When using mobile devices for learning, the context of the learner can change. This change may affect how and what is learnt. This dissertation provides a view into the field which investigates this effect: context-aware learning.

A framework was developed and used to create two prototypes. Their successful implementation and testing prove the overall effectiveness and usability of the framework as a research tool in the development of context-aware learning systems.

RICHARD A. W. TORTORELLA
FRAMEWORK FOR CONTEXT-AWARE LEARNING SYSTEMS
Richard A. W. Tortorella

FRAMEWORK FOR CONTEXT-AWARE LEARNING SYSTEMS

Publications of the University of Eastern Finland
Dissertations in Forestry and Natural Sciences
No 298

University of Eastern Finland
Joensuu
2017

Academic dissertation
To be presented by permission of the Faculty of Science and Forestry for public examination in the Auditorium M100 in the Metria Building at the University of Eastern Finland, Joensuu, on January, 12, 2018, at 13:00
Opponent: Professor Hiroaki Ogata, Ph.D.
Kyoto University
Academic Center for Computing and Media Studies, Graduate School of Informatics
Yoshida-nihonmatsu, Sakyo-ku,
606-8501, Kyoto, JAPAN
email: hiroaki.ogata@gmail.com
The field of context-awareness is continually advancing due to the proliferating and ubiquitous nature of mobile computing technologies. Such advancements are also evident in the field of context-aware learning systems. Many context-aware learning systems are mobile learning environments, which are used in a wide range of educational settings and adapt to learners’ ever-changing environmental context.

The works presented in this dissertation provide a glimpse into the research field of context-aware learning. The development of the overview of this field began with the building of a preliminary context-aware adaptive learning system. The challenges discovered during the creation of the preliminary system demonstrated the need for a systematic, comprehensive review and analysis of all of the literature in the field of context-aware learning. This resulted in the creation of a literature review framework, which was then applied in order to analyse the context-aware learning field from 2009 to 2015.

The literature review analysis, in turn, led to the creation of a framework for the development of context-aware learning systems. This context-aware learning system framework was designed to provide structure and repeatability, two main issues in the field uncovered during the literature review. The context-aware learning system framework allows for the creation of any number of varied context-aware learning systems. The components of the framework are intended to be repeatable and can be adapted to systems with a variety of learning objectives and hardware and setting requirements.

The context-aware system framework was successfully implemented in two prototypes: the Knowledge Inference Training Terminal (KITT) and the Pathogen Outbreak Prevention Instruction System (PORPOISE). Both the KITT and the PORPOISE were successfully evaluated through two separate evaluation studies, which produced very favourable results. Both systems also received very high scores in terms of overall effectiveness and usability, demonstrating the framework’s capabilities as a valid research and programming tool.
The final section of this dissertation discusses possible directions for the field of context-aware learning systems. Future directions of the field, as well as possible drawbacks and constraints related to integrating cloud computing into context-aware learning systems, are discussed. Overall, this dissertation demonstrates not only the merits of context-aware learning technologies, but also the bright future of context-aware learning systems.

**Universal Decimal Classification:** 004.78, 004.9, 37.091.33, 621.395.721.5

**Library of Congress Subject Headings:** Ubiquitous computing; Context-aware computing; Mobile computing; Mobile communication systems in education; Instructional systems; Learning; Medical education; Pathogenic microorganisms; Design; Classification; Evaluation; Cloud computing

**Yleinen suomalainen asiasanasto:** opetusteknologia; tietokoneavusteinen oppiminen; mobiilisovellukset; oppiminen; mobiilioppiminen; taudinaheuttajat; suunnittelu; luokitus; arviointi; pilvipalvelut
ACKNOWLEDGEMENTS

Working on and completing a doctoral degree is by no means a solo effort. Although, as the author, I am afforded the luxury of taking credit, there are many others who have worked behind the scenes to see this work come to fruition. There are countless people who have made this work possible, from the hard-working staff at the School of Computing at the University of Eastern Finland to my family and friends. I owe them all a debt of gratitude that I can never begin to repay.

In particular, I wish to take this moment to thank those who have played a pivotal role in this work and in my life over the last four years.

Let me begin with my supervisor Prof. Kinshuk. Words alone cannot express my gratitude towards Prof. Kinshuk for his guidance, supervision, mentorship and friendship along the path to completion of my dissertation and research. Without Prof. Kinshuk’s patient demeanour and gentle nudging, this entire journey would never have been possible. Prof. Kinshuk has seen me at every stage (both academic and emotional) along the way. If a doctoral degree is the path of enlightenment, Prof. Kinshuk has been my ever-present beacon lighting my path.

My thanks also go to Prof. Vivekanandan Kumar, whose insight and advice has been fantastic throughout my research. I also wish to thank Dr. Jarkko Suhonen for putting up with my countless emails and questions over the past four years. Dr. Suhonen welcomed me with open arms when I arrived in Joensuu as part of my workshop Ph.D. application, and his support has never waned. Additionally, I wish to thank Prof. Nian-Shing Chen, Dr. Sabine Graf and Prof. Tukiainen for their valuable insight, advice and assistance throughout the many aspects of my research. I would also like to express my thanks to my friend Miles Gibson for his support and encouragement from day one.

Although it may be unconventional, I wish to thank the musical group RUSH. Unbeknownst to them, their music has been a constant companion and inspiration, ever present as background music during my countless hours in front of the computer during my research. I think the following quote from RUSH’s song “Mission” from Hold Your Fire sums up my feelings:

In the grip of a nameless possession, a slave to the drive of obsession—a spirit
with a vision is a dream with a mission

As anyone who knows me, it should come as no surprise that I wish to thank the most important people in my world: my family. Once again, words cannot express my undying love and gratitude for their constant love, support and patience. To my lovely wife, Tanya, and my children, Alissa and Samantha, who have sacrificed so much of their time and family activities to allow me to complete this monumental task: I simply could not have done this without you! In many ways, this work is as
much yours as it is mine. You three are the foundation of who I am and the most important parts of me. You have my undying love and gratitude.

Finally, I wish to thank my loving parents, Salvatore and Mary. In my over forty-one revolutions of the sun, they have never lost faith in me nor my abilities. Their never-ending support and love have pushed me along every single step of the way. Mummy and Papa, it is with much love, gratitude and pride that I dedicate this work to you both.

Joensuu, 1st August 2017
R. A. W. Tortorella
# LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO</td>
<td>carbon monoxide</td>
</tr>
<tr>
<td>EMF</td>
<td>electromagnetic field</td>
</tr>
<tr>
<td>FSLSM</td>
<td>Felder-Silverman learning style model</td>
</tr>
<tr>
<td>GPIO</td>
<td>general purpose input output</td>
</tr>
<tr>
<td>GPS</td>
<td>global positioning system</td>
</tr>
<tr>
<td>I2C</td>
<td>inter-integrated circuit</td>
</tr>
<tr>
<td>ILS</td>
<td>index of learning style</td>
</tr>
<tr>
<td>iOS</td>
<td>iPhone operating system</td>
</tr>
<tr>
<td>K-12</td>
<td>kindergarten to grade 12</td>
</tr>
<tr>
<td>LCD</td>
<td>liquid crystal display</td>
</tr>
<tr>
<td>LED</td>
<td>light emitting diode</td>
</tr>
<tr>
<td>OLED</td>
<td>organic light-emitting diode</td>
</tr>
<tr>
<td>OSX</td>
<td>operating system Version 10+</td>
</tr>
<tr>
<td>PDA</td>
<td>personal digital assistant</td>
</tr>
<tr>
<td>QR Code</td>
<td>quick response code</td>
</tr>
<tr>
<td>RPi</td>
<td>raspberry pi</td>
</tr>
<tr>
<td>RFID</td>
<td>radio frequency identification</td>
</tr>
<tr>
<td>SSID</td>
<td>service set identifier</td>
</tr>
<tr>
<td>SUS</td>
<td>system usability scale</td>
</tr>
<tr>
<td>Wi-Fi</td>
<td>wireless fidelity</td>
</tr>
</tbody>
</table>
LIST OF ORIGINAL PUBLICATIONS

This thesis is based on data presented in the following articles, referred to by the Roman Numerals I-VI.


AUTHOR’S CONTRIBUTION

I) The author designed, programmed and implemented the evaluation of the system. The co-author provided assistance, guidance and editing throughout the entire process.

II) The author, together with the co-authors, devised the classification framework. The author implemented the framework on the subject matter of context-aware mobile learning between 2009 and 2015.

III) The author was the primary editor for the chapter and wrote a section relevant to the chapter’s contents.

IV) The author, together with the co-authors, devised the framework. The author then implemented the framework, created the hardware and coding and was responsible for the implementation and testing of the framework. The co-authors provided guidance and editing throughout the entire process.

V) The author was primarily responsible for the creation of the hardware and the coding and was also responsible for the implementation and testing of the framework. The co-author provided guidance and editing throughout the entire process.

VI) The author was the primary author of the chapter. The co-authors offered guidance throughout the chapter’s creation.
CONTENTS

ABSTRACT ......................................................................................................................... 7

ACKNOWLEDGEMENTS ................................................................................................. 9

1 INTRODUCTION ........................................................................................................ 19

2 RESEARCH QUESTIONS AND DESIGN .................................................................... 23

3 EARLY STAGES: INITIAL DEVELOPMENT .................................................................. 27

3.1 EVALUATION ............................................................................................................. 27

4 BACKGROUND: LITERATURE REVIEW FRAMEWORK .............................................. 29

4.1 PAPER REVIEW PROCESS ...................................................................................... 29

4.1.1 Journal selection process ................................................................................. 29

4.1.2 Paper selection process: Three phases .............................................................. 31

4.1.3 Initial phase ......................................................................................................... 31

4.1.4 Second and tertiary phases ............................................................................... 33

4.2 CLASSIFICATION FRAMEWORK .......................................................................... 34

4.2.1 Literature framework findings ........................................................................... 35

5 FRAMEWORK DEVELOPMENT ................................................................................. 39

5.1 EARLY STAGES OF THE FRAMEWORK ................................................................ 39

5.2 REBIRTH: FRAMEWORK CREATION AND INTERNAL STRUCTURE .................... 41

5.2.1 System setup—Component I: Sensor data acquisition ...................................... 42

5.2.2 System setup—Component II: Attribute assignment verification ...................... 42

5.2.3 Decision mechanism—Component III: Inference engine rules ......................... 43

5.2.4 Sensor Input and User Output—Component IV: Data query ............................ 45

5.2.5 Decision mechanism—Component V: Actions based on rules ......................... 46

5.2.6 Sensor Input and User Output—Component VI: User output ......................... 47

6 IMPLEMENTATION OF THE FRAMEWORK .............................................................. 49

6.1 IMPLEMENTATION: KNOWLEDGE INFERENCE TRAINING TERMINAL ............ 49

6.1.1 KITT: Hardware overview .................................................................................. 50

6.2 IMPLEMENTATION: PATHOGEN OUTBREAK PREVENTION INSTRUCTION SYSTEM ......................................................................................................................... 57

6.2.1 Medical concerns for seniors ............................................................................ 58

6.2.2 Implementation of PORPOISE .......................................................................... 58

6.2.3 System architecture ......................................................................................... 59

6.2.4 PORPOISE—Evaluation ................................................................................... 64

7 FUTURE WORK AND CONSIDERATIONS ................................................................. 67

8 BIBLIOGRAPHY .......................................................................................................... 71
1 INTRODUCTION

This dissertation highlights the journey of discovery along a research path towards a better understanding of context-aware learning systems. It contains material presented or described in a number of journal papers, book chapters and manuscripts written by the author, and it summarizes their findings.

