國 立 交 通 大 學

資訊工程研究所

碩士論文

The Design and Implementation of Dynamic Web-Based Multimedia Templates for Empowering Users of Wireless Pervasive Computing Devices

研究生: 馬志南
指導教授: 陳登吉教授

中 華 民 國 九 十 三 年 六 月

The Design and Implementation of Dynamic Web-Based Multimedia Templates for Empowering Users of Wireless Pervasive Computing Devices

研究生: 馬志南
指導教授: 陳登吉教授

Student: Edouard Ma Poon
Advisor: Dr. Deng-Jyi Chen

國立交通大學
資訊工程研究所
碩士論文

A Thesis

Submitted to Department of Computer Science and Information Engineering

College of Electrical Engineering and Computer Science

National Chiao Tung University

In Partial Fulfillment of the Requirements

For the Degree

Master of Science

In

Computer Science and Information Engineering

June 2004

Hsinchu, Taiwan, Republic of China

中華民國九十三年六月

博碩士論文授權書

本授權書所授權之論文爲本人在國立交通大學資訊工程研究所系所__組 93 學年度第 2 學期取得碩士學位之論文。

論文名稱: The Design and Implementation of Dynamic Web-Based Multimedia Templates for Empowering Users of Wireless Pervasive Computing Devices

指導教授: 陳登吉教授

1. □同意 □不同意

   本人具有著作財產權之上列論文全文(含摘要)資料，授予行政院國家科學委員會科學技術資料中心(或改制後之機構)，得不限地域、時間與次數以微縮、光碟或數位化等各種方式重製後散布發行或上載網路。

   本論文爲本人向經濟部智慧財產局申請專利(未申請者本條款請不予理會)的附件之一，申請文號爲: ____________，註明文號者請將全文資料延後半年再公開。

2. □同意 □不同意

   本人具有著作財產權之上列論文全文(含摘要)資料，授予教育部指定送繳之圖書館及國立交通大學圖書館，基於推動讀者間「資源共享、互惠合作」之理念，與回饋社會及學術研究之目的，教育部指定送繳之圖書館及國立交通大學圖書館得以紙本收錄、重製與利用；於著作權法合理使用範圍內，不限地域與時間，讀者得進行閱覽或列印。

   本論文爲本人向經濟部智慧財產局申請專利(未申請者本條款請不予理會)的附件之一，申請文號爲: ____________，註明文號者請將全文資料延後半年再公開。

3. □同意 □不同意

   本人具有著作財產權之上列論文全文(含摘要)，授予國立交通大學與台灣聯合大學系統圖書館，基於推動讀者間「資源共享、互惠合作」之理念，與回饋社會及學術研究之目的，國立交通大學圖書館及台灣聯合大學系統圖書館得不限地域、時間與次數，以微縮、光碟或其他各種數位化方式將上列論文重製，並得將數位化之上列論文及論文電子檔以上載網路方式，於著作權法合理使用範圍內，讀者得進行線上檢索、閱覽、下載或列印。

   論文全文上載網路公開之範圍及時間 –

   本校及台灣聯合大學系統區域網路： 93 年 7 月 31 日公開

   校外網際網路： 93 年 7 月 31 日公開

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權爲非專屬性發行權利。依本授權所爲之收錄、重製、發行及學術研發利用均爲無償。上述同意與不同意之欄位若未鉤選，本人同意視同授權。

研究生簽名： 學號：9117615

日期：民國 93 年 7 月 15 日

本授權書請以黑筆撰寫並影印裝訂於書名頁之次頁。

國家圖書館博碩士論文電子檔案上網授權書

本授權書所授權之論文為本人在國立交通大學資訊工程研究所系所__組 93 學年度第 2 學期取得碩士學位之論文。

論文名稱: The Design and Implementation of Dynamic Web-Based Multimedia Templates for Empowering Users of Wireless Pervasive Computing Devices

指導教授: 陳登吉教授

□同意　　□不同意

本人具有著作財產權之上列論文全文(含摘要)，以非專屬、無償授權國家圖書館，不限地域、時間與次數，以微縮、光碟或其他各種數位化方式將上列論文重製，並得將數位化之上列論文及論文電子檔以上載網路方式，提供讀者基於個人非營利性質之線上檢索、閱覽、下載或列印。

上述授權內容均無須訂立讓與及授權契約書。依本授權之發行權為非專屬性發行權利。依本授權所為之收錄、重製、發行及學術研發利用均為無償。上述同意與不同意之欄位若未鉤選，本人同意視同授權。

研究生簽名：　　　　　　　　　　　　學號：9117615

日期：民國　93 年 7 月 15 日

本授權書請以黑筆撰寫，並列印二份，其中一份影印裝訂於附錄三之一(博碩士論文授權書)之次頁；另一份於辦理離校時繳交給系所助理，由圖書館彙總寄交國家圖書館。

國立交通大學

論文口試委員會審定書

本校資訊工程研究所碩士班　馬志南君

The Design and Implementation of Dynamic Web-Based Multimedia Templates for Empowering Users of Wireless Pervasive Computing Devices

合於碩士資格水準、業經本委員會評審認可。

口試委員 ____________________ ____________________

____________________ ____________________

____________________ ____________________

____________________ ____________________

____________________ ____________________

指導教授

____________________ ____________________ ____________________

系主任

中華民國九十三年七月十五日

The Design and Implementation of Dynamic Web-Based Multimedia Templates for Empowering Users of Wireless Pervasive Computing Devices

Student: Edouard Ma Poon Advisor: Dr. Deng-Jyi Chen

Department of Computer Science and Information Engineering

National Chiao Tung University

ABSTRACT

In this study, we present an approach to reach and augment users of pervasive computing devices to create powerful multimedia presentations using dynamic web-based multimedia templates (DWMT). In a pervasive computing environment, templates provide flexibility for mobile users with little time and resources at their disposal by assisting them when creating such presentations. We designed and implemented a traffic incident system (MobileCampus) that effectively demonstrates the practicability of DWMT and extends the use of these templates beyond the desktop PC to reach users of pervasive computing devices. We note that while several applications for pervasive computing devices have been implemented and deployed, a successful application should also identify specific challenges in a computing environment that contributes towards the vision of pervasive computing and enhances the user experience. In our approach, we take an application-centric viewpoint of these challenges, as well as the characteristics of the mobile user and device limitations into account when designing our system.

ACKNOWLEDGEMENTS

I would like to extend my appreciation and thanks to the following people, for without whom; this research thesis would not have been possible:

First and foremost, to my advisor, Professor Deng-Jyi Chen, for his continuous encouragement, insight, guidance, support and invaluable contribution throughout this research project. Thank you for helping me achieve goals that were beyond the scope of expectations.

My family, for providing me with the opportunity to undertake my studies at National Chiao Tung University. They have provided me with endless support and opportunity to reach new heights.

Lastly, my friends. I am indebted to them for their understanding, encouragement and patience.

I acknowledge that all references are accurately recorded and that, unless otherwise stated, all work contained herein is my own.

Edouard Ma Poon

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE
Introduction

1.1 The Mobile Opportunity

Pervasive computing extends the traditional computing infrastructure to a new class of devices, mobilizing itself beyond the desktop PC, enabling delivering of on-demand information wherever, whenever and on any device a user prefers. It is about accessing information, applications, and services — solutions that enable anywhere, any time, access to people, information, and processes over any network [1].

The term, pervasive computing, still means different things to different people. It is often used interchangeably with ubiquitous computing. For some, pervasive computing is about mobile access and the mechanism needed to support a mobile community of nomadic users. For others, it is about *smart* environments or *active spaces*, context awareness and the way people use devices interact with their surrounding environments [2]. Pervasive computing encompasses all of these areas, but at its core, it is about three things. First, it is concerned about how people view mobile computing devices and use them within their environments to perform tasks. Second, it is concerned about the way applications are created and deployed to enable such tasks to be performed. And third, it concerns the environment and how it is enhanced by the emergence and ubiquity of new information and functionality.

The vision of pervasive computing from an application-centric perspective, according to Banavar *et* *al*. and Couloris *et al*. [2, 3], presumes that a device can be a portal into an application-data space, not a repository of custom software that a user must manage. An application is viewed as something by which a user performs a task, not as software that is written to exploit a device's capabilities. Furthermore, a computer environment is referred to as an "information-enhanced physical space", not a virtual environment that exists to store and run software. The need for perceptual information about the environment further differentiates pervasive computing from traditional computing. An overview of the vision of pervasive computing is depicted in Figure 1 [4].

