alfa Seed Software desenvolvido por Cerutti (2014). O Alfa Seed Software utiliza-se de informações das estações INMET, realizando o monitoramento do clima conforme citado na seção 2.2.1.3, possibilitando: i) a exibição de diversos dados meteorológicos; ii) a realização do cálculo de informações relevantes como a ETo, ETc, irrigação requerida e produtividade esti-

mada; iii) geração de gráficos com informações de temperatura do ar, umidade, pressão, radiação, ETo, ETc, irrigação requerida e produtividade estimada entre outros (conforme Figura 6.1).

Data	Temp. Ar	Temp. Min.	Temp. Máx.	Umidade	Umid. Min.	Umid. Máx.	Pressão	Vel. Vento	Radiação	Luz Solar	Chuva Ef.	Estágio	GDA	ETo	Kc	Etc	Irrig. req.
25/06/2014	16.9	15.9	18.3	98.7	97	99	133.36	2.07	93.61	9	14.72	Intermediário	880	0.86	1.15	0.99	-13.73
26/06/2014	15.5	14.9	16.2	97.8	94	99	130.88	2	82.74	9	1.92	Intermediário	888.55	0.82	1.15	0.95	-0.97
27/06/2014	16.1	15.0	17.2	96.3	92	99	129.98	2.84	63.68	9	2.40	Intermediário	897.65	0.86	1.15	0.98	-1.42
28/06/2014	14.9	14.1	15.7	96.7	92	99	134.36	4.38	35.24	9	67.52	Intermediário	905.55	0.79	1.15	0.9	-66.62
29/06/2014	13.8	12.9	15.1	98.2	97	99	133.18	2.89	76.96	9	66.24	Intermediário	912.55	0.7	1.15	0.81	-65.43
30/06/2014	12.8	11.8	14.0	95.3	88	98	133.72	2.5	116.81	10	6.24	Intermediário	918.45	1.01	1.15	1.16	-5.06
01/07/2014	11.4	7.1	14.9	91	70	99	134.12	1.14	379.41	10	0.00	Intermediário	922.45	1.16	1.15	1.34	1.34
02/07/2014	10.2	5.9	16.0	95.7	81	100	133.97	1.9	292.7	10	0.32	Intermediário	926.4	1.21	1.15	1.39	1.07
03/07/2014	16.4	13.4	21.7	95.9	85	100	133.79	1.68	319.1	9	16.80	Intermediário	936.95	1.58	1.15	1.82	-14.98
04/07/2014	20.9	17.9	26.6	90.6	61	100	133.65	2.08	252.14	10	37.76	Intermediário	952.2	2.54	1.15	2.93	-34.83
05/07/2014	26.5	21.1	29.8	55.8	43	76	133.23	5.53	482.13	11	0.00	Intermediário	970.65	6.97	1.15	8.02	8.02
06/07/2014	16.1	13.5	24.6	97.2	66	99	133.8	1.44	0	0	28.00	Intermediário	982.7	1.16	1.15	1.34	-26.66

📈 Produtividade Máxima: 75 ton/ha
📈 Produtividade Estimada: 71.625 ton/ha

Figura 6.1: Resultados do Alfa Seed Software (CERUTTI, 2014).

O Alfa Seed Software é uma boa alternativa para ser utilizado na camada de aplicação da arquitetura, pois foi desenvolvido baseado em um estudo acerca dos programas existentes com objetivo semelhante (ETo Calculator, CROPWAT 8.0 e AquaCrop), tendo como vantagens: sua portabilidade, entrada de dados com formato padronizado que permite a interação com outras aplicações, e ser uma ferramenta *open source*.

Com a utilização do Alfa Seed Software, a camada de aplicação fica mais completa, pois irá disponibilizar ao produtor rural duas opções de monitoramento: monitoramento do solo conforme descrito na seção 2.2.1.2 efetuado no presente trabalho, e monitoramento do clima conforme citado na seção 2.2.1.3 utilizando o Alfa Seed Software.

- Realizar um estudo referente ao consumo de água e da produtividade de algumas culturas com métodos tradicionais de irrigação, em comparação com a irrigação realizada baseado nas informações disponibilizadas pela camada de aplicação da arquitetura do presente trabalho.
- Implementar uma aplicação que seja capaz de gerar alertas para o produtor rural, informando se algum problema com o medidor de umidade aconteceu, através do envio de um SMS por exemplo. Ainda pode ser enviado um relatório diário com os gráficos gerados para o e-mail do produtor rural. Também podem ser implementados atuadores que

conforme a configuração estabelecida (limite de umidade no solo para uma determinada cultura), automaticamente realizem a irrigação.