Context-aware learning systems are mobile learning environments used in both formal and non-formal educational settings that adapt based on the device’s—and, thus, the learner’s—ever-changing environmental context. These context-aware learning systems may be standalone devices or may involve a number of server-based resources, all accessible via a user interface. The creation of these context-aware systems involves various types of technologies. These include, but are not limited to, a myriad of sensor technologies, wireless (IEEE 802.11XX) technologies, and microprocessors and user interfaces. Thus, the means by which a learner’s context is detected can vary greatly from system to system.

The origins of context-aware expert systems can be traced back several decades. One of the precursors of context-aware expert systems was ubiquitous computing. As early as the 1980s, ubiquitous computing was described as computing in which sensors and computational elements are embedded seamlessly into everyday objects (Weiser, Gold, & Brown, 1999). Ubiquitous computing provided the foundation for ubiquitous learning, an educational paradigm that focused on the needs and dynamics of learning (Cope & Kalantzis, 2008). In turn, ubiquitous learning led to context-aware ubiquitous learning, which can be described as a means for integrating context-aware technologies and allowing them to detect and adapt to the varying situations and contexts of learners in the real world (Hwang, Yang, Tsai, & Yang, 2009).

To achieve the aforementioned contextual learning, a learning system may be paired with a knowledge base and an inference engine from an expert system. An expert system is a computer program designed to achieve the same results as actual experts in a particular domain or field (Franklin, Carmody, Keller, Levitt, & Buteau, 1988). In 1989, Levi argued that one of the key benefits of expert systems is that they are potentially more accurate than human experts, since they do not suffer from the types of negative issues that may affect human performance. Levi (1989) further suggested that, given this increased accuracy, expert systems could surpass the performance of both human experts and statistical models. Yet, dealing with expert systems involved certain practical problems (Kusiak, 1989). Kusiak (1989) acknowledged the difficulty of formulating a model (which could be used in an expert system) that relies on easily available data and that, in turn, can be easily solved.

In 1987, Kusiak defined two distinct types of expert systems: the stand-alone system and the tandem expert system. A large percentage of existing expert
systems can be classified as stand-alone expert systems. These systems involve straightforward procedures that use problem-specific data and constraints to provide solutions. The other type of expert systems, tandem expert systems, are similar to stand-alone expert systems, but are linked to a database containing various models and algorithms (Kusiak, 1987, 1989). Such tandem expert systems can be considered adaptive; that is, they modify themselves to suit the problem using various models and diagrams (Kusiak, 1987, 1989). The tandem expert system was one of the predecessors of the systems found in this dissertation.

In the early 1990s, McBryan et al. (1990) described a novel and quite revolutionary implementation of an expert system that interacted with flight avionic sensors to provide a link between the mission computer, the sensors and the pilot. Specifically, the sensor management expert system permitted sensor data and target data to be displayed to the pilot (McBryan et al., 1990). A few years later, in his work on more specific and smaller-scaled sensors, Cooper (1994) suggested the application of sensors that could be linked to expert systems as a means for automatic sensor data processing. Cooper (1994) believed that multiple sensors could improve the overall performance of rate-responsive pacemakers by developing a set of rules combining activity and minute ventilation sensor inputs (Cooper, 1994). These types of rules are central to expert systems, which are based on conventional reasoning methods and knowledge representation schemes (Chen, 1994).

Today, mobile computing devices possess an ever-increasing set of powerful, yet inexpensive embedded sensors that can aid mobile learning technologies. Tan et al. (2009) described a location-based framework for mobile learning using such compact devices. Yet, for years, a common concern among researchers was that the addition of the hardware necessary for a mobile application could reduce mobile devices' overall compactness (Tan et al., 2009). The current miniaturization of smartphones has significantly alleviated this concern. Therefore, it seems that technology is acting as a driving force in the research in this field. Ever-increasing device computational power and an ever-decreasing technological footprint have made advanced context adaptation and personalization in mobile computing research an ever-increasing reality. It, therefore, should come as no surprise that the miniaturization of sensors have made mobile devices more attractive to users because of their portability, small size and computational capabilities (Stratulat & Popa, 2011). Indeed, the usage of sensors is becoming more and more widespread. In 2009, Won et al. described a system that utilized sensors to identify fastening tools and bolts. The expert system used a variety of gyroscopic sensors to calculate tilt angles and correctly identify bolt types (Won et al., 2009). These ubiquitous sensors include accelerometers, digital compasses, gyroscopes, GPS receivers, microphones and the ever-present digital camera (Lane et al., 2010).

From combining wireless sensor networks with expert systems to improve bomb detection (Prabhakaran, Sharon Rosy, & Shakena Grace, 2010) to accurately
measuring pH levels in specific samples (Capel-Cuevas, Pegalajar, de Orbe-Paya, & Capitan-Vallvey, 2012), the technological front is moving forward. Recent technological advancements are rapidly overcoming previous drawbacks, such as the lack of computational power and memory (Capel-Cuevas et al., 2012). In fact, the range of context-aware systems is ever-increasing, from context-aware systems for natural science courses (Chu, Hwang, Tsai, & Tseng, 2010) to adaptive personal fitness systems (Kranz et al., 2013). However, the context of the user is not the only variable relevant to context-aware systems. For example, El-Bishouty et al. (2010) proposed a system that adapted to the user’s location (context) and best-matched peer helpers. Similarly, Liu and Hwang’s (2010) and Huang, Yang and Liaw’s (2012) applications of context-aware learning utilized both context and additional student-specific information. Such combinations of sensor and additional data are performed by an inference engine: a component of the expert system. The inference engine is designed to find and match rules satisfied by the current contents of the data store (Singh & Karwayun, 2010). Inference engines are becoming more widely used. For example, in 2014, Hwang (2014), proposed a framework for smart learning environments utilizing an inference engine and learning tools. With a similar focus on learning, in 2015, Huang and Chiu (2015) proposed a framework to evaluate context-aware mobile learning based on meaningful learning. The research presented in this dissertation takes a different focus than previous works, emphasizing the need to increase technical details in order to build a framework for context-aware learning systems.

The first chapter of this manuscript introduces the rationale behind the research presented in this dissertation. The second chapter discusses the main driving forces, or research questions, guiding the research. The research path taken in this dissertation is presented in the third chapter (Early stages: Initial development and considerations). This chapter offers a brief introduction to the field of context-aware learning. However, in order to proceed beyond the early stages of this research and move further down the research path, as described in chapter three, it was necessary to develop a better understanding of the field of context-aware learning systems. The best way to gain a better understanding of any field of study is to perform a comprehensive review of the current state of the research in said field: that is, to perform a literature review. Any literature review begins with developing a suitable way to identify appropriate journals and papers with which to catalog and perform the review. The challenge uncovered when commencing the literature review of context-aware learning systems, however, was the lack of a pre-defined research area specific to context-aware learning systems. The solution presented in chapter four outlines the development of a literature review framework for selecting papers and establishing a structured and repeatable analysis of the field.

The lack of accepted definitions and structure appears to be a common issue in context-aware learning systems research. Indeed, another missing component uncovered along the research path was a lack of a formal pre-defined scaffold on
which to design and build existing context-aware learning systems. This lack of structure represented a considerable obstacle in being able to delve into the deeper waters of the field. As a proposed solution, chapter five depicts the creation of a framework for context-aware learning systems capable of providing much-needed structure to the field. The framework was devised from both knowledge drawn from the literature in the field and repeated aspects of systems uncovered during the literature review. The generic framework allows for the creation of any number of varied context-aware learning systems. The components of the framework are intended to be repeatable and can be adapted to suit a great number of different learning systems, allowing the systems to vary in terms of both intended learning objectives and hardware and setting requirements. The context-aware learning framework is then evaluated in chapter six via the creation of two main systems, which are themselves evaluated in real-world scenarios for both their effectiveness as a learning tool and their relative ease of use.

The future direction of the research path has yet to be determined. The final chapter shines light on the path ahead by reviewing which road signs we have passed and which possible future roadmaps the research on context-aware learning systems may use.
2 RESEARCH QUESTIONS AND DESIGN

Although the point of the journey is not always to arrive, when it comes to scientific research, there are objectives along the path that are worthy of notice. The research path described in this dissertation has four main objectives in the form of research questions, all centered around context-aware learning. These objectives provide the landmarks for the research path and describe the goals of the research and the direction taken.

The first objective is to introduce an adaptive approach for learning in mobile settings. This approach considers both learners’ learning style and their context information to determine the most appropriate learning format for each learner. The second objective is to design and implement a framework for identifying trends present within the published literature on context-aware mobile learning systems. The third objective, which builds on lessons learnt from the previous objectives, is to design, implement and evaluate a framework for the creation of generic context-aware systems. The fourth and final objective is to present and discuss concerns regarding the potential future technological directions of context-aware learning.

Table 1 summarizes the research questions asked and answered in this dissertation and discusses the various methods used to answer the questions. Additionally, Table 1 shows the various papers that encompass the research goals and identifies the chapters in which the associated papers are located.

<table>
<thead>
<tr>
<th>Research question</th>
<th>Addressed in paper(s)</th>
<th>Methods used to answer research question</th>
<th>Chapter</th>
</tr>
</thead>
<tbody>
<tr>
<td>How can one provide adaptivity in a mobile setting based on a learner’s particular learning style and contextual information?</td>
<td>1</td>
<td>Software development, implementation and evaluation</td>
<td>3</td>
</tr>
<tr>
<td>How can a framework be designed and evaluated to identify current trends in context-aware learning systems?</td>
<td>2</td>
<td>Framework development and literature analysis</td>
<td>4</td>
</tr>
<tr>
<td>How can one design, implement and evaluate a generic framework for context-aware learning systems?</td>
<td>3, 4, 5</td>
<td>Framework development, followed by software and hardware development, implementation and evaluation</td>
<td>5, 6</td>
</tr>
<tr>
<td>What does the future hold for context-aware learning systems, and what are the possible issues?</td>
<td>6</td>
<td>Literature review on the integration of new types of technologies</td>
<td>7</td>
</tr>
</tbody>
</table>
The research work comprising this dissertation was conducted from 2014 to 2017 and performed in Canada. The four years of work presented herein defines a path which includes the creation of frameworks, numerous prototypes and system evaluations. It is also an immersion into the waters of private industry: where collaborations with industry professionals are required not only to provide insight into the knowledge aspects of various systems, but also to test systems in real-world situations. As a general rule, the entire system development process incrementally built on the strengths of previously developed systems. This produced a final cumulative framework for context-aware learning systems based on knowledge learnt from the trials and tribulations of each successive iteration (Figure 1).

![Diagram of research questions](image)

**Figure 1. Development of final research questions**

Research Question 1: How can one provide adaptivity in a mobile setting based on a learner’s particular learning style and contextual information?

This was the initial research question that led to the main research topic of a framework for context-aware learning systems. It plays a key role in understanding what processes take place and what is required to identify the key aspects necessary for the development of a context-aware framework, addressed in the next research question.

In order to properly address the first research question, a system was designed and evaluated which considered two aspects of the learner:

1. The learner’s learning style, as defined by the Felder-Silverman learning style model (FSLSM) (Felder & Silverman, 1988).
2. The context of the learner in terms of environmental context parameters.

For the purposes of this research question, a system was developed on a platform built from standard off-the-shelf mobile iOS hardware. The system’s software comprised a series of components, including an adaptive engine, an account manager and a learning style model questionnaire.
A course on astronomy was developed and recorded in a number of formats (e.g. audio, video, text) in order to test the effectiveness of the adaptive system. The final evaluation of the system demonstrated that the application improved participants' subject matter comprehension by 23%.