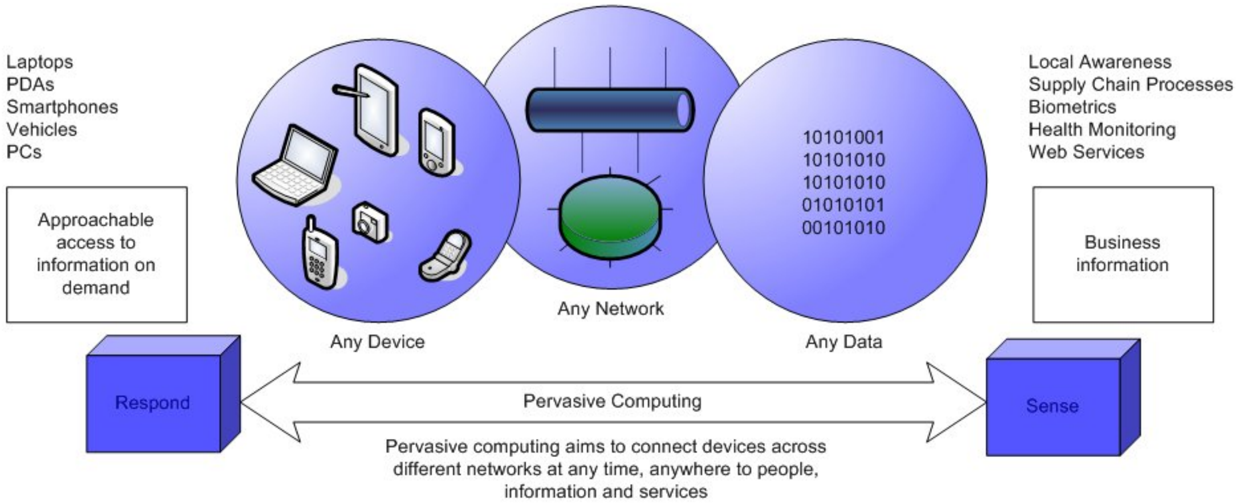

Figure 1. **Overview of pervasive computing**

Recently, pervasive computing has been maturing from its origins as an academic research area to a technical and commercial reality [2, 8, 15]. Recent hardware developments and advances in location sensors, wireless communications infrastructures and global networking [8] have made this possible. To complement this, we have seen an increase in the number of mobile users [1] as access becomes pervasive. Their ease of use concerns are driving new forms of interaction, the infrastructure is extending to new devices, new devices are demanding access to information-rich applications over high-speed networks and applications are becoming increasingly connected. The diversity of these technologies coming together has contributed towards the vision of pervasive computing – anywhere, anytime data access on any device [9].

An increasing amount of electronic information takes place in the form of multimedia – a mixture of animation, images, video, audio and text. As users become more reliant on pervasive computing devices, there will be a growing need to bring multimedia information to these devices [4, 9, 10]. These advances have opened several new *challenges*, opportunities and perspectives for developing and interacting with pervasive computing applications that allow these devices to effectively access, store and process multimedia information [1, 2, 8,15]. In addition, applications will have a greater awareness of the user context and environment, and thus be able to provide more intelligent services on behalf of the user by exploiting these rich sets of services available within the computing environment, reducing the burden of the user having to direct and interact with applications [11].

It is also becoming obvious that the nature of interactions between users and computers must evolve [11]. The way mobile computing devices and applications are developed, deployed, managed and used today does not meet the expectations of the user community and falls short of the potential for pervasive computing [2]. In the realm of pervasive computing, every aspect of development creates a challenge that is intermittently linked with other challenges that realize the potential and vision of pervasive computing.

1.2 Motivation

1.2.1 Challenges for Pervasive Computing and Mobile Users

Enterprises and academic institutions are realizing the advantage of mobile computing and are beginning to mobilize existing applications and implementing new mobile applications [12]. The proliferation of heterogeneous devices over recent years has spurred the need for increasingly complex applications that demand access to information-rich resources. While such diversity has led to different vendors pursuing different standards, the challenges for pervasive computing remain, essentially the same. These challenges are divided into several categories [1, 8, 13], namely:

- Mobility challenges;
- Device challenges;
- Usability challenges;
- Business challenges; and
- Application challenges.

The exploration areas for each of these challenges are vast, but together, they contribute towards and strengthen the vision of pervasive computing. We discuss these challenges in more detail in the following chapter.

When developing, deploying, managing and using applications, we are also concerned about the way mobile users performs tasks as well as the imposed device limitations. Mobile users are identified as having the following characteristics:

- They are constantly *on the move*;
- They have little time and resources at their disposal to perform tasks; and
- They need to perform tasks with the least amount of time, effort and skill.

Device heterogeneity poses unique constraints and limitations. In particular, we identified the following relevant characteristics:

- Screen size, resolution and color depth
- Processing power and battery power
- Other unique device capabilities

From an application-centric view, the successful development and deployment of a pervasive computing application is dependent on primarily two areas. First, how well the system addresses particular application challenges, supports the vision of pervasive computing and enhances the user experience, and second, how well the system *assists* the user perform tasks efficiently and effectively given the above-mentioned characteristics while considering the device limitations.

1.2.2 Dynamic Web-Based Multimedia Templates (DWMT)

According to Webster's [14], a template is defined as "a document or file having a preset format, used as a starting point for a particular application so that the format does not have to be recreated each time it is used". Templates are useful, because they save time and create flexibility and opportunity for end users to customize them according to their preferences.

The focus of this research is on dynamic web-based multimedia templates or DWMT (herewith referred to as *templates*) for pervasive computing devices. Templates contain a specific set of criteria that describe the structure of the template, the semantic relationships between multimedia elements and rules for customization. The purpose of this research is to propose extending these templates to users of pervasive computing devices as a way of assisting users to create powerful multimedia presentations from their devices, relieving them of intensive task duties and limitations posed by these devices.

It is believed that the dynamic feature of these templates is particularly suitable for the pervasive computing environment that enables on-demand information. Dynamically assembled templates have several advantages over static-based templates, specifically in terms of managing, integrating and communicating information to constituents in real-time. By implementing dynamic templates we can also address different device specifications and user requests at run-time without having to create several presentations of a single template to support different devices or possible user requests. Section 3.3 will present a detailed explanation on DWMT.

1.2.3 Features of a Successful Pervasive Computing Application

A successful pervasive computing application is perceived as one that addresses the challenges for pervasive computing set out in section 1.2.1. In such an environment, great emphasis is placed on the user and the user tasks rather than on the goals of the application [8]. In summary, a successful pervasive computing application can be considered as one that:

- Addresses specific application challenges found relevant to the purpose of the application, supports the vision of pervasive computing and enhances the user experience
- Identifies the characteristics of users that will use such an application and take these into account in the development of the application; and
- Considers device limitations by working around them to achieve desired outcomes

As a result, in designing and implementing the proposed system, we focus our efforts on addressing the above-mentioned properties of a successful pervasive computing application from an application-centric perspective.

1.3 The Problem and it's Setting

1.3.1 The Statement of the Problem

This study is to design and implement a system that uses dynamic web-based multimedia templates to assist users to create powerful multimedia presentations with the aim of contributing towards the vision of pervasive computing and enhancing the user experience.

1.3.2 The Statement of the Sub- Problems

- Investigate and identify certain challenges of developing and deploying successful applications in a pervasive computing environment
- Establish some of the specific criteria from an application-centric viewpoint with particular focus on the application challenges, mobile user characteristics and device limitations; and analyze and study the attributes of related applications considered necessary for the development and deployment of successful pervasive computing applications
- Design and implement a system that demonstrates the use of dynamic web-based multimedia templates and addresses the criteria for successful pervasive computing applications; Integrate the proposed system with the BestWise International Computing Company Multimedia Visual Authoring Tool [5] with the intention of extending the use of this tool to users of pervasive computing devices
- Discuss and evaluate how the proposed system's use of templates and its implementation as a whole contributes to the vision of pervasive computing and enhances the user experience

1.4 The Scope of the Research

1.4.1 Delimitations

Currently, there is no major supported open standard supported by vendors developing pervasive computing applications. While the proposed system was developed for the Microsoft Windows CE platform for PDA and Smartphone devices, the concept and goal of the application remains applicable across all development platforms and devices.