Os protótipos desenvolvidos, juntamente com os pontos sugeridos como trabalhos futuros deverão formalizar uma arquitetura completa, a qual deverá possibilitar sua aplicação na prática em um ambiente rural. Por fim, cabe salientar que a arquitetura proposta não possui todos os elementos desenvolvidos completamente em virtude de sua complexidade e da quantidade de aperfeiçoamentos a serem realizados. Em função disso, a validação completa da arquitetura junto aos produtores rurais é deixada como uma tarefa futura.

6.2 Publicações

Abaixo segue uma lista com os artigos relacionados a esta dissertação que foram publicados:

1. Boufleuer, R., Del Fabro Neto, A., Romero de Azevedo, B., Lima, J. C. D., Augustin, Petry, M. T. I., Carlesso, R., Basso, L. J., Haubert, D. I. *Evaluation of a Remote Sensing Architecture for Precision Irrigation Using Open Hardware Technologies*. In: The 2015 International Conference on Wireless Networks, 2015.

- Qualis: B2.
- Aceito para publicação.

2. Boufleuer, R., Romero de Azevedo, B., Del Fabro Neto, A., Lima, J. C. D., Augustin, I., Carlesso, R., Petry, M. T. *Desenvolvimento de um medidor de umidade e pluviometria baseado em uma arquitetura de sensoriamento remoto sensível ao contexto para agricultura de precisão*. In: VI Simpósio Brasileiro de Computação Ubíqua e Pervasiva, 2014, 14. p. 858-867.

- Qualis: B5.

Demais artigos publicados:

1. Del Fabro Neto, A., Boufleuer, R., Romero de Azevedo, B., Augustin, I., Lima, J. C. D., e C. Rocha, C. (2013). *Towards a middleware to infer the risk level of an activity in context-aware environments using the srk model*. UBICOMM 2013, The Seventh International

Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, páginas 38–42.

- Qualis: B3.

2. Romero de Azevedo, B., Del Fabro Neto, A., Boufleuer, R., Augustin, I. e Lima, J. C. D. (2014). *Modelo para classificação do esforço cognitivo de atividades humanas em um sistema sensível ao contexto*. SBCUP - VI Simpósio Brasileiro de Computação Ubíqua e Pervasiva.

- Qualis: B5;
- Escolhido como um dos melhores artigos do evento.

3. Romero de Azevedo, B., Del Fabro Neto, A., Boufleuer, R., Augustin, I. e Lima, J. C. D. (2015). *Classification model for the cognitive workload of human activity in a context-aware system*. Journal of Applied Computing Research, ISSN: 2236-8434. <http://dx.doi.org/10.4013/jacr.2014.41.02>

- Qualis: B5;

4. Del Fabro Neto, A., Boufleuer, R., Romero de Azevedo, B., Augustin, I., Lima, J. C. D., e Pasin, M. *An approach based on activity theory and the SRK model for risk and performance evaluation of human activities in a context-aware middleware*. In : Proceedings of the 13th International Conference on Mobile and Ubiquitous Multimedia. ACM, 2014. p. 40-47. dx.doi.org/10.1145/2677972.2677982

- Qualis: B2.

5. Del Fabro Neto, A., Romero de Azevedo, B., Boufleuer, R., Lima, J. C. D., Augustin. *Model for Action Prediction and Inference of Risk Situation in Smart Environments*. In: The 2015 International Conference on Wireless Networks, 2015.

- Qualis: B2;
- Aceito para publicação.

6. Del Fabro Neto, A., Romero de Azevedo, B., Boufleuer, R., Lima, J. C. D., Augustin, I. *Modelo para predição de ações e inferência de situações de risco em ambientes sensíveis*

ao contexto. In: 7^o SBCUP - Simpósio Brasileiro de Computação Ubíqua e Pervasiva, 2015.

- Qualis: B5.
- Aceito para publicação.

7. Barcelos, P. P., Charão, A. S., Boufleuer, R., Lima, J. C. *Analysis of Fault Injection Approaches in Apache Hadoop*. In: The 2015 Int'l Conf on Parallel and Distributed Processing Techniques and Applications, 2015.

- Qualis: B2.
- Aceito para publicação.