Research Question 2: How can a framework be designed and evaluated to identify current trends in context-aware learning systems?

Given the many gaps in the extant research uncovered while answering the first research question, a systemic analysis of the state of context-aware research was required to better understand the field. The rationale behind this second research question is twofold. First, answering this question provides an overview—and, thus, a solid definition—of the domain of context-aware learning systems between 2009 and 2015 (inclusive). This gives the reader a solid foundation from which to view the entire domain and, thus, the work presented in this dissertation. The second rationale for answering the second research question is that the creation of a framework to identify current trends in context-aware learning systems will allow future researchers to apply similar methods and analyses. Such a framework will permit researchers to investigate trends beyond the publication of this manuscript and provide a foundation on which to base future literature reviews.

Paper II details the literature review framework and applies the framework to the top 20 journals in education technology and context awareness. Paper II then presents and analyses the findings of the framework. Additionally, although not included in the published paper due to length restrictions, this dissertation will include the rationale and methods to select journals and papers prior to the application of the framework.

Research Question 3: How can one design, implement and evaluate a generic framework for context-aware learning systems?

Once the systemic literature review was completed, the resulting analysis indicated a need for a system framework capable of providing technical details on the creation and layout of context-aware learning systems. This, then, set the stage for not only the creation of the framework, but also the application of the framework in two systems. The first system, which was designed primarily to prove the concept of the framework, was called the Knowledge Inference Training Terminal (KITT). KITT was created to provide real-time driver advice and training based on the inputs of numerous sensors both inside and outside a travelling automobile. The onboard system would then advise the driver of potential safety issues or hazards on the road ahead.
The second and much more in-depth context-aware learning system was the PORPOISE, or Pathogen Outbreak Prevention Instruction System. PORPOISE provided real-time information about the potential risks and contaminations of pathogens, given the context of the learner’s proximate medical environment (primarily long-term care facilities). Therefore, the PORPOISE system was designed as a live training tool to not only train, but also refresh already knowledgeable staff within long-term care facilities. A detailed description of both the framework and the two implementations can be found in Papers III, IV and V.

Research Question 4: What does the future hold for context-aware learning systems, and what are the possible issues?

With the culmination of the context-aware framework and the successful publishing of the paper, the next logical step was to determine which direction the framework and the technology may take. This led to an investigation of the potential drawbacks of stand-alone systems in terms of both storage capacity and computational capabilities. Therefore, the final research question represents a question asked in most academic papers: What is the future direction of the research? This directed the investigation toward the possibility of implementing cloud computing within the educational domain and, thus, context-aware learning. It also led to an exploration of the possible challenges in this field. The potential of this field is indeed great, with the tantalizing possibility of limitless computing power for processing more and more complex sensor data. However, there are also several drawbacks, which are covered in the final paper (paper VI) of this dissertation.
3 EARLY STAGES: INITIAL DEVELOPMENT

The path towards a framework for the development of context-aware learning systems started with the integration of two fields: context awareness and mobile adaptive learning.

Paper I described an iOS-based system designed to evaluate an adaptive approach for learning in mobile settings. The adaptive system considers learners’ learning styles and context information in order to determine the most appropriate learning format to present to each learner. Compared to the ultimate end goal of this research, the system itself was relatively simplistic; however, it proved very valuable in determining the potential and possible future applications of the technology.

The developed system taught a lesson (specifically created for this research) in astronomy, which was presented to the learner in a way that best suited both the learner’s context and his/her learning style. At first, the learner was presented with a standard Index of Learning Style (ILS) questionnaire (Felder, 1997) comprising a series of 44 questions created by Felder and Soloman (1988) to identify users’ learning styles based on the Felder-Silverman learning style model (FSLSM). These questions yielded a score representing a learner’s preference for one of the eight learning styles.

Then, the adaptive engine assumed control, using the aforementioned ILS score and sensor data from a small number of built-in iOS device sensors to provide the course material in an appropriate format. There were three basic formats: audio, video and text-based. The format was initially presented based on a learner’s ILS score; however, the presented learning format would then switch depending on the presented context. The context comprised the user’s movement, the user’s location and the ambient light conditions.

Although quite simple in terms of actual context, the system was quite effective at adapting the lesson to a learner’s changing context. To determine the context, the system used the GPS, movement and light sensors already present within the iOS device. The adaptive engine merged the values of the ILS with the preferences provided by the content to display the optimal lesson format to the learner.

3.1 EVALUATION

Although a single prototype of the device was successfully created, due to the necessity to test numerous learners, an alternative was needed to obtain and distribute 45 mobile devices. It was decided to utilize the iOS simulator on the OSX platform in order to evaluate the adaptive learning portion of the system based on the ILS questionnaire.
For testing, 45 senior high school students participated in the study as part of their non-graded computer science classwork. The students were tested before using the adaptive lesson system and then re-evaluated after using the system. The pre-test and post-test quizzes comprised the same set of 20 questions on basic astronomical principles, listed in a different order. All of the questions posed in the quizzes were covered as content within the course. After the pre-test, the students did not receive feedback on whether their answers were correct or not. Therefore, it is possible that the students’ post-test scores were equal to or even lower than their pre-test scores.

For the pre-test, the mean and median score for all 45 students participating in the study were both 13 (out of 20), with a standard deviation of 3. For the post-test, the mean score was 16 (out of 20) with a standard deviation of 3, and the median was 17 (out of 20). Since the average score obtained in the pre-test was 13 (out of 20) and the average score obtained in the post-test was 16 (out of 20), the average improvement was 3 marks out of 20. These three marks represent an improvement of approximately 23% of the pre-test quiz score.

The creation and evaluation of the prototype provided the means of answering the first research question (Table 1): How can one provide adaptivity in a mobile setting based on a learner’s particular learning style and contextual information? As demonstrated by the results, in this study, the learners’ particular learning styles were evaluated, and the created system successfully adapted to each learner by implementing the ILS questionnaire. The prototype was further able to adapt to the learners’ ever-changing contexts via the device’s onboard sensors.

Although this research found very positive results in terms of system viability, it also uncovered a lack of any accepted framework for developing context-aware learning systems. This finding set the stage for the remainder of the research presented within this dissertation. Specifically, the first step was to research all existing findings concerning context-aware learning systems. This research became the basis for Paper II: A Classification Framework for Context-aware Mobile Learning Systems.
As was uncovered during the research conducted in Paper I, there seemed to be a lack of any accepted means of identifying papers pertaining exclusively to (or, at the very least, involving) context-aware learning systems. A thorough understanding of the current state of research is necessary to understand and potentially advance any field. Thus, to adequately set the stage and the background for the research, a suitable overview of the current literature in the research field was required, and concise and repeatable method for correctly evaluating the current state of the research on context-aware learning systems was needed. The identified lack of any such method gave rise to the second research question (shown in Table 1): How can a framework be designed and evaluated to identify current trends in context-aware learning systems? This question was the genesis and reasoning behind the research and developments in Paper II. The resulting literature overview was used to determine what and how the research presented within this dissertation adds to the current body of knowledge on the subject matter.

With quite literally hundreds of thousands of potential papers to review on the subject of context-aware learning systems, it was necessary to devise a method that would allow for the suitable distillation of the current research field into a condensed form. This condensed form would ultimately serve as the foundation for a framework that could be used to describe the previous and current trends within the context-aware learning field.

It was therefore necessary to develop a suitable process for obtaining papers that would represent a comprehensive overview of the field. The following section describes the process that was ultimately developed and used to select papers suitable for the framework’s analysis.

4.1 PAPER REVIEW PROCESS

The paper review process had two stages: a journal selection process and a paper selection process. The journal selection process involved the selection of appropriate journals for the study. Similarly, the paper selection process involved the selection of appropriate papers from the selected journals using suitable criteria.

4.1.1 Journal selection process

To ensure the proper identification of trends in context-aware mobile learning, it was necessary to compile a list of journals suitable for performing a search for
papers pertaining to the subject matter. Since there is no specific research field that directly covers context-aware mobile learning in computing science, two parent research fields whose common elements adequately incorporated the targeted field were identified. These two research fields, which were selected because their merger encompasses the field of context-aware mobile learning, are context awareness and educational technology (Figure 2).

![Figure 2. Merger of Two Research Fields](image)

The inclusion of the context awareness field in the search allowed for the inclusion and incorporation of context-aware systems. The inclusion of the educational technology field in the search facilitated the incorporation of learning systems.

Although an overall definition of educational technology may be generally accepted or intrinsically known, the same cannot be said for context awareness. Therefore, in order to develop a proper understanding of context awareness, this study adopted the following definition of context-aware systems:

Context-aware systems are able to adapt their operations to the current context without explicit user intervention and thus aim at increasing usability and effectiveness by taking environmental context into account. Particularly when it comes to using mobile devices, it is desirable that programs and services react specifically to their current location, time and other environment attributes and adapt their behaviour according to the changing circumstances as context data may change rapidly. The needed context information may be retrieved in a variety of ways, such as applying sensors, network information, device status, browsing user profiles and using other sources. (Baldauf, Dustdar, & Rosenberg, 2007)

This definition by Baldauf et al. (2007) was selected because of its preciseness in defining a context-aware system. In addition, this definition is widely cited within the field, including in existing surveys on context-aware systems (Hong, Suh, & Kim, 2009; Perera, Zaslavsky, Christen, & Georgakopoulos, 2014).
After identifying context awareness and educational technology as the two main parent research fields, a list of journals from these fields in which papers relating to context-aware mobile learning systems could be found was assembled. Google Scholar was used to select the top 10 journals using the journals’ 2014 h5-index scores. The next step in the process was to find appropriate papers relevant to the subject matter of context-aware mobile learning systems in each of the journals.

4.1.2 Paper selection process: Three phases

With the journals selected, the next task was to search for all papers that pertained to the subject matter of context-aware mobile learning. The methodology implemented to search for such papers consisted of three phases, each of which narrowed the findings of the previous phase, with the final phase resulting in the papers described in this manuscript. The initial phase involved selecting the keywords for the search criteria and the preliminary search for potential papers. The secondary phase involved a manual check to remove erroneous and duplicate entries. The tertiary and final phase involved reading each of the identified papers’ abstracts and, when necessary, entire texts in order to determine each paper’s possible inclusion in the literature review.

4.1.3 Initial phase

The first part of the initial phase of the search involved the process of selecting keywords covering the two selected fields of educational technology and context awareness. This sub-section discusses and examines the search criteria used to select papers utilizing the keywords.

Two distinct groups of keywords

Journal websites generally include search functions that allow visitors to search their publications by specifying varying search parameters. Thus, since one may search for articles by entering an appropriate search query, a listing of suitable search query criteria was needed. Several methods for searching for journal articles, such as electronic databases and internet search engines, were available. However, despite the wide variety of web-based search engines, in order to ensure that no critical papers were missed, a decision was made to use specific journals’ own search engines in order.

With regard to date, the search incorporated all papers published from 2009 to 2015 (inclusive). A series of keywords to populate the paper search query was devised for each of the parent fields. These keywords (shown in Table 2) represent common words found in each of the two fields.
Table 2. Keyword Selection (Paper II)

<table>
<thead>
<tr>
<th>Educational Technology Keywords</th>
<th>Context Awareness Keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>School</td>
<td></td>
</tr>
<tr>
<td>Teach</td>
<td>Context</td>
</tr>
<tr>
<td>Education</td>
<td>Location</td>
</tr>
<tr>
<td>Learn</td>
<td>Mobile</td>
</tr>
<tr>
<td>Instruction</td>
<td>Pervasive</td>
</tr>
<tr>
<td>Training</td>
<td>Position</td>
</tr>
<tr>
<td>Curriculum</td>
<td>Sensors</td>
</tr>
<tr>
<td>Academic</td>
<td>Ubiquitous</td>
</tr>
<tr>
<td>Student</td>
<td></td>
</tr>
</tbody>
</table>

The rationale behind the keyword selection process was to identify a list of keywords that adequately and suitably represented the research field of context-aware mobile learning. Terms that fit both parental fields were included in only one of the keyword lists. This ensured that the elements of each keyword list were as unique as possible and prevented false positives during the search for papers.