The proposed system attempts to conform to open standards wherever possible to support seamless interoperability and platform independence. Templates were developed using XML/XSLT, two open standard technologies supported by all major vendors. Furthermore, the proposed system conforms to standards specified by open standard consortiums such as the World Wide Web Consortium (w3C). However, standards do have some drawbacks, such as limiting some capabilities that could resolve some problems by vendor-specific platforms. The proposed system combines these capabilities by offering integrated customized applications and web content to deliver a rich application and content delivery environment to address the mobile users' tasks.

Considering all the challenges for pervasive computing is extremely complex and not economically practical. Thus, the proposed system considers an application-centric viewpoint of the specific challenges, characteristics of mobile users and device limitations – this viewpoint is deemed credible for the determination of a successful pervasive computing application – and defines the scope of the computing environment.

1.4.2 Assumptions

It is assumed that the proposed system operates on a pervasive computing infrastructure centered on a high-level conceptual model consisting of integrated devices, users, software components and user interfaces.

The proposed system is dependent on Wi-Fi (IEEE 802.11b technology) that guarantees continuous connections wherever and whenever necessary. Network traffic, stability, signal strengths and other hardware issues are not taken into specific account in this study.

1.5 Research Methodology

In this research, we propose designing and implementing a pervasive computing application, MobileCampus, which demonstrates the use of web-based multimedia presentation templates to assist mobile users perform tasks efficiently and effectively. In addition, we integrate the proposed system with the Multimedia Visual Authoring Tool [5] so that the functionality of this tool is extended to users of pervasive computing devices. The success of the proposed system is evaluated from a qualitative perspective in terms of the set application challenges, mobile user characteristics and device limitations as well as its contribution towards the vision of pervasive computing and enhancing of the user experience.

The rest of study will employ the use of qualitative methods whereby an extensive literature study will contribute to the part of the dissertation. Books, journals, website publications, magazines, newspapers, conference proceedings and other articles will be used to study different perspectives. The research thesis will be covered by an extensive literature review of the identified problem in order to validate the problem. Discussions will be subjective, expressive and flexible and provide a holistic view of the topic.

The primary mode of communication used to discuss these issues with researchers in the field and professionals will be through e-mail and use of electronic discussion groups. A classification matrix will be used to identify professionals and their perspectives and how their studies or opinions relate to the sub-problems identified.

This study will be discussed and evaluated from an application-centric perspective.

1.6 Thesis Organization

The organization of this paper is divided into two discrete parts. The first part is concerned with the importance of the challenges for pervasive computing. Chapter Two provides a clear outline of the evolution of pervasive computing and the general challenges associated with the development and deployment of successful pervasive computing applications. It provides a background to the understanding of the thesis. In addition, we emphasize the specific criteria for the successful development and deployment of the proposed system from an application-centric viewpoint and define this as our scope.

The second part of the study is concerned about the design and implementation of the proposed system. Chapter Three describes the proposed system, MobileCampus, in detail. This is followed by a detailed discussion and qualitative evaluation of the system in terms of the specific criteria identified for the successful development and deployment of the proposed system in Chapter Four. Chapter Five concludes the thesis and offers suggestions for future research.

CHAPTER TWO

Challenges for Pervasive Computing and Related Work

2.1 Introduction

Creating successful pervasive computing applications is dependent on several factors, which we identify as challenges. These challenges do not exist in isolation but together, contribute towards the vision of pervasive computing. This chapter explains the evolution of pervasive computing from its basic beginnings to where it stands today and how it fits into our lives; it also discusses the properties of pervasive computing and how these might apply to the proposed system. Based on this, we identify the general challenges for pervasive computing and explain these in detail. As a result, this chapter forms as a basis for the groundwork towards stressing the specific challenges, characteristics of mobile users and limitations imposed by devices that need to be addressed from an application-centric viewpoint and define the scope of the proposed system.

2.2 The Evolution of Pervasive Computing

The continuing evolution of computing (Figure 2) has led to an explosion of information collection and access. Pervasive computing defines a major evolutionary step in the work that began in the mid 1970s when the PC first brought computers closer to people [4, 8].

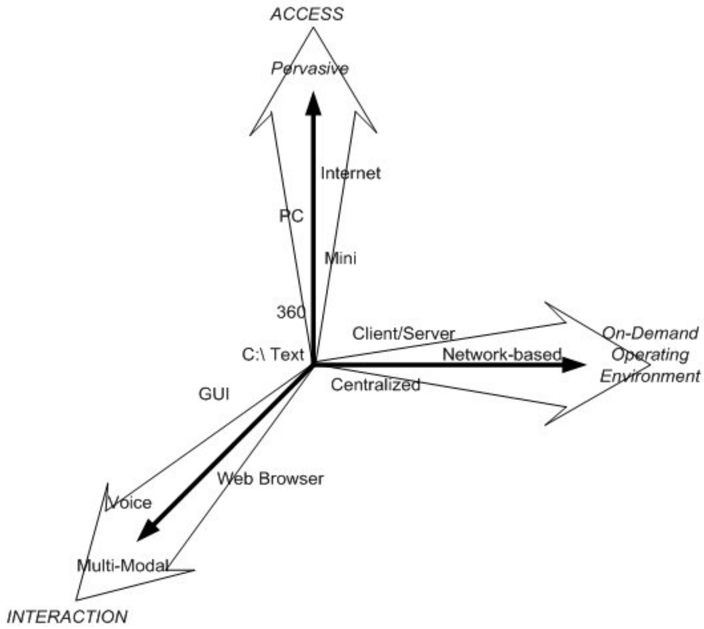

Figure 2. **The continuing evolution of computing**

Although the PC has not delivered the full potential of information technology to users, it has certainly taken the first step towards making computers popular (if not pervasive). It has also been an instrumental factor in the phenomenal growth of hardware components and the development of graphical user interfaces as well as the beginnings of distributed systems and mobile computing devices.

In the rest of this section, we try to sort out the complex relationships between distributed systems and mobile computing and develop a taxonomy of issues characterizing each phase of this evolution towards pervasive computing.

2.2.1 Distributed Systems

Networking provided the opportunity for personal computers to evolve into distributed computing. Computers connected by network – whether mobile or static, wired or wireless, sparse or pervasive – offered sharing capabilities [8, 15]. Distributed computing marked the next step towards pervasive computing by introducing seamless access to remote information resources and communication with fault tolerance, high availability and security [15]. The advent of the World Wide Web's networking ubiquity made it an attractive choice for experimenting with distributed computing concepts. The ad-hoc nature of the Web's growth proved we can distribute computing capabilities in a big way without losing scalability. But, most importantly, the Web pioneered the creation of a nearly ubiquitous information and communications infrastructure creating an essential starting point for pervasive computing.

2.2.2 Mobile Computing

Mobile computing emerged from the integration of cellular technology and the Web [16]. This has extended to wireless networks and different types of devices that operate in these environments. In mobile computing, the basic principles of distributed systems design continued to apply [15], but four key constraints of mobility forced the development of specialized techniques: unpredictable variation of network quality, lowered trust and robustment of mobile elements, limitations on local resources imposed by weight and size constraints, and concern for battery power consumption [17].

Mobile Computing's goal, according to Saha and Mukherjee [8], of "anytime, anywhere" is essentially a reactive approach to information access but prepares for pervasive computing's proactive approach. As Figure 3 shows, pervasive computing forms as a superset of mobile computing and an extension of distributed computing. Pervasive systems require support for interoperability, scalability, smartness and invisibility to ensure that users have seamless access to computing whenever they need it.

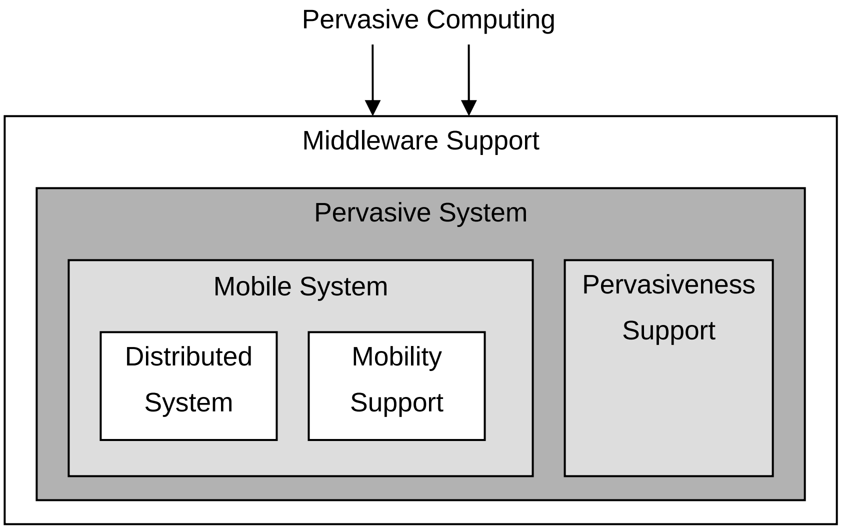

Figure 3. **System view of pervasive computing**

2.2.3 The Pervasive Computing Model

Technological advances necessary to build a pervasive computing environment fall into four broad areas: devices, networking, middleware and applications. Figure 4 shows the relationships between these areas concerned.