REFERÊNCIAS

- AHMED, Z. **Design of Autonomous Low Power Sensor for Soil Moisture Measurement**. 2012. Dissertação (Mestrado em Ciência da Computação) — Linköping University, Electronics System, The Institute of Technology, 2012.
- ALBERGEL, C. et al. Evaluation of remotely sensed and modelled soil moisture products using global ground-based in situ observations. **Remote Sensing of Environment**, [S.l.], v.118, p.215–226, 2012.
- ALLEN, R. G. et al. Crop evapotranspiration-Guidelines for computing crop water requirements-FAO Irrigation and drainage paper 56. **FAO, Rome**, [S.l.], v.300, p.6541, 1998.
- APACHE. **Apache HTTP Server Project**. Disponível em: <<http://www.apache.org/>>. Acesso em: 20 dez. 2014.
- ARDUINO. **Arduino Platform**. Disponível em: <<http://arduino.cc/>>. Acesso em: 12 ago. 2014.
- AUGUSTIN, I. **Abstrações para uma linguagem de programação visando aplicações móveis em um ambiente de Pervasive Computing**. 2004. Tese (Doutorado em Ciência da Computação) — Universidade Federal do Rio Grande do Sul. Instituto de Informática. Curso de Pós-Graduação em Ciência da Computação, Porto Alegre, 2004.
- BALDAUF, M.; DUSTDAR, S.; ROSENBERG, F. A survey on context-aware systems. **International Journal of Ad Hoc and Ubiquitous Computing**, [S.l.], v.2, n.4, p.263–277, 2007.
- BÖHME, B.; BECKER, M.; DIEKKRÜGER, B. Calibrating a FDR sensor for soil moisture monitoring in a wetland in Central Kenya. **Physics and Chemistry of the Earth, Parts A/B/C**, [S.l.], v.66, p.101–111, 2013.
- BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. **UML: guia do usuário**. [S.l.]: Elsevier Brasil, 2006.
- BREVIK, E. C.; FENTON, T. E.; HORTON, R. Effect of daily soil temperature fluctuations on soil electrical conductivity as measured with the Geonics® EM-38. **Precision Agriculture**, [S.l.], v.5, n.2, p.145–152, 2004.

CALAMITA, G. et al. Electrical resistivity and TDR methods for soil moisture estimation in central Italy test-sites. **Journal of Hydrology**, [S.l.], v.454, p.101–112, 2012.

CAMPBELL. **PS12 Power Supplies and Charging Regulators. Copyright© 2002-2014 Campbell Scientific, Inc.** Disponível em: <<http://s.campbellsci.com/documents/us/manuals/ps12la.pdf>>. Acesso em: 30 ago. 2014.

CAMPBELL-SCIENTIFIC. **Instruction Manual - MSX10, MSX20, MSX20R, MSX30R, and MSX50R Solar Panels. Copyright© 2002-2015 Campbell Scientific, Inc.** Disponível em: <http://s.campbellsci.com/documents/ca/manuals/msx_man.pdf>. Acesso em: 20 dez. 2014.

CARLESSO, R. et al. **Usos e Benefícios da Coleta Automática de Dados Meteorológicos na Agricultura.** [S.l.]: Editora UFSM, 2007.

CARLESSO, R.; PETRY, M.; TROIS, C. The use of a meteorological station network to provide crop water requirement information for irrigation management. In: **Computer and Computing Technologies in Agriculture II, Volume 1.** [S.l.]: Springer, 2009. p.19–27.

CERUTTI, D. H. H. **Desenvolvimento de um Modelo "Água-Produção" para Culturas de Expressão Econômica no Brasil visando a Produtividade da Água em Áreas irrigadas.** Monografia (Sistemas de Informação) – Universidade Federal de Santa Maria, Santa Maria, 2014.

CHEN, G.; KOTZ, D. et al. **A survey of context-aware mobile computing research.** [S.l.]: Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College, 2000.

CORWIN, D.; LESCH, S. Apparent soil electrical conductivity measurements in agriculture. **Computers and electronics in agriculture**, [S.l.], v.46, n.1, p.11–43, 2005.

CS616. **Instruction Manual - CS616 and CS625 Water Content Reflectometers. Copyright© 2002-2014 Campbell Scientific, Inc.** Disponível em: <<http://s.campbellsci.com/documents/us/manuals/cs616.pdf>>. Acesso em: 20 nov. 2014.

DACCACHE, A. et al. Implementing precision irrigation in a humid climate—Recent experiences and on-going challenges. **Agricultural Water Management**, [S.l.], 2014.

DALLAS. **DS1307 Tiny RTC I2C Real Time Clock**. Disponível em: <<http://www.dipmicro.com/?datasheet=DS1307.pdf>>. Acesso em: 30 dez. 2014.

DE JEU, R. et al. Global soil moisture patterns observed by space borne microwave radiometers and scatterometers. **Surveys in Geophysics**, [S.l.], v.29, n.4-5, p.399–420, 2008.