**Searching criteria**

Once the keywords were selected, a search criterion incorporating the keywords was created to support the selection of suitable articles. Since both parent fields (i.e. education technology and context awareness) were required for an article to be relevant, it was decided that one or more keywords from each column must be present in the title or abstract of a paper for the paper to make it to the next phase of the selection process. In other words:

- **Given that** $X$ = a keyword from the Education Technology Keywords column in Table 2
- **Given that** $Y$ = a keyword from the Context Awareness Keywords column in Table 2
- **Given that** $n$ = a number from 1 to 9
- **Given that** $m$ = a number from 1 to 7

A suitable paper would have:

$$X_n \ AND \ Y_m \quad \text{(Eq1)}$$
within the title and/or the abstract. The method of applying (Eq1) to each journal’s search engine was typically unique, since each search engine constructed the search query in a different way. Thus, it was often necessary to re-write the database query directly at the URL level.

Applying (Eq1) to the keywords in Table 2 yielded searches that returned a positive hit whenever they detected any possible combination of a keyword from the educational technology field and a keyword from the context awareness field. Although there are 63 possible permutations, some examples of searches are:

- Learn AND Mobile
- Training AND Position
- School AND Sensors
- Academic AND Context

Following the first round of searches, the initial phase yielded a total of 2968 hits. These were not necessarily unique hits, as the searches often returned a paper twice: once when run against the title and once when run against the abstract. Therefore, papers for which both the title and the abstract contained relevant information and keywords were often listed twice. In addition, the search results included papers that, due to the limitations of some of the search engines used, contained information that satisfied (Eq1) within the paper body. These papers were deemed false positives or duplicate entries.

4.1.4 Second and tertiary phases

The secondary phase involved a manual review of all of the papers identified in the first phase in order to duplicate entries due to (Eq1) being satisfied by either the title and/or the abstract. Finally, the tertiary phase of the paper selection process was the most laborious and intensive of the three phases. The abstracts of all 2137 papers were read to determine each paper’s suitability for inclusion in the research study. Specifically, the papers’ abstracts (and, when necessary, the full texts) were manually reviewed to determine whether:

1. the paper described a context-aware system;
2. the paper described the occurrence of some type of automatic adaptivity based on context; or
3. the paper involved some type of learning.

For the first check, each paper was compared to the previously mentioned definition of a context-aware system to ensure that the definition applied. For the second check, the adaptivity of the paper’s system was reviewed. If there was no adaptivity based on context, then the paper was discarded. It must be noted that adaptivity had to be done automatically by the system; there could be no direct user intervention. In other words, the device itself had to inherently adapt to the context without the user being either aware of the adaptation or needed to assist or participate in the adaptation in any direct way. This second review point (i.e.
concerning automatic adaptivity) was included to satisfy the selected definition of a context-aware system, as described by Baldauf et al. (2007). Finally, for the third and final check, each paper was reviewed to ensure that it involved some type of learning. Learning could include formal learning, informal learning or training.

These three phases yielded 41 papers that not only satisfied the initial paper selection process, but also successfully passed all three checks for context-aware mobile learning.

### 4.2 CLASSIFICATION FRAMEWORK

A classification framework was required to provide a repeatable and standard method of reviewing the classifications and trends within the field. Therefore, a framework comprising three layers, each subdivided into classification categories, was developed (Table 3).

**Table 3. The classification framework (Paper II)**

<table>
<thead>
<tr>
<th>Layer</th>
<th>Device used</th>
<th>System infrastructure</th>
<th>Connection type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware/Architecture Layer</td>
<td>PDA</td>
<td>Standalone</td>
<td>Wireless (Wi-Fi)</td>
</tr>
<tr>
<td></td>
<td>Smartphone/mobile</td>
<td>Server based</td>
<td>Mobile/Cellular</td>
</tr>
<tr>
<td></td>
<td>Tablet</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Handheld</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Context Determination Layer</td>
<td><strong>Type of context</strong></td>
<td><strong>Type of sensor</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ambient: temp, humidity</td>
<td>Accelerometer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Location</td>
<td>Global positioning system (GPS)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movement of device</td>
<td>Radio frequency identification (RFID)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Movement of user</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Temporal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Evaluation Layer</td>
<td><strong>System evaluation</strong></td>
<td><strong>Duration of testing</strong></td>
<td><strong>Participant</strong></td>
</tr>
<tr>
<td></td>
<td>Survey / Questionnaire</td>
<td>Full day</td>
<td>Number of participants</td>
</tr>
<tr>
<td></td>
<td>Pre-post tests</td>
<td>Full week</td>
<td>Participant age</td>
</tr>
<tr>
<td></td>
<td>Interview</td>
<td>Part day</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Other</td>
<td>Part week</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Part month</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Subject matter</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Learning type</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discipline</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

There were several reasons for including each layer of the framework. Since the purpose of the framework was to classify and evaluate trends related to the context awareness field of context-aware mobile learning, the means by which the system adapted to context needed to be investigated. This need led to the context determination layer, which reviewed both the type of context and the sensor used to determine the context.
Another key part of context determination is the system hardware, which was documented by the hardware architecture layer. This layer reviewed the type of device being used, the system infrastructure and the connection types used by the various systems.

Lastly, since the field of context-aware mobile learning includes the field of educational technology, an evaluation layer was incorporated into the framework. This layer addressed the educational aspects of the various systems being reviewed. Specific points of review included the means by which the system was evaluated during the study, the duration of the testing, the participants’ age and number and the subject matter being taught.

This section has discussed the framework presented in Table 2 and the above-described layers, which were designed to classify and summarize trends within the context-aware mobile learning field. The following sub-sections will present an overview of the findings of the framework’s application to the field of context-aware mobile learning and some of the potential implications of this application.

4.2.1 Literature framework findings

The contents of Paper II present the framework and its application to the field of context-aware mobile learning. Although the specific details are too involved to present here and can be read in full within the paper, this manuscript will provide an overview of the major findings and the direction of the research field.

![Type of Device Being Used (Number of Papers)](image)

**Figure 3. Type of Device Being Used (Number of Papers)**

As can be seen in Figure 3 the predominant type of mobile device used was the PDA. Although the popularity of the PDA was certainly in decline prior to the 2009 start date of the literature review, the reason for its continued use may be twofold: cost and functionality. Although one may expect smartphones to be the dominant
devices uncovered by the study, the PDA certainly held its own in terms of both cost per unit (as PDAs are considerably less expensive than smartphones) and functionality. PDAs also have greater expandability (via the rather ubiquitous PCMCIA slot) than smartphones. As can be seen in Figure 4 the vast majority of papers used location as the context to which their systems adapted.

![Figure 4. Context Type (Number of Papers)](image)

Location contexts were determined using two types of sensors: GPS and RFID. GPS was used in 16 of the 41 papers, and RFID was used in 23. Since RFID transceivers are not included in either the smartphone or the PDA hardware suite, RFID hardware had to be added to the used devices. The expandability of the PDA via the PCMCIA slot was, therefore, invaluable in providing a suitable interface for the end user.

![Figure 5. Type of Sensor (Number of Papers)](image)
Overall, the various systems were well adopted by learners, with a special focus on the K–12 age group for formal learning (Figure 6).

![Figure 6. Subject Matter Learning Type (Number of Papers)](image)

However, the future of the research and the direction it may take were difficult to determine using current trends. It was evident that the era of the PDA had come to an end; however, was the smartphone able to “pick up the slack”, as it were, and continue on? The biggest hurdle seemed to be the reliance on and relative simplicity of using and focusing on spatial location (Figure 4), since spatial location offers only a limited amount of context. Therefore, it seemed that the field required more elaborate sensors and, thus, more elaborate hardware to adequately address context-aware mobile learning. This need for more information served as the foundation for the development of the hardware framework discussed in the next section.
5 FRAMEWORK DEVELOPMENT

Once the current status of the field was determined, the lack of a standardized framework for the development of context-aware learning systems became apparent. In much of the current literature on the subject of context-aware learning, systems are designed individually, as one-off projects coded from scratch for the sole purpose or task at hand. These systems tend to have commonalities in terms of an overarching concept; however, they did not appear to stem from a single generalized framework.

Therefore, it was decided that, in order to support the advancement of the field, a framework capable of providing both direction and methods for creating context-aware learning systems would need to be developed. This objective became the driving force being the third research question (shown in Table 1): How can one design, implement and evaluate a generic framework for context-aware learning systems? This question, in turn, became the core of this dissertation and is discussed in the next two chapters of this manuscript.

This framework addresses the relationship between sensor data and the learning system by explicitly defining the actions and rules governing context adaptation. The following sections describe the framework and the prototype system created using the framework in further detail.

5.1 EARLY STAGES OF THE FRAMEWORK

The generation of any kind of framework is bound to involve several iterations along the path to completion. The creation of the framework for developing context-aware learning systems was no different. The original idea behind the framework (Paper III) was to develop a generic platform, rather than a framework in the conventional sense. The resulting platform comprised several aspects that were eventually utilized in the final version. The workflow of these aspects is shown in Figure 7.
Initially, the goal was to create a generic platform/system that could be modified to produce as many varied context-aware learning systems as desired. The resulting platform was expected to be a ready-made software tool that would allow developers to select sensors via a drop-down menu (or the like) and enter rules and actions to create a ready-made context-aware system in hours instead of months. However, in the attempt to realize this ambition, several technical issues were uncovered. Some of these are addressed in the following.

The first issue was that of language: In which programming language should the code be written? The chosen language needed to support the most devices. Additionally, it was necessary to consider the variants of the chosen programming language and determine which was best. The question of programming language longevity that seemed to be unanswerable, since many programming languages have come and gone, and few have stood the test of time against the persistent advancement of technology.

There was also, of course, the question of platform support. Selecting a particular language/platform (e.g. iOS vs Android) would alienate a large percentage of potential global devices from the beginning of the development. Additionally, the need to manually code all available sensors for the generic system made this an insurmountable and infeasible task.

Finally, there were several other relevant factors that influenced the creation of the desired generic platform. For example, it was necessary to consider which data structures would work best for any given number of sensors and systems.
However, compared to the two considerations listed above, these challenges were relatively trivial.

Given these many questions and the lack of a clear answer for any of them, it was decided to abandon the platform idea. However, the basic layout of the generic platform laid the groundwork for the creation and development of the final framework.

5.2 REBIRTH: FRAMEWORK CREATION AND INTERNAL STRUCTURE

After carefully considering and reformulating the direction of the research from a generic platform to a more all-encompassing framework, everything seemed to finally be in place. As mentioned, several aspects of the original platform were kept; however, even these were reformulated to ensure maximum possible compatibility with any and all hardware and software systems. In sum, the framework was written in the hopes that the tenets would still be valid and the framework could be used with any possible future system.

In terms of layout, the structure changed from a platform to a framework. The final framework comprises three main sections: System Setup, Decision Mechanism and Sensor Input and User Output. Furthermore, each section comprises several components (Figure 8).

![Diagram](image)

Figure 8. Context-Aware Learning System Framework (Paper IV)
These components are discussed in detail in the following few subsections. This division by section allows for both a high-level, overall view of the setup and a detailed understanding of how to split up any context-aware system in terms of both functionality and of programming.