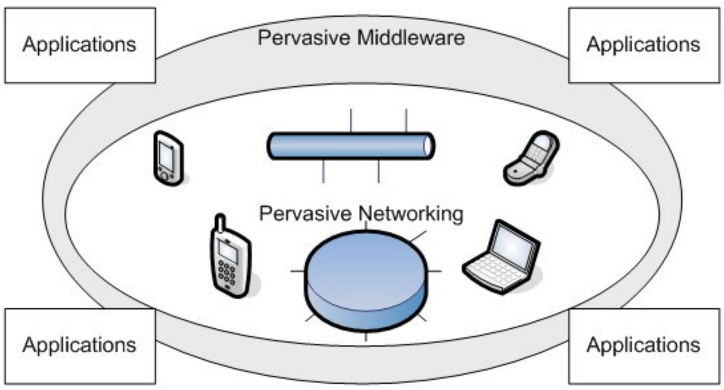

Figure 4. **Pervasive computing framework**

Pervasive computing addresses these areas in the following way: devices will be heterogeneous, and intelligent; networks will be integrated, middleware will act as an interface between the networking kernel and end-user applications and the applications will be environment-centric, either web-based or mobile computing or a combination of both [8].

2.3 Pervasive Challenges

To migrate from a pervasive computing vision to reality, fundamental issues in numerous areas need to be addressed from both technical and non-technical viewpoints. We identified previously in section 1.2.1, five primary categories of challenges that need to be considered when developing and deploying applications in a pervasive computing environment: mobility challenges, device challenges, usability challenges, business challenges, and application challenges. We discuss these in more detail to determine their applicability and emphasize some of the specific challenges found important to the development and deployment of the proposed system, MobileCampus.

2.3.1 Mobility Challenges

Mobility challenges associated with pervasive computing encompass a range of issues that are associated with the nomadic user of pervasive computing devices. These issues are largely hardware related and depend much on the state of the wireless environment, for example, the maintenance of connections as devices move between areas of differing network connectivity, and the handling of network disconnections [11]. While protocols for wireless networking do handle some of the problems associated with mobility, such as routing and handovers, there are some problems that cannot be solved at the network level as they require knowledge of application semantics. According to Henricksen *et al*. [11], the computing infrastructure needs to co-operate with applications in order to perform tasks related to the device mobility. Some of the mobility challenges are outlined below [4]:

- **Connection state**. As users migrate from location to location, they will not always receive continuous connections. Protocols and the computing infrastructure need to be able to intelligently handle connected, intermittent and disconnected states with applications.
- **Authentication, authorization and security**. Wireless environments are often vulnerable to eavesdropping and are considered less secure than their wired counterparts. This makes authentication and access to secure services important.
- **Voice and data access**. The ability to access information using different modes of information transfer presents a challenge, not only for enhancing user interaction but also an infrastructure challenge.
- **Device management**. Devices in a pervasive computing environment will not only include devices that users will interact directly with such as handheld organizers, mobile phones and traditional PCs, but also include devices such as sensors and actuators that mediate between physical and virtual environments; embedded devices in objects such as watches and shoes; and home and office appliances such as videos, toasters and telephones [11]. These devices will demand a computing infrastructure that can manage them well.
- **Scalability**. Scalability is a universal challenge for pervasive computing, both from a hardware and software perspective [4, 11]. As the number of devices increase and decrease and demand services at irregular intervals, the computing infrastructure, the interactions between components and the software services provided must all be scalable. According to Henricksen *et al*. [11], a powerful software platform would have characteristics such as scalability, fault-tolerance and distributed components. The Ninja service architecture [19] is an example of one such platform.
- **Services**. The need for integrated services that communicate with each other to support mobile users is an essential characteristic of pervasive computing. Messaging services, location and

context awareness (invisibility) as well as intelligent notification are all properties of pervasiveness.

2.3.2 Device Challenges

The heterogeneity in computing systems will not disappear in the future, but instead increase as the range of computing devices widens. Heterogeneous devices will be required to interact seamlessly, despite the varying differences in hardware and software capabilities. This will require an infrastructure that maintains knowledge of device characteristics and capabilities and manages the integration of devices into a coherent system that enables arbitrary interactions [4, 11]. Some of the more specific challenges include:

- **Unique device challenges**. Each device creates a new pervasive computing challenge as new devices have unique characteristics and capabilities. Support for existing and future devices presents challenges for developing and deploying pervasive computing applications.
- **Varying programming models**. There are several existing programming models, languages and frameworks for developing pervasive computing applications. Currently there are no dominant or major standards for programming models.
- **Wide range of target environments**. Varying operating systems and underlying software infrastructures pose integration challenges for deploying universal pervasive computing applications.

2.3.3 Usability Challenges

Pervasive computing presents an increasingly difficult challenge when developing and deploying pervasive computing applications. Many analysts argue that usability is the number one barrier to the successful deployment of pervasive computing applications [20]. Without a simple, intuitive and efficient user experience, organizations may end up with huge investments in applications that mobile workers refuse to use. As mentioned earlier, a pervasive computing application is whereby a user performs a task rather than software written to exploit a device's capabilities [8, 15]. The heterogeneity and the particular characteristics of mobile users make the interaction challenges extremely complex. Demand for ubiquitous access will create a requirement for universal interfaces [11]. Device heterogeneity will introduce a further requirement that user interfaces are highly adaptable. And, the diminishing amount of user interaction with applications and the changing nature of interactions will mandate the creation of new types of user interfaces.

Universal interfaces refer to the creation of generic interfaces that will allow the semantics of user interaction to be specified without reference to rendering or input modalities. Currently, there are attempts to create a special language that meets this challenge, such as MoDAL [21].

Highly adaptable interfaces needed to cope with different user contexts (for example, when a user switches from working in an office to driving a car) are one of the usability challenges for pervasive computing applications. Novel application behaviors are needed to cope with this demand.

User interfaces for pervasive computing must be designed carefully with several factors in mind. First, the ergonomics of the interface must be designed to keep the user's attention focused on the task at hand rather than on peripheral matters. Second, it should be enjoyable to use. Third, the user interface must allow novel types of interaction that will become more common as computing tasks become more ubiquitous, such as the delegation of tasks and provision of guidance to software agents. And finally, user interfaces should be designed for ordinary people, rather than for technologists [22].

2.3.4 Business Challenges

Taking advantage of the pervasive computing opportunity means leveraging non-traditional, embedded computing technologies – both wired and wireless – to enable, integrate and extend e-business opportunities and new applications [23]. Improving productivity, developing and applying better cost-management strategies, finding new markets, adding value to existing customer relationships and maintaining competitive advantage in an ever-changing marketplace are all part of the business challenges. While the business challenges aren't new, circumstances have changed dramatically. One misconception about migrating to pervasive devices as simply screen scraping or content transformation [13], the challenge lies in reconfiguring the business logic necessary to process the workflow of the application.

Pervasive computing has enabled and created a demand for mobile business capabilities. Enterprises are establishing strategies for mobilizing their businesses and integrating them with existing solutions and business partners to maximize returns.

By mobilizing workforces, pervasive computing applications provide several advantages, including less room for error, relaying critical data residing on corporate networks to devices when an immediate decision needs to be made, and delivering information in a time and format that is useful to mobile workforces [23]. By assisting workforces, pervasive computing can help bring valuable productivity gains to the enterprise and adding value to customer relationships.

There are several ways in which pervasive computing can meet the business needs in a mobile environment [12, 23]. There exist several opportunities both in terms of customer and enterprise perspectives.

2.3.5 Application Challenges

Although all challenges for pervasive computing are linked in one way or another, application challenges remain one of the most important challenges when creating applications for pervasive computing devices. We place great emphasis on specific application challenges in this project, and discuss them in more detail later.