DEY, A. K. **Providing architectural support for building context-aware applications**. 2000. Tese (Doutorado em Ciência da Computação) — Georgia Institute of Technology, 2000.

DJANGO. **Django Framework**. Disponível em: <<https://www.djangoproject.com/>>. Acesso em: 22 dez. 2014.

ELLIOTT, R. et al. The role of automated weather networks in providing evapotranspiration estimates. In: DECENNIAL NATIONAL IRRIGATION SYMPOSIUM, 4. **Proceedings...** [S.l.: s.n.], 2000. p.243–250.

GONZALEZ-SANCHEZ, A.; FRAUSTO-SOLIS, J.; OJEDA-BUSTAMANTE, W. Predictive ability of machine learning methods for massive crop yield prediction. **Spanish Journal of Agricultural Research**, [S.l.], v.12, n.2, p.313–328, 2014.

HARGREAVES, G. H. et al. Food, water, and a possible world crisis. In: NATIONAL IRRIGATION SYMPOSIUM. PROCEEDINGS OF THE 4TH DECENNIAL SYMPOSIUM, PHOENIX, ARIZONA, USA, NOVEMBER 14-16, 2000. **Anais...** [S.l.: s.n.], 2000. p.187–194.

HARNETT, C. Open source hardware for instrumentation and measurement. **Instrumentation & Measurement Magazine, IEEE**, [S.l.], v.14, n.3, p.34–38, 2011.

HAULE, J.; MICHAEL, K. Deployment of wireless sensor networks (WSN) in automated irrigation management and scheduling systems: a review. In: SCIENCE, COMPUTING AND TELECOMMUNICATIONS (PACT), 2014 PAN AFRICAN CONFERENCE ON. **Anais...** [S.l.: s.n.], 2014. p.86–91.

HONG, J.-y.; SUH, E.-h.; KIM, S.-J. Context-aware systems: a literature review and classification. **Expert Systems with Applications**, [S.l.], v.36, n.4, p.8509–8522, 2009.

HOWELL, T. et al. Irrigation's role in enhancing water use efficiency. In: OF THE 4TH DECENNIAL NATIONAL IRRIGATION SYMPOSIUM. **Proceedings...** [S.l.: s.n.], 2000. p.66–80.

ICOMSAT. **GSM/GPRS shield for Arduino based on the SIM900**. Disponível em: <<http://wiki.itheadstudio.com/IComSat>>. Acesso em: 30 ago. 2014.

INMET. **Instituto Nacional de Meteorologia**. Disponível em: <www.inmet.gov.br/>. Acesso em: 12 ago. 2014.

IRRIGA. **Sistema Irriga®**. Disponível em: <<https://www.sistemairriga.com.br>>. Acesso em: 30 ago. 2014.

JUNIOR, M. M.; NUNES, R. O.; CELINSKI, V. G. Comparison of the responses of low cost electrical soil sensors, and a arduino microcontroller plataform. **Iberoamerican Journal of Applied Computing**, [S.l.], v.2, n.1, 2013.

KJAER, K. E. Designing middleware for context awareness in agriculture. In: MIDDLEWARE DOCTORAL SYMPOSIUM, 5. **Proceedings...** [S.l.: s.n.], 2008. p.19–24.

KUROSE, J. F. et al. **Redes de computadores ea Internet: uma abordagem top-down**. [S.l.]: Pearson, 2010. 411p.

LAIOLO, P. et al. Validation of remote sensing soil moisture products with a distributed continuous hydrological model. In: GEOSCIENCE AND REMOTE SENSING SYMPOSIUM (IGARSS), 2014 IEEE INTERNATIONAL. **Anais...** [S.l.: s.n.], 2014. p.3319–3322.

LIAGHAT, S.; BALASUNDRAM, S. K. A Review: the role of remote sensing in precision agriculture. **American Journal of Agricultural & Biological Science**, [S.l.], v.5, n.1, 2010.

LIMA, J. C. D. **Sistema de recomendação sensível ao contexto para apoio à autenticação implícita em ambientes móveis e pervasivos baseado em conhecimento comportamental do usuário**. 2013. Tese (Doutorado em Ciência da Computação) — Universidade Federal de Santa Catarina, Florianópolis, 2013.

MA, R. et al. Comparing temperature correction models for soil electrical conductivity measurement. **Precision Agriculture**, [S.l.], v.12, n.1, p.55–66, 2011.

MANTOVANI, E. C.; BERNARDO, S.; PALARETTI, L. F. **Irrigação: princípios e métodos**. [S.l.]: UFV, 2ª Edição, 2007.