5.2.1 System setup—Component I: Sensor data acquisition

This component is responsible for determining which sensors are available to the system and which attributes can be measured by the system. As such, this component is further divided into two sub-parts:

- Part A – Sensor determination
- Part B – Attribute determination

Part A — Sensor determination

This part represents an inventory of each sensor present or available to the system. Each sensor is given a name (\(S_1, S_2, \ldots, S_n\)), which is then utilized in component II of the framework.

Part B — Attribute Determination

The attributes to which the system is adapting must also be identified and stated. These attributes (\(A_1, A_2, \ldots, A_n\)) are key to the framework, as they directly represent the contexts of the user and the system. They are, therefore, the means by which the framework determines how to adapt for each user.

Once both of the available sensors are collected and listed and the desired attributes are understood, it is necessary to understand the relation between the two. Component II of the framework ensures an adequate correlation between the desired attribute and the available sensors.

5.2.2 System setup—Component II: Attribute assignment verification

In order to properly lay the foundation for the context-aware system, it is necessary to define which sensor represents which desired context data and, thus, to ensure that the attributes (\(A_1, A_2, \ldots, A_n\)) are assigned to corresponding sensors (\(S_1, S_2, \ldots, S_n\)). Therefore, it is necessary to ensure that:

\[
\text{every } A_x \text{ correctly correlates to } S_x
\]  

alternatively:
Component I Part B corresponds to Component I Part A \((Ax \rightarrow Sx)\) \((2)\)

As an example of this, can an attribute \(A_1\), representing current air temperature, be obtained/measured with a particular \(Sn\)? If \(Sn\) is a thermistor, then the answer is yes. Then, \(A_1 \rightarrow S_i\), or temperature \(\rightarrow\) thermistor.

So, the rules governing the system (which are found in Component III) can be written; if not, then additional sensors must be added to Component I in order to fulfill the needs of the system. This verification is of key importance, since one may desire a particular attribute but lack the sensor necessary to provide information about the attribute.

It is important to note that attributes may, in fact, have multiple complex relationships with several sensors:

\[Ax \rightarrow S_x + S_y + S_z\] \((3)\)

Compound attributes, such as the one described above, are possible; however, they are not recommended, since this type of processing is best left to the realm of the inference engine rules described in the next section.

5.2.3 Decision mechanism—Component III: Inference engine rules

Once the sensors and attributes have been correlated, the logic or programming of the inference engine can begin. In component III, the inference engine rules, the various rules and logics governing the system’s inference engine are described.

Once again, there are two aspects to this component. Part A of component III covers the possible need to translate the raw sensor data (RSD) obtained from sensor (S) into specific desired attribute data (DAD). Part B of the component describes the specific inference engine rules developed based on the DAD and the querying threshold (QT).

Part A—Correlate RSD to DAD

Frequently, analog sensors do not provide the RSD for an attribute in a useful manner. It may be necessary for the RSD to be adjusted or correlated in order to reflect the DAD. In such cases, the DAD become a function of the RSD.

\[DAD_x = f(RSD_x)\] \((4)\)

For instance, in the aforementioned temperature example, the sensor may provide temperature as a resistance rather than a straightforward temperature reading. With this in mind, and given the example of a standard thermistor, the possible correlation function may be:
e.g. Obtaining the temperature (DAD₁) from thermistor value (RSD₁) where A, B, C = Steinhart-Hart coefficients

\[
DAD₁ = f (RSD₁) = \frac{1}{(A + B\ln(RSD₁)) + C(\ln(RSD₁))^3}
\] (5)

However, the conversion from the RSD to the DAD may also be as simple as a scale conversion: for example, converting from °F (RSD₁) to °C (DAD₁):

\[
DAD₁ = f (RSD₁) = \frac{(RSD₁ - 32)}{1.8}
\] (6)

Regardless of the complexity of the relationship between the DAD and the RSD, and even if they have a simple one-to-one correlation, they are governed by (Eq. 4). Normally, the DAD value implies human-readable code, which can greatly facilitate and speed up the writing of the rules governing the inference engine, presented in Part B.

**Part B—Inference engine: RULES (R₁, R₂, …, Rₙ)**

The logic and rules that govern the inference engine are central to context-aware learning systems. At the heart of the inference engine resides the decision mechanism, where logical statements are evaluated to adjust to the context of the learner. The rules present within the inference engine can vary greatly in terms of detail and complexity.

As is the case with any expert system, the accuracy and overall quality of the logics and rules comprising the inference engine and the knowledge base define the overall capabilities of the system. However, simple or not, the logics employed have intrinsically the same basic format, which is based on the threshold for activating rule Tx to perform activity Actx:

\[
Rx = \text{If } DADx \text{ (operator) } Ax \text{ (Tx) then } ACTx
\] (7)

where:

- Rx is Rule X
- Tx is Threshold X
- Ax is Attribute X
- DADx is Desired Attribute Data X
- ACTx is the Activity X to be performed

This can be illustrated using the above temperature example:
If a test needs to be made to determine whether the water is boiling, if DAD₁ is already set to °C, one can set a threshold T₁ of 100. The resulting rule R₁ would have the following format:

\[
R₁ = \text{If } DAD₁ > T₁ \text{ Then } ACTx
\]

The activity ACT₁ to be performed could be anything. This activity will be the focus of the fifth component (component V) of the framework.
As noted before, compound rules are also possible. These are, in fact, directly related to the complexity of the context-aware system. Here, a rule may comprise several smaller rules that, when verified, produce a compound rule:

\[
Rₓ = \text{Check (Rw) & Check (Ry) & Check (Rz)}
\]

Compound rules allow the system designer to add more advanced levels of complexity to the framework as needed or required by the task or system being addressed.

The rules Rx described here are, in effect, mixtures of the knowledge base rules and the inference engine’s ability to process and interpret these rules. Simple examples have been given for the purpose of explanation; however, it must be noted that the expert system’s complexity is as boundless as the designers wish and can be adjusted utilizing either forward or backward chaining.

5.2.4 Sensor Input and User Output—Component IV: Data query

A sensor’s data is only as relevant as the time it was last obtained; therefore, the frequency by which a sensor is queried by the system is of vital importance. The system’s ability to determine the value of an attribute (Ax) is based on the frequency of the corresponding query. In terms of the framework, querying sensors and checking the resulting values against rules is of key importance, since any failure to query within an appropriate timeframe may have disastrous consequences.

Given, for example, the previously mentioned boiling water scenario, querying the temperature sensor and associated rule every 10 minutes is likely to result in the system being blissfully unaware of the water growing to a boil and causing possible damage or harm. On the other side of the coin, checking for the presence of water droplets on a rain sensor every millisecond may return a value of a “clear day”, when, in reality, the system simply does not detect a new drop within each fraction of a second.

Given these considerations, the rate at which the sensors and rules from component III are checked must be stated both in order to understand the
functionality of the framework and to properly code the system. The query rate is termed the Query Frequency (QF$_1$, QF$_2$, ..., QF$_n$). Thus, each rule within the system Rx has an assigned query frequency of QFx:

Check Rx at an interval of QFx (9)

For example, should the temperature of the system (R$_1$) need to be measured every 5 seconds, then, per (Eq. 9), R$_1$ would have a query frequency (QF$_1$) of 5 seconds, or R$_1$ → QF$_1$ = 5 seconds.

As has been shown, the rule itself can be queried; however, if the sensor data are not queried at the same (or a more frequent) rate, the results are potentially meaningless. As a general outline, if sensor flags may be implemented when a sensor has a specific query frequency, then the associated rule may only check whether a flag has been triggered (i.e. is true) at another frequency. So, in this type of scenario, the resulting system would need to consider multiple query frequencies for a particular Rx: one for the Rx and one for the Ax.

5.2.5 Decision mechanism—Component V: Actions based on rules

The inference engine rules found in the decision mechanism described in component IV are only useful if something is done as a consequence of the rules. Thus, a rule (Rx) should trigger an action (ACT$_x$), which must be specified. The specified action can take any form, such as a trigger for a rule or a series of other actions.

ACT$_x$ → ACT$_y$ + ACT$_z$ (10)

or

ACT$_x$ → Output to User (play video, show message, play audio) (11)

As should be expected, this section of the framework is quite generic, since the types of actions possible are limited only to the imagination. Generally speaking, this part of the framework deals with how to describe or define the details of a particular action. Once again, to use the simple example of water boiling, the action could be to play an audio file or sound an alarm notifying the learner that the water has begun to boil. In an example geared more towards learning, it could be that once a particular location is reached, the contents of a classroom file are displayed on a predefined interface.

Yet, regardless of its format, every action takes place as a direct result of a rule, which, itself, can be defined as one or more attributes designed to reach one or more desired or pre-determined conditions or values. As alluded to in the example, actions often require some type of interaction with the end user; therefore, the final step in the framework addresses the output to the user, or the user output.
5.2.6 Sensor Input and User Output—Component VI: User output

This section describes the final output to the learner or end user. Once again, this section provides a rather generic overview of a part of the system that can vary in countless ways; thus, the section focuses on the specific details of the system’s output to the learner. Given the myriad of possible systems, this component will usually be hardware or system-specific, and methods of user delivery will differ from system to system.

This section however, is also not limited to output and may also include user input; however, given the system’s environmentally driven adaptive behaviour, any user input would have no impact on the adaptive mechanism. Any such input would, therefore, be limited to altering the way in which information is relayed to the end user (e.g. a simple push of a button or a more detailed heads up display). Therefore, while the varying specific aspects of user interface interactions are beyond the scope of this framework, it is necessary to include them nonetheless.

This framework proposes an underlying foundation on which to base context-aware learning systems. Its overall generality allows for specific customizations that intrinsically support guided improvements to the interface to maximize both user satisfaction and learning effectiveness.

Once the framework was established, the next step was to implement the framework in two different scenarios in order to test both its viability and end user acceptance.
6 IMPLEMENTATION OF THE FRAMEWORK

Once the framework was outlined, its applicable functionality needed to be proven through applications to real-world scenarios. Thus, two learning scenarios were developed and implemented using the framework: the Knowledge Inference Training Terminal (KITT) and the Pathogen Outbreak Prevention Instruction System (PORPOISE). This section discusses the reasonings behind the creation of these systems and their evaluation.

6.1 IMPLEMENTATION: KNOWLEDGE INFERENCE TRAINING TERMINAL

This implementation was first performed primarily to verify that the framework was, indeed, a viable means for designing and implementing context-aware learning systems. This first implementation of the framework—the KITT system—was designed as a driving training system. The system comprises a multitude of sensors and detectors processed through a mobile processing unit that provides training to the driver/learner. The training described in KITT takes the form of insights and advice concerning safe driving practices that are provided to the driver of an automobile given the contexts of both the vehicle in the outside environment and the driver within the vehicle’s cabin (Figure 9). A small-scale (16 participants) evaluation of the system was performed, during which participating drivers utilized the training tool and evaluated their experiences of the system.

Figure 9. Functionality of the Knowledge Inference Training Terminal
In order to adequately describe the KITT system, an overview of the hardware is provided, including information on the workings of each of the components of the framework.

6.1.1 KITT: Hardware overview

At the heart of any context-aware system is the decision mechanism, which contains the inference engine. To ensure that the inference engine was capable of properly querying the varying sensors and processing the information quickly, a robust central processing unit was needed. Raspberry Pi 2 (RPi2) was selected to achieve this objective because of its impressive quad-core A7 processors, audio output (via speaker) and general-purpose inputs and outputs (GPIO).