In the area of service platforms in a pervasive computing environment, there are currently studies on adaptation, context awareness, smart environments, scalability, integration and interoperability, invisibility and so forth [8, 9, 11, 23, 24], however, on the specific challenges for pervasive computing applications, the following issues are relevant [4, 9, 20, 24]:

- **Automation**. The idea of *doing more by doing less* is achieved through three things [24]: new technologies should be brought into our lives, not vice versa; new technologies must increase human productivity; and integrating technologies so that they become ubiquitous and require minimal intervention. A thin client approach supports this vision – by minimizing the client's functionality and providing support within the network at servers [15, 25]. Client thickness is usually determined by the worst-case environmental conditions under which the application must run satisfactorily. For example, a very thin client suffices if one can always count on a high bandwidth, low latency wireless communication to nearby computing infrastructure and batteries that can be recharged or replaced easily [15]. On the other hand, thin client approaches, from a usability perspective, can assist users with their tasks by transferring processing of data to back-end systems and hiding system complexity. This optimizes the user experience by allowing the user to do less to achieve more.
- **Content aggregation**. In a pervasive computing environment that enables on-demand information, content aggregation offers aggregating of disparate data dynamically from

different sources or services and converting it to a single presentation for the users' pervasive computing device.

- **Customization and personalization**. Customization and personalization solve individual user preferences adds value to the overall user experience.
- **Multi-device capability**. An impedance mismatch is apparent between devices in terms of capability and support. Supporting existing and future devices is a challenge for any pervasive computing application. Applications should be able to handle multi-device profiles.
- **Web content and custom applications**. Combining the use of both types empowers users with a rich application and content delivery environment. This provides rich context information that is gathered from a wide range or sources, interpreted, and disseminated in a scalable fashion to end users [11].
- **Integration of services**. Integration across heterogeneous protocols and services should be seamless and unnoticeable to the user.

2.4 Mobilizing a Successful Application

Creating successful pervasive computing applications is always a possibility, however, special attention needs to be paid on how such applications might adapt and be effective in a particular pervasive computing environment [9]. One can argue with the notion that mobile devices will always be one order of magnitude *weaker* than their desktop counterparts. As a result, any application expected to run on pervasive computing devices has to account for smaller screen sizes, lower processing power, smaller storage and memory capacity and lower network bandwidth [9]. Clearly, there is a need for alternative thinking how applications can be architected to provide reasonable levels of responsiveness despite the device hardware limitations.

In the previous chapter, we discussed the aspects of pervasive computing environments and how they differed compared to traditional computing environments. What is important to note at this stage is that while pervasive computing devices will almost never achieve the power of their desktop counterparts, they cannot be compared directly because their environment of use, purpose of use, and approach is altogether different. We approach our problem by exploiting these limitations and taking advantage of alternatives to achieve the reasonable results from a pervasive computing perspective.

There have already been several successful commercially deployed pervasive computing applications, such as at National Express [26], The Waterbury Police Department [27], Lotte Sam Kang [28], Kudos Restaurant [29] and Enning GmbH [30]. In the previous chapter, we identified three specific criteria that we identified as necessary for the development of successful pervasive computing applications within a particular computing environment; a successful pervasive computing application is considered as one that:

- Addresses specific application challenges found relevant to the purpose of the application, supports the vision of pervasive computing and enhances the user experience
- Identifies the characteristics of users that will use such an application and take these into account in the development of the application; and
- Considers device limitations by working around them to achieve desired outcomes

In the following sub-sections we attempt to discuss the relevant properties of successful pervasive computing applications and define the scope of the proposed system.

2.4.1 The Pervasive Computing Environment

The pervasive computing environment extends beyond the traditional networked environment to utilize wireless technologies such as IEEE 802.11, CDMA, GPRS, GSM and so forth. Such an environment addresses the goal of providing “anytime, anywhere” information access by decoupling users from devices and viewing applications as entities that perform tasks on behalf of users. There are a number of ongoing projects in this area that focus on holistic pervasive computing environments, including PIMA [2], Carnegie-Mellon University’s Aura project [31], The University of Washington’s Portolano project [32], HP’s CoolTown project [33], MIT’s Project Oxygen [34], The University of California at Berkeley’s Endeavour project [35] and IBM’s Websphere Anyplace project [36].

Most of these projects focus their efforts on creating a platform or environment in which pervasive computing applications might adapt and excel. Other efforts include infrastructure issues, dynamic service discovery, device independent models and so forth.

Due to the extensive and complicated process of setting up a pervasive computing environment, the proposed system will not focus on the network or hardware infrastructure, but rather, as mentioned in the previous chapter, approach the problem from an application-centric perspective and assume such an environment.

2.4.2 Specific Application Challenges

In Section 2.3.5, we identified the application challenges that we found necessary to the development and deployment of successful pervasive computing applications. We focus our efforts on these challenges:

- Automation;
- Content aggregation;
- Customization and personalization;
- Multi-device capability;
- Web content and custom applications; and
- Integration of services.

The proposed system will address each of these application challenges in an attempt to support the vision of pervasive computing from an application-centric perspective. Each of these challenges will be discussed in greater detail in the concluding chapters and will be critically evaluated to identify particular contributions and limitations.

2.4.3 Mobile Users' Characteristics

Usability is an expansive topic in pervasive computing. While it is a research topic on its own, it does not exist in isolation and needs to be considered when developing and deploying pervasive computing applications. In the previous chapter, we identified the characteristics of mobile users and earlier discussed the usability challenges. In a pervasive computing environment, great importance is placed on the user, how users perform tasks [2] and how results are achieved. We define our target users have the following characteristics:

- They are constantly *on the move*;
- They have little time and resources at their disposal to perform tasks; and
- They need to perform tasks with the least amount of time, effort and skill.

While there are several usability aspects concerned with the development of such applications, the proposed system is particularly focused on how the users perform tasks and enhance the user experience rather than on other aspects associated with usability. In the development of the proposed system, we hope to address these characteristics.

2.4.4 Pervasive Computing Device Constraints and Limitations

In addition to considering how users perform tasks, we also need to consider the devices they use and how we can exploit each device constraints and limitations to maximize user efficiency while maintaining a reasonable computing performance. We address the following device limitations:

- Screen size, resolution and color depth
- Processing power and battery power
- Other unique device capabilities

2.5 Summary

This chapter presented an overview of the evolution of pervasive computing which provided insight into the realm of developing applications for such an environment. Challenges facing the development and deployment of pervasive computing applications were identified and discussed in detail. It is understood these challenges do not exist in isolation and by taking into account and addressing these challenges presents opportunity for successful development and deployment of applications in a pervasive computing environment. This chapter provided a foundation and understanding about the context of pervasive computing in which we outlined criteria (application challenges, characteristics of the mobile user and the limitations imposed by devices) considered important for the success of the proposed system.

The following chapter introduces and discusses, in detail, the proposed system, MobileCampus.

CHAPTER THREE

System Design and Implementation

3.1 Introduction

The previous chapter introduced background to the evolution of pervasive computing and the concept of a successful pervasive computing application. As a result, specific criteria were determined from an application-centric viewpoint that needs to be considered for the proposed system to be successful. It defined the scope for the system deployment as well as the foundation for evaluation.

This chapter introduces the proposed system, MobileCampus. While the proposed system is conceived to be application for general multimedia data applications, MobileCampus is a specific demonstration of the applicability of the system in such an environment. MobileCampus is discussed in great detail from a high and low level viewpoint to provide a holistic view of the system from various perspectives. The translation and transformation process of templates explain how we integrate MobileCampus with the Multimedia Visual Authoring Tool and deliver these templates to users of pervasive computing devices. To demonstrate the practicability of the application in such an environment, we present two scenarios that fully exploit the functionality of MobileCampus.

3.2 System Design and Implementation: MobileCampus

3.2.1 MobileCampus

In this research, we design and implement a system that uses dynamic web-based multimedia templates to assist users to create powerful multimedia presentations from their pervasive computing devices. The use of this application is not limited and can be conceived to be applicable for general multimedia data applications. MobileCampus is a traffic incident management system that demonstrates the use of templates and how they might assist mobile users and enhance the user experience while considering device limitations.

Business Situation

In a university, parking is often an ongoing day-to-day problem because the number of vehicles often exceeds the number of parking places available. In the case of conferences or exhibitions, the problem usually gets worse. Often, vehicles are found to be in the wrong place at the wrong time and officers are not equipped to deal with the situation immediately. One problem is that owners are unable to be contacted at that time causing disruptions in traffic and instability to the traffic situation on campus. Another problem is, even if an officer can produce a ticket and leave it on the car, the owner may only be notified once he or she returns to his or her car in the late evening. In addition, in the case of campus speeding, officers are often in poor positions to deal with such situations. MobileCampus utilizes the university wireless environment by equipping campus officers with mobile devices that interact in this environment and capture necessary information about the traffic violation, for example, parking and speeding on campus, and uploaded this information to a database and make the violation available to violators online. By having a central repository of vehicle registration numbers and owner's contact details, campus officers can, at anytime and anywhere access such information and inform owners of their violations in real time. In addition, we demonstrate how templates can be used to assist the officer perform tasks efficiently and effectively.