MARGOLIS, M. **Arduino Cookbook**. [S.l.]: O'Reilly Media, 2nd Edition, 2011.

MILLS, G. A. et al. GSM Based Irrigation Control and Monitorin System. **International Journal of Engineering Science & Technology**, [S.l.], v.5, n.7, 2013.

MOHAMMADREZAEI, M.; FATHI, M.; ATTARZADEH, N. Proposing a new approach to applying pervasive computing in agriculture environments. **Global Journal of Computer Science and Technology**, [S.l.], v.11, n.16, 2011.

PESSL. **Metos Compact®. Users' Manual PESSL INSTRUMENTS**. Disponível em: <http://efesaro.com/pdf/Manual_Metos_Compact_ingles.pdf>. Acesso em: 30 ago. 2014.

PORTAPOW. **Portable Power Supplies**. Disponível em: <<http://www.portablepowersupplies.co.uk/>>. Acesso em: 25 fev. 2015.

POSTGRESQL. **PostgreSQL Open Source Object-Relational Database System**. Disponível em: <<http://www.postgresql.org/>>. Acesso em: 23 dez. 2014.

PRASAD, K. et al. Water-Saving Irrigation System Based on Automatic Control by Using GSM Technology. **Middle-East Journal of Scientific Research**, [S.l.], v.12, n.12, p.1824–1827, 2012.

ROBINSON, D. et al. Soil moisture measurement for ecological and hydrological watershed-scale observatories: a review. **Vadose Zone Journal**, [S.l.], v.7, n.1, p.358–389, 2008.

RODRIGUES, G. **Considerações sobre os impactos ambientais da agricultura irrigada**. [S.l.]: Embrapa Meio Ambiente, 2004.

ROTIMI OJO, E. et al. Calibration and Evaluation of a Frequency Domain Reflectometry Sensor for Real-Time Soil Moisture Monitoring. **Vadose Zone Journal**, [S.l.], 2015.

SALAZAR, R. et al. Irrigation System through Intelligent Agents Implemented with Arduino Technology. **ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal**, [S.l.], v.1, n.6, p.29–36, 2013.

SCHILIT, B.; ADAMS, N.; WANT, R. Context-aware computing applications. In: **MOBILE COMPUTING SYSTEMS AND APPLICATIONS**, 1994. WMCSA 1994. FIRST WORKSHOP ON. **Anais...** [S.l.: s.n.], 1994. p.85–90.

SCUDIERO, E. et al. Simultaneous monitoring of soil water content and salinity with a low-cost capacitance-resistance probe. **sensors**, [S.l.], v.12, n.12, p.17588–17607, 2012.

SIM900. **SIM900 AT Command Manual. Version:** 1.03. copyright shanghai simcom wireless solutions ltd. 2010.

TE525. **TE525 Tipping Bucket Rain Gage. Copyright© 2002-2015 Campbell Scientific, Inc.** Disponível em: <<http://s.campbellsci.com/documents/us/manuals/te525.pdf/>>. Acesso em: 25 mai. 2015.

THALHEIMER, M. A low-cost electronic tensiometer system for continuous monitoring of soil water potential. **Journal of Agricultural Engineering**, [S.l.], v.44, n.3, p.e16, 2013.

UBUNTU. **Sistema Operacional Ubuntu.** Disponível em: <<http://ubuntu-br.org/>>. Acesso em: 20 dez. 2014.

VILLORDON, A. et al. Combining linear regression and machine learning approaches to identify consensus variables related to optimum sweetpotato transplanting date. **HortScience**, [S.l.], v.45, n.4, p.684–686, 2010.

VIRTUALBOX. **VirtualBox Virtual Machines.** Disponível em: <<https://www.virtualbox.org/>>. Acesso em: 20 jan. 2015.

WAGNER, W. et al. Soil moisture from operational meteorological satellites. **Hydrogeology Journal**, [S.l.], v.15, n.1, p.121–131, 2007.

WEISER, M. The computer for the 21st century. **Scientific american**, [S.l.], v.265, n.3, p.94–104, 1991.

WEKA. **Data Minig Software.** Disponível em: <<http://www.cs.waikato.ac.nz/ml/weka/>>. Acesso em: 25 fev. 2015.

ZACHARIADIS, S.; KASKALIS, T. H. An Embedded System for Smart Vineyard Agriculture. **2nd Pan-Hellenic Conference on Eletronics and Telecommunications**, [S.l.], 2012.

ZHAO, Q.; LIU, Y. Remote Sensing Monitoring System for Agriculture Condition. In: COMPUTATIONAL AND INFORMATION SCIENCES (ICCIS), 2012 FOURTH INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2012. p.966–968.