The GPIO pins were used primarily for the myriad of connected sensors and the thin-film-transistor liquid-crystal display. The system was then packaged in a portable shock-resistant enclosure (Figure 10).

![Figure 10. Knowledge Inference Training Terminal Enclosure](image)

The operating system was a modified Raspbian kernel specifically designed and constructed to comply with the Knowledge Inference Training Terminal.

KITT—Component I: Sensor data acquisition

This section lists and describes the available sensors and the attributes measured by the system (Table 4).
Table 4. KITT - Component I (Paper IV)

<table>
<thead>
<tr>
<th>Sensor Determination</th>
<th>Attribute Determination</th>
</tr>
</thead>
<tbody>
<tr>
<td>S₁  Photo-resistor</td>
<td>A₁  Day or night: ambient cabin light</td>
</tr>
<tr>
<td>S₂  High sensitivity microphone module</td>
<td>A₂  Noise level in cabin</td>
</tr>
<tr>
<td>S₃  CO gas sensor module</td>
<td>A₃  Cabin carbon monoxide gas levels</td>
</tr>
<tr>
<td>S₄  Magnetic sensor</td>
<td>A₄  Magnetic field</td>
</tr>
<tr>
<td>S₅  3-axis accelerometer</td>
<td>A₅  G-forces on passengers</td>
</tr>
<tr>
<td>S₆  Relative temperature sensor</td>
<td>A₆  Cabin temperature level</td>
</tr>
<tr>
<td>S₇  Temperature probe</td>
<td>A₇  Outside temperature level</td>
</tr>
<tr>
<td>S₈  Relative humidity sensor</td>
<td>A₈  Cabin humidity sensor</td>
</tr>
<tr>
<td>S₉  Pyroelectric infrared motion sensor</td>
<td>A₉  Cabin motion detection</td>
</tr>
<tr>
<td>S₁₀ Smoke/combustible gas sensor</td>
<td>A₁₀ Cabin smoke levels</td>
</tr>
<tr>
<td>S₁₁ Ethanol sensor</td>
<td>A₁₁ Cabin alcohol air levels</td>
</tr>
<tr>
<td>S₁₂ Water detection Sensor</td>
<td>A₁₂ Outside rain detection</td>
</tr>
<tr>
<td>S₁₃ Global positioning sensor (GPS)</td>
<td>A₁₃ Velocity</td>
</tr>
<tr>
<td>S₁₄ Barometric pressure sensor</td>
<td>A₁₄ Altitude</td>
</tr>
<tr>
<td></td>
<td>A₁₅ Location (longitude/latitude)</td>
</tr>
<tr>
<td></td>
<td>A₁₆ Outside barometric pressure</td>
</tr>
</tbody>
</table>

Given this list of available sensors and attributes to be measured and evaluated, the next step was to ensure that the attributes could, in fact, be measured using the sensors present.

**KITT – Component II: Attribute assignment verification**

As mentioned in component I, there is not always a simple one-to-one correlation between sensors and attributes. Even in this implementation, in which such one-to-one correlations are indeed present, it is important to note that several attributes are provided by the same sensor, as seen in Table 5.

Table 5. KITT - Component II

<table>
<thead>
<tr>
<th>Attribute → Sensor</th>
<th>Attribute → Sensor</th>
<th>Attribute → Sensor</th>
<th>Attribute → Sensor</th>
</tr>
</thead>
<tbody>
<tr>
<td>A₁</td>
<td>S₁</td>
<td>A₆</td>
<td>S₅</td>
</tr>
<tr>
<td>A₂</td>
<td>S₂</td>
<td>A₆</td>
<td>S₅</td>
</tr>
<tr>
<td>A₃</td>
<td>S₃</td>
<td>A₁₁</td>
<td>S₇</td>
</tr>
<tr>
<td>A₄</td>
<td>S₄</td>
<td>A₁₂</td>
<td>S₈</td>
</tr>
</tbody>
</table>
In order to give an example of what Table 5 means in terms of the system, let us review attribute A1 (Day or night: Ambient cabin light). A1 is obtained from sensor S1, a photo resistor. As mentioned, the attribute-to-sensor relationship is not always one-to-one. For example, three of the attributes (A13 through A15) are all obtained from the GPS sensor: S13. The effect of the varying types of cardinality is that the query frequency (discussed later in this section) must vary between sensor and attribute. This is not an obstacle to overcome, but, rather, a condition to note when implementing (programming) the system. Thus, upon designing this or any such system, it is often helpful to think about the attribute Ax one wishes to measure first, and then to determine which sensor Sx would best fit the data needs. However, this is not always possible, since attributes often need to be calculated based on the availability of sensors.

**KITT—Component III: Inference engine rules**

Once the relationships between the sensors and the attributes have been determined, it is necessary to address the correlation between the RSD and the DAD. Table 6 shows the RSD and the DAD values and illustrates how the two are related.

<table>
<thead>
<tr>
<th>Sensor</th>
<th>RSD</th>
<th>DAD</th>
</tr>
</thead>
<tbody>
<tr>
<td>S₁</td>
<td>0-3.3V DC</td>
<td>Lux = (2500/V - 500)/3.3</td>
</tr>
<tr>
<td>S₂</td>
<td>Digital 1-0</td>
<td>1 → Sound, 0 → No Sound (manual threshold)</td>
</tr>
<tr>
<td>S₃</td>
<td>0-3.3V DC</td>
<td>f(V) → approx. V / 0.0055 = ppm CO</td>
</tr>
<tr>
<td>S₄</td>
<td>0-3.3V DC</td>
<td>f(V) → approx. V / 0.0013 = Gauss</td>
</tr>
<tr>
<td>S₅</td>
<td>0-3.3V DC</td>
<td>f(V) → approx. V / 67.584 = Angular Acceleration</td>
</tr>
<tr>
<td>S₆</td>
<td>Digital °C</td>
<td>No Conversion Needed</td>
</tr>
<tr>
<td>S₇</td>
<td>Digital °C</td>
<td>No Conversion Needed</td>
</tr>
<tr>
<td>S₈</td>
<td>Digital % Humidity</td>
<td>No Conversion Needed</td>
</tr>
<tr>
<td>S₉</td>
<td>Digital 1-0</td>
<td>1 → Motion, 0 → No Motion</td>
</tr>
<tr>
<td>S₁₀</td>
<td>Digital</td>
<td>Manually Set Threshold</td>
</tr>
<tr>
<td>S₁₁</td>
<td>Digital 1-0</td>
<td>1 → Water on Sensor, 0 → No Water</td>
</tr>
<tr>
<td>S₁₂</td>
<td>Lat/Long Decimal Degree</td>
<td>No Conversion Needed</td>
</tr>
<tr>
<td>S₁₃</td>
<td>Velocity Km/h</td>
<td>No Conversion Needed</td>
</tr>
<tr>
<td>S₁₄</td>
<td>Altitude</td>
<td>No Conversion Needed</td>
</tr>
<tr>
<td>S₁₅</td>
<td>Digital KPa</td>
<td>No Conversion Needed</td>
</tr>
</tbody>
</table>
The information shown in Table 6 has been simplified to aid the reader in understanding the concepts rather than the specifics of the system. In the code itself, the displayed relationships are much more complex. For example, the various gas sensors (S₃, S₁₀ and S₁₁) require considerably complex code in order to “zero them out”, or calibrate them to ambient values, as well as detailed code to correctly determine the measured atmospheric quantities. On the other hand, it may not always be necessary to translate from RSD to DAD. This is often the case with digital outputs on sensor modules, since the outputs of such modules are typically already translated by the sensor package (via onboard sensor hardware) into a human-readable and usable form.

Now that the sensors and their associated attributes have been identified and presented in a usable format, the next step in the framework is to implement the rules of the inference engine.

As mentioned, the rules governing the decision mechanism are at the core of the overall system. Once again, for the sake of simplicity, only a few selected rules are shown:

R₁ → If DAD₁ < 500 then ACT₁ (i.e. Is the ambient light less than that of an overcast day?)
R₂ → If DAD₂ == 1 then ACT₂ (i.e. Is there sound in the cabin?)
R₃ → If DAD₃ > 70 then ACT₃ (i.e. Are CO levels greater than 70 ppm?)
R₄ → If DAD₄ > 2 mG then ACT₄ (i.e. Are EMF fields greater than 2 mG?)
R₅ → If DAD₅ > 0.005 then ACT₅ (i.e. Is the vehicle experiencing a bumpy ride?)

Once again, rules are not always simple, and several factors often need to be considered. As previously described in the framework definition, it may be necessary to develop compound rules, as is the case for a specific example from the KITT:

R₉ → If ((DAD₆ > 26) & (DAD₁ < 500) & (S₂ == 0)) Then ACT₉

In the above example, in order to implement ACT₉, rule 9 (R₉) must check several variables. Specifically, rule nine checks that the temperature in the cabin is above 26°C, that it is evening/night (with a lux value of less than 500) and that there is no sound present in the cabin of the vehicle. If all of these conditions are true, then ACT₉ (defined later), which warns the driver of a higher risk of falling asleep and provides tips and tricks for remaining alert, is implemented. A similar rule utilizes the carbon monoxide sensor to ensure that the cabin’s air quality is conducive to remaining alert. Therefore, even rules can have multiple variants, further adding to the system’s complexity. Thus, generally speaking, the framework’s rule list provides insight into the system’s complexity. However, the rules themselves must
be checked (queried) in order to be used. Thus, it is necessary to address how often the rules are checked and what actions they may produce.

**KITT—Component IV: Data query**

The rate at which a rule is checked and evaluated is the key aspect to the fourth component of the frame-work. One of the most common checks is to test whether the condition or variable is true: a simple binary check. Yet, as the old adage says, even with a simple check, timing is everything. For example, there is little to be gained from checking the ambient light every few milliseconds. In such a situation, a check every five minutes is not only much more appropriate, but also better able to reflect humans’ perception of the world. There are, of course, instances outside of the KITT in which millisecond querying is needed; however, this is not the case for this specific example.

Therefore, in order to ensure that rules (and sensors) are queried at appropriate times, it is necessary to know the query frequency for each rule (and sensor, where needed). Thankfully, since RPi2 is capable of spawning threads and interrupts, developing a query function with varying times is very easy to implement via timeouts, bounce times and threading. Table 7 shows some of the query frequencies employed in the KITT.

<table>
<thead>
<tr>
<th>Rule</th>
<th>Query Frequency (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>R₁</td>
<td>QF₁ = 300</td>
</tr>
<tr>
<td>R₂</td>
<td>QF₂ = 300</td>
</tr>
<tr>
<td>R₃</td>
<td>QF₃ = 10</td>
</tr>
<tr>
<td>R₄</td>
<td>QF₄ = 30</td>
</tr>
<tr>
<td>R₅</td>
<td>QF₅ = 5</td>
</tr>
<tr>
<td>R₆</td>
<td>QF₆ = 10</td>
</tr>
<tr>
<td>R₇</td>
<td>QF₇ = 10</td>
</tr>
<tr>
<td>R₈</td>
<td>QF₈ = 60</td>
</tr>
<tr>
<td>R₉</td>
<td>QF₉ = 300</td>
</tr>
<tr>
<td>R₁₀</td>
<td>QF₁₀ = 300</td>
</tr>
</tbody>
</table>

For complex rules, as previously discussed in R₉, the query frequency considers two components: one based on how often the rule itself is queried and one based on is how often its components/sensors are queried.

Once the rules have been written and the query times have been determined, the next step is to determine the next course of action when rule’s conditions are met. In other words, what is the required action? The following section covers the
implementation of the fifth component of the framework’s implementation: actions based on rules.