MobileCampus is divided into six services that operate dependently and independently of one another:

- Multimedia Capturing Service
- Template Management Service
- File Transfer Service
- Template Creation Service
- Content Management Service
- Internet Service (Web and E-mail)

The Multimedia Capturing Service resides on the end users' client device and incorporates the capturing of multimedia. This service communicates with the File Transfer Service to transfer files and templates between the server and the client device. The File Transfer Service connects to the Template Management Service (and the template library), which handles the template selections, importing of templates and storing of media files from the client device. The Content Management Service is activated when the end user requests to view a presentation. It dynamically assembles the content, data, layout and style to a single presentation using the Internet Service which in turn renders the presentation for the client device. Figure 5 depicts a system overview of MobileCampus and how each of the system's services relates to each other.

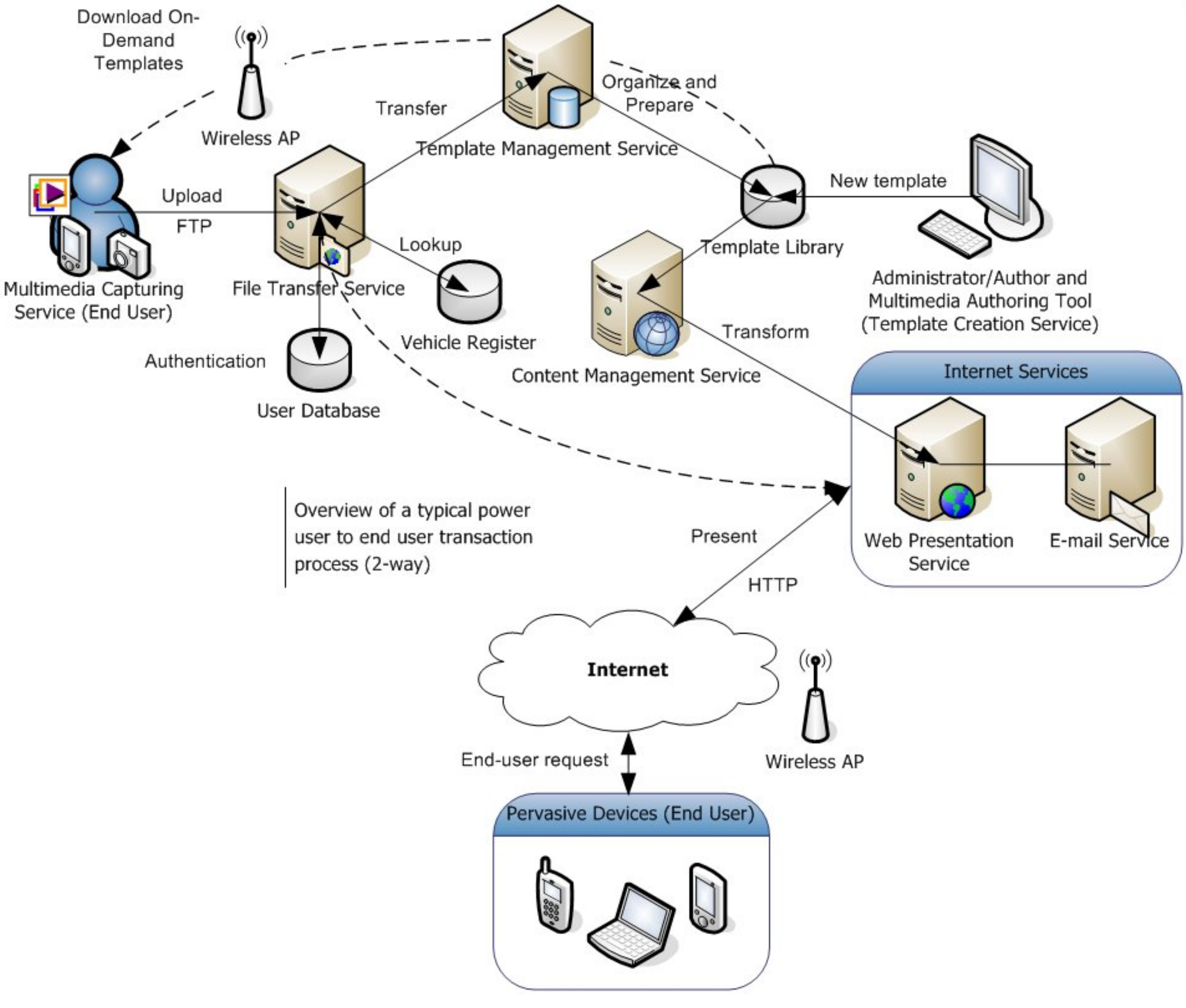

Figure 5. **MobileCampus system overview**

Figure 6 presents the system architecture of the proposed system mapped from the system overview and how each service constitutes to the system as a whole and the interactions between end users and system services.

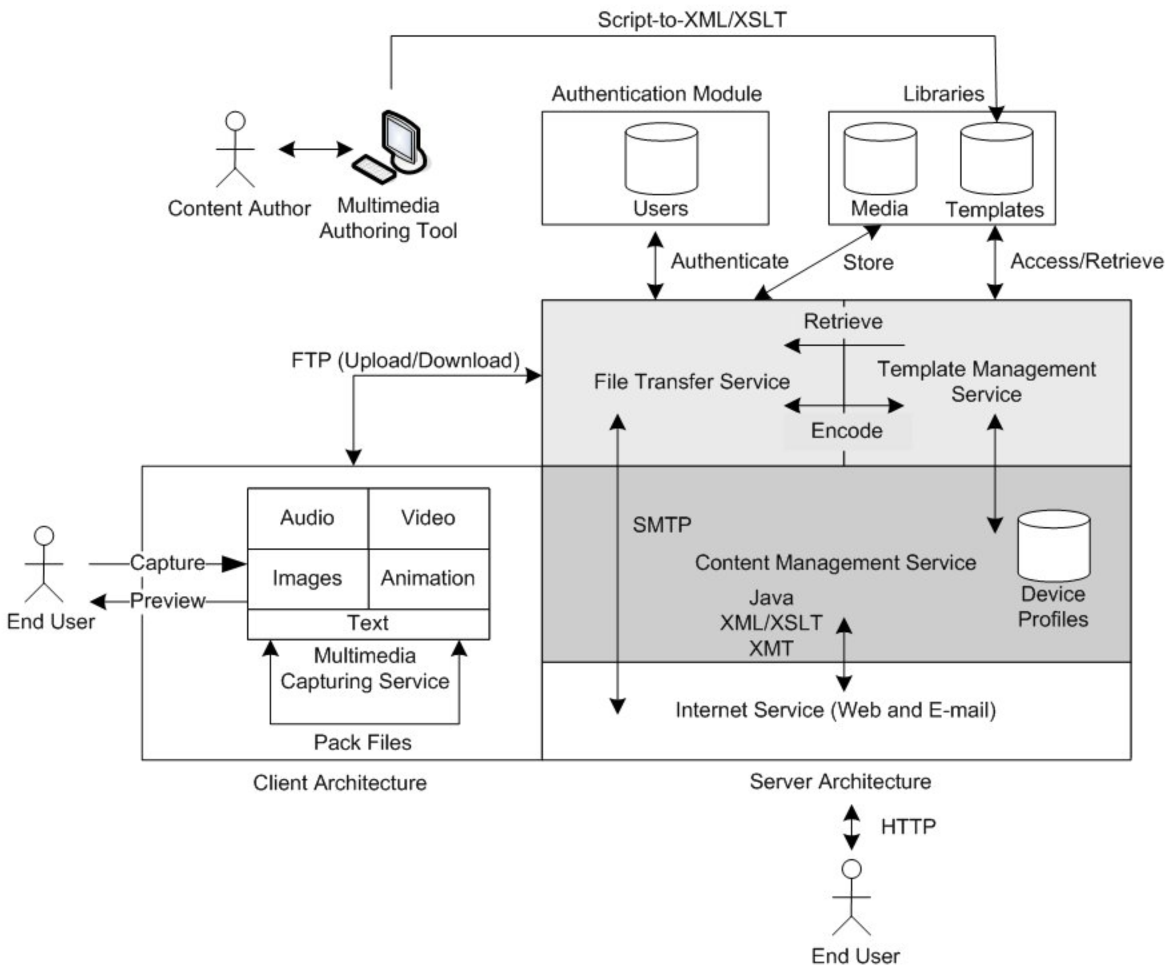

Figure 6. **MobileCampus system architecture**

We first describe each of these services in detail and later run through two scenarios to demonstrate the applicability of MobileCampus.