**KITT — Component V: Actions based on rules**

This section defines the actions that take place when a particular rule check proves to be true. Like rules, actions can have a simple bijective functions or one-to-many correlations. They usually include things like warning the driver of perilous driving conditions or potential hazards or offering advice for avoiding potentially hazardous situations.

Some other actions (ACTx) are as follows:

- Warn the user/driver of high temperatures outside and describe the condition of heat stroke.
- Warn the user/driver of the perils of high-speed driving on wet roads and explain hydroplaning.
- Warn the user/driver of low temperatures outside and describe the perils of black ice.
- Warn the user/driver of what possible types of wildlife may be on the road, given the location, time and date.
- Warn the user/driver of extremely low outside temperatures and the dangers of frostbite.

The above actions are some of the actions taken by the system. However, the final step of any action is to trigger some type of user output. Such outputs are described in the next section.

**KITT — Component VI: User output**

This is the most generic of the various components of the framework, as it inherently differs from system to system. In the case of the KITT, the output cannot interfere with the learner’s safe operation of the motor vehicle. The means that outputs were selected to adhere to not only local motor vehicle regulations, but also safe driving practices. Therefore, user outputs were primarily delivered through a text-to-speech synthesizer that relayed information to the driver, as well as a small 3.5” (90 mm) display, which was used only as a means to get the driver’s attention (by flashing) when immediate hazards were present (e.g. low oxygen or high carbon dioxide or carbon monoxide present within the cabin of the vehicle).

With this description of user output completed, the next sub-section briefly discusses the final evaluation of the KITT.
KITT—Evaluation

Once created, the KITT was tested for both functionality and user satisfaction. The inherent challenges of assessing the functionality of an adaptive system meant that a survey was one of the best methods for determining users' attitudes towards and acceptance of the context-aware learning system. For the purposes of this evaluation, 16 adults participated in the study. The participants were equally divided between males and females, aged 19 to 70+ years and possessed valid driver’s licenses.

Each participant was given the KITT, and asked to drive (in his or her own vehicle) around the local area. There was no time limit on the driving experience, and participants were asked to drive as long as they felt was necessary to properly evaluate the system. The average duration of the participants' testing was 20 to 25 minutes. Once a participant completed his or her drive, he or she was provided a written questionnaire. The questionnaire included a list of 10 statements with which the participant indicated his or her agreement on a standard Likert scale of 1 to 5, where 1 represented strong agreement and 5 indicated strong disagreement with the statement. The questions and the average response values are provided in Table 8.

Table 8. Driver Questionnaire and Results (Paper IV)

<table>
<thead>
<tr>
<th>Item</th>
<th>Ratings</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1. I enjoyed using this system</td>
<td>1.38</td>
<td></td>
</tr>
<tr>
<td>Q2. The system was unobtrusive</td>
<td>1.56</td>
<td></td>
</tr>
<tr>
<td>Q3. The system helped me as a driver</td>
<td>1.81</td>
<td></td>
</tr>
<tr>
<td>Q4. The system improved my driving capabilities</td>
<td>1.94</td>
<td></td>
</tr>
<tr>
<td>Q5. The system taught me something I did not previously know or consider</td>
<td>2.19</td>
<td></td>
</tr>
<tr>
<td>Q6. I think the system could use improvement</td>
<td>3.75</td>
<td>Reverse wording question: 3.75 means leaning toward disagree</td>
</tr>
<tr>
<td>Q7. I would like to see the system’s capabilities expanded</td>
<td>2.94</td>
<td>Reverse wording question: 2.94 means leaning toward agree (although it is close to the median)</td>
</tr>
<tr>
<td>Q8. This system was too complex</td>
<td>4.31</td>
<td>Reverse wording question: 4.31 means leaning toward disagree</td>
</tr>
<tr>
<td>Q9. I would use this system again</td>
<td>1.44</td>
<td></td>
</tr>
<tr>
<td>Q10. I would recommend this system to other drivers</td>
<td>1.5</td>
<td></td>
</tr>
</tbody>
</table>
As shown in Table 8, the overall responses to the questionnaire were very positive. The findings demonstrate the viability of not only the KITT driver training system, but also the framework as a whole.

Given this positive feedback on the functionality of the framework (from both the success of the KITT and the development of the system itself), it was time to implement a much more elaborate system: one to test the viability of the proposed framework in a more complex environment. The results of the design, creation and testing of this system are discussed in the following section.

6.2 IMPLEMENTATION: PATHOGEN OUTBREAK PREVENTION INSTRUCTION SYSTEM

With each passing day, mobile computing devices are becoming more and more ubiquitous. It should, therefore, come as no surprise that the integration between our everyday world and our computing world is becoming increasingly entrenched. This integration particularly affects the world of medicine, in which care is provided in a multitude of locations and situations. The union of computing and health care systems has produced two new areas: e-health and m-health. E-health is the electronic exchange of healthcare-related information across organizations (Deluca & Enmark, 2000), while m-health, a sub-component of e-health, is healthcare facilitated by the union of desktop and mobile healthcare and mobile wireless technology (Yu, Wu, Yu, & Xiao, 2006). In recent years, m-health has attracted worldwide attention and popularity with its low-cost and high-yield solutions (Akter & Ray, 2010).

Given these advancements, it is not surprising that the medical field is being inundated with new types of technologies, including context-aware systems and applications (Immanuel & Raj, 2015). Recent focus in context-aware m-health has been primarily on ICT, with particular attention paid to patient-centered care and monitoring (Broens, Van Halteren, Van Sinderen, & Wac, 2007; Pawar, van Beijnum, Mei, & Hermens, 2009). This type of care often involves direct patient monitoring, as in a recent study involving long-term care requirements for diabetic patients and the detection of abnormal blood glucose levels (Chang, Chiang, Wu, & Chang, 2016).

The idea of focusing on the context of the end user is by no means new to the computing field. As early as the mid-1990s, Schilit, Adams, and Want (1994) described a system that analysed and reacted to a person’s changing context and location. Similarly, the union between learning and mobile technology is also evolving, growing and developing (Astrasheedi & Capretz, 2013). This union, called mobile learning, or m-learning, focuses not on the learners or their technologies,
but, rather, on the interaction between them in order to advance learners’ knowledge (Sharples, Taylor, & Vavoula, 2010).

Both m-learning and context-aware research are attracting growing interest; yet, little research has been done to explore the training of healthcare professionals using context-aware m-health systems. Therefore, to help solve the aforementioned research gap between m-health and m-learning, it was decided to apply the context-aware framework to this field. The resulting device, which is described here, provides real-time, context-aware medical training.

The proposed system, called the Pathogen OutbReak PreventiOn Instruction SystEm (PORPOISE), provides real-time information about potential pathogen-related risks and contaminations given the context of the medical environment in which a learner is situated. The system was designed for use in long-term care facilities, with a particular focus on cleaning staff and medical professionals. As such, it can be used not only as a training tool, but as a daily operational tool for constantly reminding care staff about the risks and dangers of pathogen transmission.

**6.2.1 Medical concerns for seniors**

Long-term care facilities are medical facilities that typically house weak, elderly and terminal patients. In such a critical environment, the spread of any type of common pathogen can have deadly consequences.

The Western world is facing an increasing and aging population; therefore, the need for long-term care facilities is escalating. According to a study performed by the American Center for Disease Control (2014), in 2014, the US was home to approximately 15,600 nursing homes and 1.7 million residents or patients. This staggeringly large population suffers approximately 2 million infections per year (Montoya & Mody, 2011). Even if only one percent of all infections are lethal, this still amounts to 20,000 pathogenic deaths per annum. Therefore, it is essential that long-term care facilities implement adequate awareness training and cleaning techniques to ensure the simple health and wellbeing of their residents.

**6.2.2 Implementation of PORPOISE**

The PORPOISE system is a prototype training system designed to provide context-aware education to the cleaning staff of long-term care facilities. The system provides training by adapting to learners’ environmental and locational contexts. The training offers information about and suggests means of cleaning possible sources of infection in order to reduce the spread of potential pathogens.

The prototype was tested by a team of support managers of long-term care facilities. For the purposes of testing, the system was presented with various mock scenarios and a sample patient room layout. The
managers were given the system and asked to evaluate it from a professional standpoint in terms of both its performance and its potential educational benefit to staff and the industry as a whole.

There exists research within the medical field on the effects of reducing pathogen contamination via improved cleaning methods (Munoz-Price et al., 2012). Sadly, however, scant research has focused on the non-medical aspects of patient care in the context of pathogen transmission. Even less attention has been devoted to using context-aware systems to aid in training to reduce the spread of pathogens.

The following section describes the system architecture of the PORPOISE system. Specifically, it provides an overview of the system hardware and the evaluation of the system.

6.2.3 System architecture

Via a suite of sensors, the PORPOISE system is able to provide medical training staff with information on the spread of potential pathogens present within their proximate environment. Once the system has evaluated its context via the information from its sensors, it adapts automatically (without any user involvement) to its surroundings. Based on this adaptation, the system provides pertinent training information to the end user/learner via its user interface.

The system was designed around the context-aware framework and follows the same layout as that seen before: the system setup group, the decision mechanism group and the input and output group (Figure 11).
As in the previous adaptation of the framework, the system setup group is responsible for translating the sensor inputs into the values needed for the inference engine. The decision mechanism group comprises an inference engine and actions based on rules. It is responsible for providing information to the learner based on the context. Finally, the input and output group is responsible for the means by which the system not only deals with the sensor querying process, but also provides information to the learner. These three groups will be discussed further in the next sections.

**PORPOISE — System setup group**

The PORPOISE uses several types of sensors to determine the environmental context. These sensors are handled by the system setup group. As in the parent framework, the system setup group is responsible for translating and correlating various types of sensor data into a format that is understandable by the decision mechanism. The sensors the PORPOISE system uses to detect and analyze its context are as follows:

- Location sensors (will be addressed later via beacons)
- Temperature sensors
- Relative humidity sensors
- Ambient air pressure sensors
- Date sensors

The raw data values returned by each of these sensors need to be coded or translated into meaningful information in order to be utilized by the decision mechanism.

**PORPOISE — Decision mechanism**

The PORPOISE’s decision mechanism is key to the system’s adaptation and is composed of an inference engine and a rule database. Given particular sensor data checks/process rules, the inference engine trigger varying types of user output. The database is simply the location where the system stores the rules and actions that may be triggered by the inference engine.

The inference engine on the PORPOISE was programmed based on extensive interviews and consultations with long-term care industry experts, medical professionals, and the local Center for Disease Control (CDC). These sources enabled the system to offer accurate information on environmental data and best industry practices relating to pathogen growth, transmission and prevention.
The inference engine was programmed with rules to indicate the likelihood of a particular pathogen provided a given set of conditions. For the purposes of testing the system, rules were written for the top 10 pathogens present in most long-term care facilities in North America.

**PORPOISE—Input and output group**

As described in the framework overview, this group is responsible for how the system detects and interacts with its surroundings, including, most notably, the learner/user. The input aspect is a series of sensors located in the sensor bay enclosure (Figure 12). The output, or the means by which the PORPOISE system interacts with the learner, comprises several LCD and OLED screens and informational LEDs. These hardware aspects are discussed in further detail in the following section.

**PORPOISE—Hardware overview**

Much like the KITT, the PORPOISE is based on Raspberry Pi. However, the PORPOISE used a newer architecture and Raspberry Pi 3 (RPi3) to achieve superior computational speed. Outside of the user/learner-held device is the Wi-Fi locator beacon, which is responsible for determining the user’s relative location within a building. This component will be discussed in a further sub-section.