3.2.2 The Multimedia Capturing Service

The Multimedia Capturing Service is client-oriented and works on the pervasive computing device. This service facilitates the capturing of multimedia such as the acquiring of audio, images, text and video on the PDA and compresses them for optimal bandwidth transfer. Figure 7 depicts the process of acquiring multimedia. For this service, the pervasive computing device needs a camera attached to the device in order to acquire images and video. The built-in microphone facilitates the voice recordings and stylus input for text.

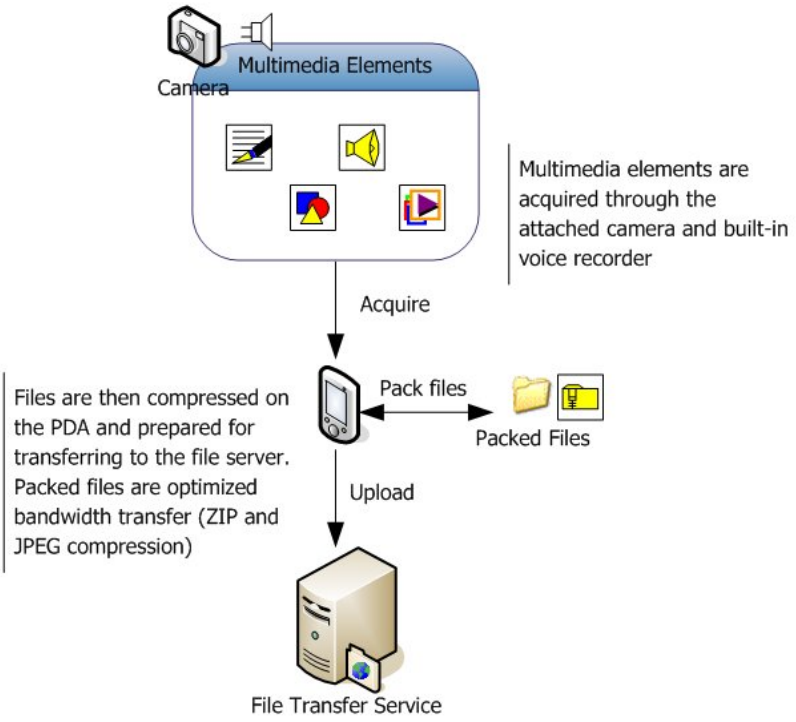

Figure 7. **The Multimedia Capturing Service**

The user can specify the following resolutions:

- 640 x 480;
- 320 x 280; and
- 160 x 120.

In addition, the user can specify certain lighting and color conditions such as:

- Auto-white balance;
- Fluorescent light;
- Incandescence light;

- Outdoor light; and
- Increase and decrease brightness.

Figure 8 depicts the screenshots of the Multimedia Capturing Service. Its simplistic interface makes it very convenient for users to capture multimedia.

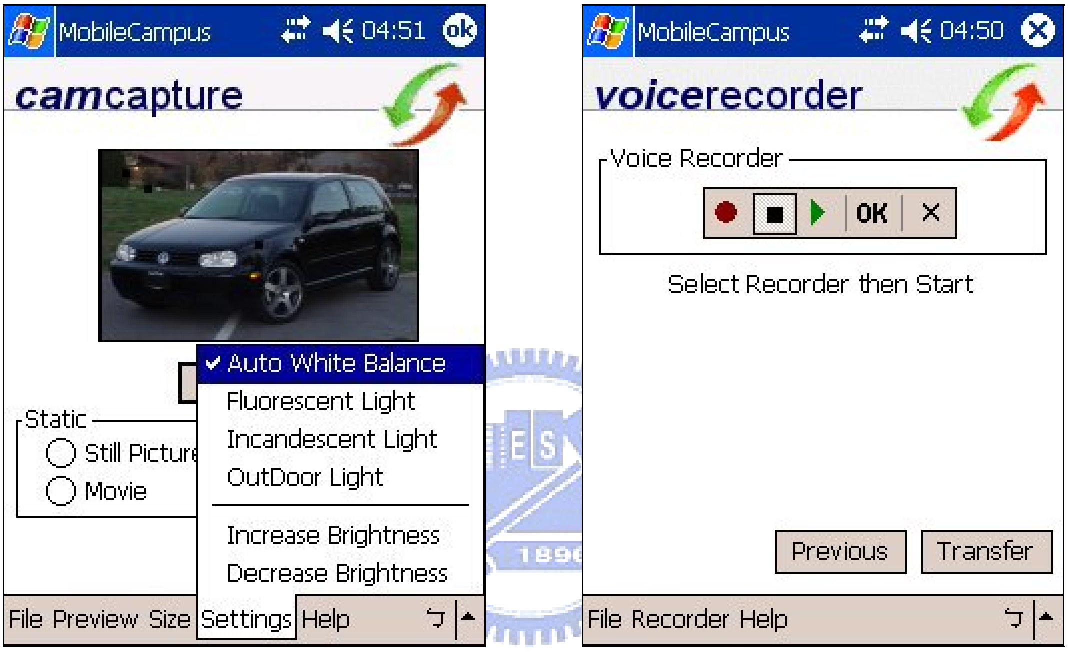

Figure 8. **Multimedia Capturing Service screenshots**

Figure 9 depicts the hardware components of the device used by the end user. The entire set includes the handheld device [7], the expansion pack and the Veo PhotoTraveler [6], used for capturing multimedia. The built-in voice recorder in the handheld device facilitates the acquiring of audio.

Figure 9. **Hardware device components**

3.2.3 The Template Management Service

The Template Management Service is called by the File Transfer Service when the files are transferred from the client to server. The purpose of the Template Management Service is to organize and store the media in the template and media database as depicted in Figure 11. It separates the content, data, layout and style. This information is stored in an XML (Figure 10) that describes the business logic and disparate data information for each transaction.

```
<!DOCTYPE usertemplate [
        <!ELEMENT usertemplate (template)*>
        <!ELEMENT template (registration, time, date, location,
         comments, officername, officeremail, officerphone, imagelocation,
        videolocation1, videolocation2, voicelocation, templatechoice)>
                <!ATTLIST template id CDATA #REQUIRED>
                <!ELEMENT registration (#PCDATA)>
                <!ELEMENT time (#PCDATA)>
                <!ELEMENT date (#PCDATA)>
                <!ELEMENT location (#PCDATA)>
                <!ELEMENT comments (#PCDATA)>
                <!ELEMENT officername (#PCDATA)>
                <!ELEMENT officeremail (#PCDATA)>
                <!ELEMENT officerphone (#PCDATA)>
                <!ELEMENT imagelocation (#PCDATA)>
                <!ELEMENT videolocation1 (#PCDATA)>
                <!ELEMENT videolocation2 (#PCDATA)>
                <!ELEMENT voicelocation (#PCDATA)>
                <!ELEMENT templatechoice (#PCDATA)>
]>
```

Figure 10. **The XML Data Type Declaration**

In addition, the Template Management Service encodes video into popular formats such as MPEG-4 [45] and WMV for optimal playback on devices.