**PORPOISE—Main processing handheld unit**

The handheld unit is built on RPi3. This unit, along with the various sensor input bays and the LCD and OLED screens for user output, can be seen in Figure 12.
The main hardware difference between the KITT implementation and the PORPOISE implementation, other than the sensors themselves, is the user interface. Whereas the KITT utilized a text-to-speech synthesizer as the primary means to relay information to the user, the PORPOISE system uses a number of displays and lights. The status LED in Figure 12 shows, among other things, a warning status concerning the occupant of a room in a long-term care facility. The colour code displayed by the RGB LED is derived from industry standards to ensure staff/cleaners’ safety. For example, a green light means an all clear, while a violet (purple) light indicates that the resident of the room is potentially violent. However, the most interesting sensor system utilized in the PORPOISE is the Wi-Fi locator beacon system.

**Wi-Fi locator beacons**

Most medical facilities are built as large, concrete structures. This is also the case for most long-term care facilities. Within such structures, it is generally difficult to obtain reliable and accurate positioning. Therefore, since the PORPOISE system needs to be aware of a cleaner’s location within the building and even within specific rooms (bathroom, bedroom, etc.), another means of location determination was needed.
The solution was a Wi-Fi locator beacon built around a ESP-8266 chip. The ESP-8266 is a low-cost, low-power microcontroller that is able to act as, among other things, an SSID access point. These devices were set up strategically in various rooms, programmed to identify the rooms in which they were placed and positioned equidistant from any/all ingresses and exits.

Figure 13. PORPOISE Wi-Fi Location Beacon

In Figure 13, each of the two colours represents a differently named beacon (SSID). As the PORPOISE system enters a room, the PORPOISE’s onboard Wi-Fi detects the strongest SSID signal and determines its location accordingly. To ensure the PORPOISE’s accuracy when entering different rooms, the sensors need to be equidistant from all ingresses and exits. During testing, the Wi-Fi beacon system was able to detect the correct room within 30 cm of entering or exiting a room.

Since there are potentially hundreds of rooms in a long-term care facility, a suitable naming convention for the various SSIDs was required. Therefore, the following format, which could be expanded to any number of rooms or facilities, was adopted:

SITE-RoomNumber-Location
- SITE: The building site code.
- RoomNumber: The room number, often in the following format: RoomXXX
- Location: Differentiations among various ancillary rooms.

For example, the testing beacons were: WC01-Room101-Main and WC01-Room101-WR.

The onboard database on the PORPOISE system contained room information and patient/resident names and details, including, as mentioned before, the patient/resident’s threat level. With the description and layout of the system done, the next step was to evaluate PORPOISE’s ability to provide specific training on how to prevent the spread of potentially deadly pathogens.
6.2.4 PORPOISE—Evaluation

In order to evaluate the system beyond lab bench testing, a proper test environment was needed. However, the large range of environmental conditions required in order to trigger the various aspects of the decision mechanism posed a problem: How could these conditions be duplicated in a real-world scenario? Thankfully, research has already been conducted on this very issue. Kjeldskov et al. (2004) demonstrated that testing a context-aware mobile system under controlled settings (in comparison to real-world settings) does not decrease the ability of the evaluation to detect usability issues. Therefore, a suitable mock environment was set up so that the designers could demonstrate the functionality of the system by controlling the sensory inputs without modifying the results.

A total of 16 participants were involved in the study. Each of the participants was a manager who oversaw more than 500 healthcare cleaning professionals. The usage of the system was demonstrated to each participant, and the participants were then given the opportunity to test the device.

In order to ascertain the real-world viability and usability of the device, several questionnaires were reviewed. Ultimately, the System Usability Scale (SUS) (Brooke, 1996) was chosen as the most effective means of testing the system. The questions from the SUS were modified to better represent the PORPOISE system. Additionally, two questions (numbers 11 and 12) were added to directly assess the device’s training potential. The questions and the averages of the participants’ responses are shown in Table 9.
Table 9. PORPOISE Questionnaire Results (Paper V)

<table>
<thead>
<tr>
<th>Question</th>
<th>Average Value (0-4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. I think my staff and I would like to use this system frequently</td>
<td>3.3</td>
</tr>
<tr>
<td>2. I found the system unnecessarily complex</td>
<td>0.9</td>
</tr>
<tr>
<td>3. I thought the system was easy to use</td>
<td>3.4</td>
</tr>
<tr>
<td>4. I think that my staff and I would need the support of a technical</td>
<td>2.2</td>
</tr>
<tr>
<td>person to be able to use this system once it has been set up for my site</td>
<td></td>
</tr>
<tr>
<td>5. I found the various functions of this system were well integrated</td>
<td>3.4</td>
</tr>
<tr>
<td>6. I thought there was too much inconsistency in this system</td>
<td>0.8</td>
</tr>
<tr>
<td>7. I would imagine that most of my staff would learn to use this system</td>
<td>3.4</td>
</tr>
<tr>
<td>very quickly</td>
<td></td>
</tr>
<tr>
<td>8. I found the system very cumbersome to use</td>
<td>0.7</td>
</tr>
<tr>
<td>9. I felt very confident using the system, and that it provided useful</td>
<td>3.6</td>
</tr>
<tr>
<td>information</td>
<td></td>
</tr>
<tr>
<td>10. My staff and I will need to learn a lot of things before we could</td>
<td>1.5</td>
</tr>
<tr>
<td>get going with this system</td>
<td></td>
</tr>
<tr>
<td>11. My staff would benefit from using this system</td>
<td>3.6</td>
</tr>
<tr>
<td>12. This would serve as a good training/reminder tool</td>
<td>3.7</td>
</tr>
</tbody>
</table>

Each of the questions used a standard Likert scale ranging from 0 to 4, where 0 meant strong disagreement and 4 meant strong agreement. The questionnaire was administered after the participants had an opportunity to use and see the PORPOISE system being evaluated, but before any form of discussion.

After the questionnaire was administered, an informal discussion was conducted to gauge the participants’ overall feelings concerning the PORPOISE device. Factoring in the mean score values for each of the first 10 questions, the SUS scoring yielded a value of 90.6. According to Bangor, Kortum, and Miller (2009), this score equates to an adjective rating of “Best Imaginable”.

The last two questions, which were not part of the original SUS questionnaire, asked the participants to rate the system’s overall ability to achieve the intended outcome: that is, training. Thus, these questions directly measured the system’s potential to train workers to aid in the prevention of the spread of potentially deadly pathogens without the need for medical field testing. The mean responses were 3.6 (out of 4) for question 11 and 3.7 (out of 4) for question 12. These answers, together with the SUS results and the post-discussion, indicate an overall very positive outcome and well received prototype. The general post-questionnaire discussion was also very positive, and the participating managers demonstrated perceivable excitement about using the device within their various sites and locations.
Paper V represented the final step of the initial application of the created framework. The next and final section of this manuscript will address the future direction of this research and further considerations.
7 FUTURE WORK AND CONSIDERATIONS

This dissertation has described the research path that culminated in the creation of and answers to the four key research questions presented in Table 1. This path began with an attempt to discover how to provide adaptivity in a mobile setting based on a learner’s particular learning style and contextual information. This question was successfully answered in chapter three through the development, creation and evaluation of a context-aware adapted learning system on an iOS device.

The limitations uncovered upon answering the first research question led to the second research question: How can a framework be designed and evaluated to identify current trends in context-aware learning systems? This question was answered as shown in the fourth chapter through the creation and application of a framework to identify current trends in the literature on context-aware learning systems from 2009 to 2015 (inclusive).

The next step along the research journey revealed the need for a framework to provide a repeatable means for creating context-aware learning systems. This led to the formulation of the third research question on how to design, implement and evaluate a generic framework for context-aware learning systems. The research question was successfully answered by developing said framework and applying it to the systems described in chapters five and six.

All of the above systems developed in this manuscript were based on standalone devices. Although the results of the literature review (Paper II) demonstrated that server-based systems were more common (Figure 14), the reasoning behind any architectural choice depends on when the choice was made and, thus, the hardware available at the time.

In the past, server-based systems used servers as a means for data storage and processing. However, today’s modern hardware now possesses both the storage
and the computational abilities to complete almost any task. In fact, today’s higher-end mobile devices possess hardware capabilities similar to those of many servers just a decade ago.

The ever-changing hardware environment was the foundation for the final research question posed in this manuscript, which, in turn, drove the final question asked in the presented research path: What does the future hold for context-aware learning systems, and what are the possible issues?

This query raises the following question: Are server-based requirements becoming obsolete as technology advances? Server-based systems ease of collaboration and device communication; yet, how can they be achieved in a day and age when communicating devices may be on different continents?

One of the possible ways of answering (and, thus, solving) the collaboration question is to employ cloud computing. Cloud computing provides both practically unlimited offsite storage and incredible computational processing power. Yet, before this technology can be adopted, several other issues must be addressed.

Paper VI attempts to outline the main issues of which one should be aware before using a cloud computing service for education. These issues range from relatively benign problems, such as issues relating to bandwidth connectivity issues between mobile devices (Dinh, Lee, Niyato, & Wang, 2013), to more sinister concerns regarding security and privacy (Hashizume, Rosado, Fernández-Medina, & Fernandez, 2013). As with any technology, the safeguarding of sensitive data is of key importance to the educational domain (González-Martínez, Bote-Lorenzo, Gómez-Sánchez, & Cano-Parra, 2015). Therefore, safeguarding and protection must be addressed in terms of both the management of cloud computing and the security of any software using a cloud service provider (Almorsy, Grundy, & Müller, 2016). Security and access to data residing within cloud-computing are critical to educational contexts because communicated data may include student records, student accounts or student learning data. Therefore, understanding security policy concerns and potential gaps in current laws and regulations is vital to the educational domain (Jaeger, Lin, & Grimes, 2008).

However, security is not the only potential setback to cloud computing in education. Another possible setback is infrastructure limitations. More specifically, collaboration and interactivity may be negatively affected by network performance and latency (González-Martínez et al., 2015). Since cloud computing is closely related to the old client/server models of years past, it may be an interesting way to advance the field of context-aware learning systems using proven ideas.

The research and findings presented in the previous chapters and associated papers are not limited to the confines of this manuscript; they can also be used with confidence by other researchers in the field of computing. The frameworks have been demonstrated to be functional and applicable to the field and to potentially save researchers both time and effort.
The benefits of this research go beyond context-aware learning systems. The overall findings presented in this manuscript may also be of value to other research areas within the field of computing science. Possible research areas in which the research presented in this manuscript may be of value include augmented and virtual reality (AR and VR). The ability of AR and VR systems to directly interact and acknowledge users’ context information and provide relevant information could be of great benefit to further research. Additionally, as was alluded to with the PORPOISE system, the field of medical training and monitors could also benefit from real-time information based on context. However, in all cases, the potential benefits of the proposed frameworks require further validation and represent a possible direction of future study.

Although this research was limited and bounded by current hardware limitations, technologies are ever-improving. In the years to come, we may see context-aware learning technologies become ubiquitous in our daily lives. Whether in formal or informal education settings, learning should always remain a key part of our growth as a society. Integrating learning into our daily lives will allow us to continually expand our horizons and investigate the world around us.

Regardless of which path future iterations of context-aware learning systems take, the future of this field is exciting. It is the hope of the author of this manuscript that the work presented herein may nudge the field further down the research path, moving the current body of knowledge forward in a direction that helps learners advance and, ultimately, helps shape the next generation.
8 BIBLIOGRAPHY


When using mobile devices for learning the context of the learner can change. This change may affect how and what is learnt. This dissertation provides a view into the field which investigates this effect: context-aware learning.

A framework was developed and used to create two prototypes. Their successful implementation and testing shows the overall effectiveness and usability of the framework as a research tool in the development of context-aware learning systems.