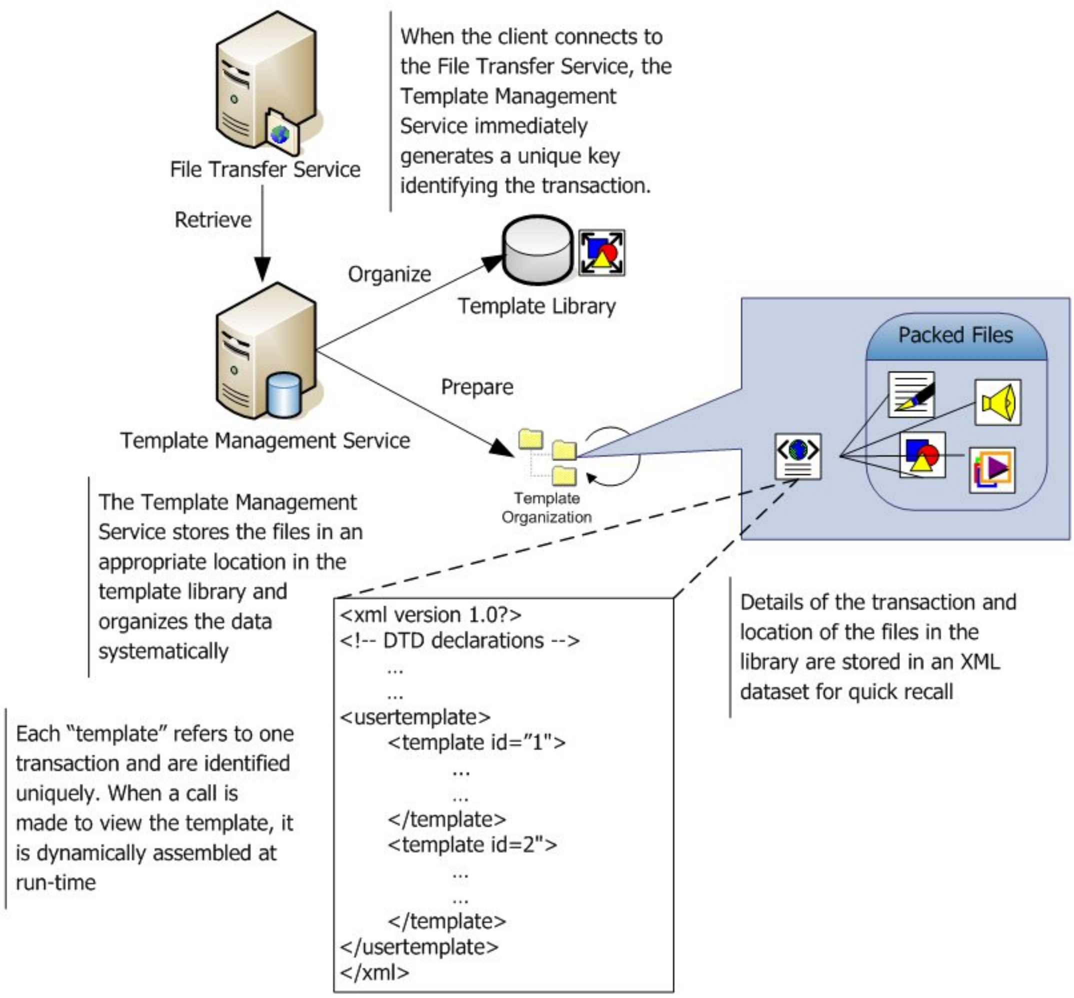

Figure 11. **The Template Management Service**

The Template Management Service also handles the importing of new templates created by the Multimedia Visual Authoring Tool into the template and media library.

3.2.4 The File Transfer Service

The File Transfer Service operates on both the client and server sides over the FTP protocol. The

•

File Transfer Service can handle multiple clients and deals primarily with the user authentication and the transferring of files to and from the server. It also connects to the e-mail server when necessary, for example, if the owner of a vehicle is registered in the database, the owner is e-mailed immediately and notified of the violation. At the same time, the File Transfer Service connects to the Template Management Service to store necessary information about the transaction and organize the transferred files on the server.

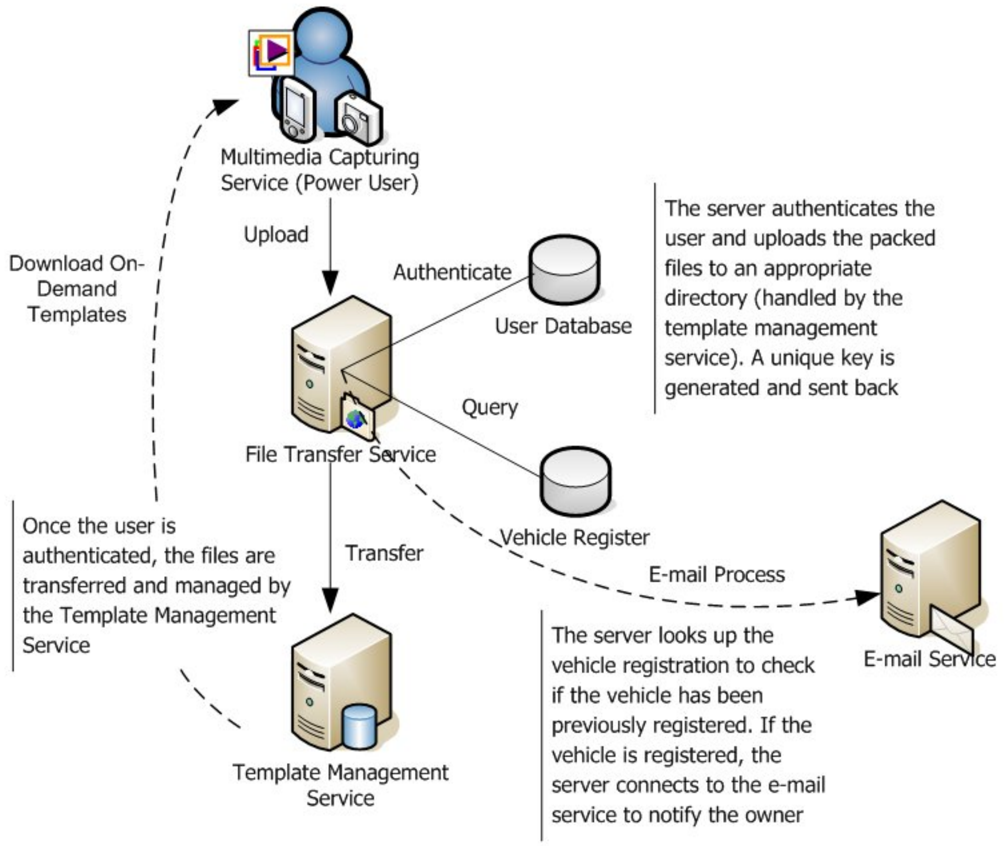

Figure 12. **The File Transfer Service**

In particular, the File Transfer Service is called in the following two situations (Figure 13):

- When the user wishes to transfer acquired multimedia to the server
- When the user wishes to access on-demand templates and preview them

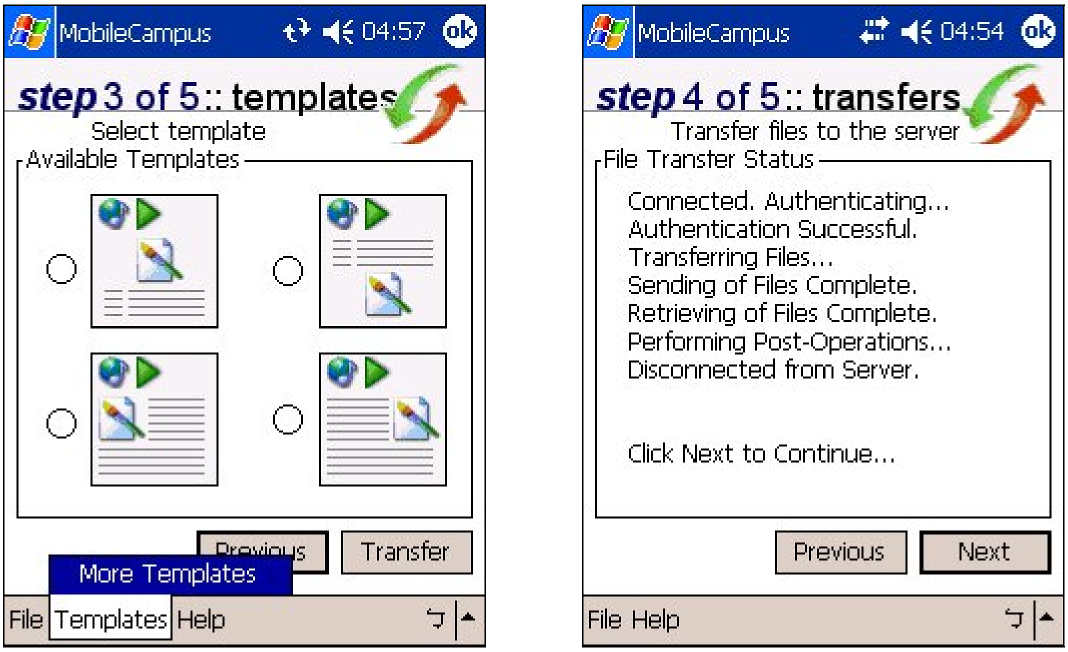

Figure 13. **File Transfer Service screenshots**

Figure 14 depicts a screenshot of the transactions on the server side.

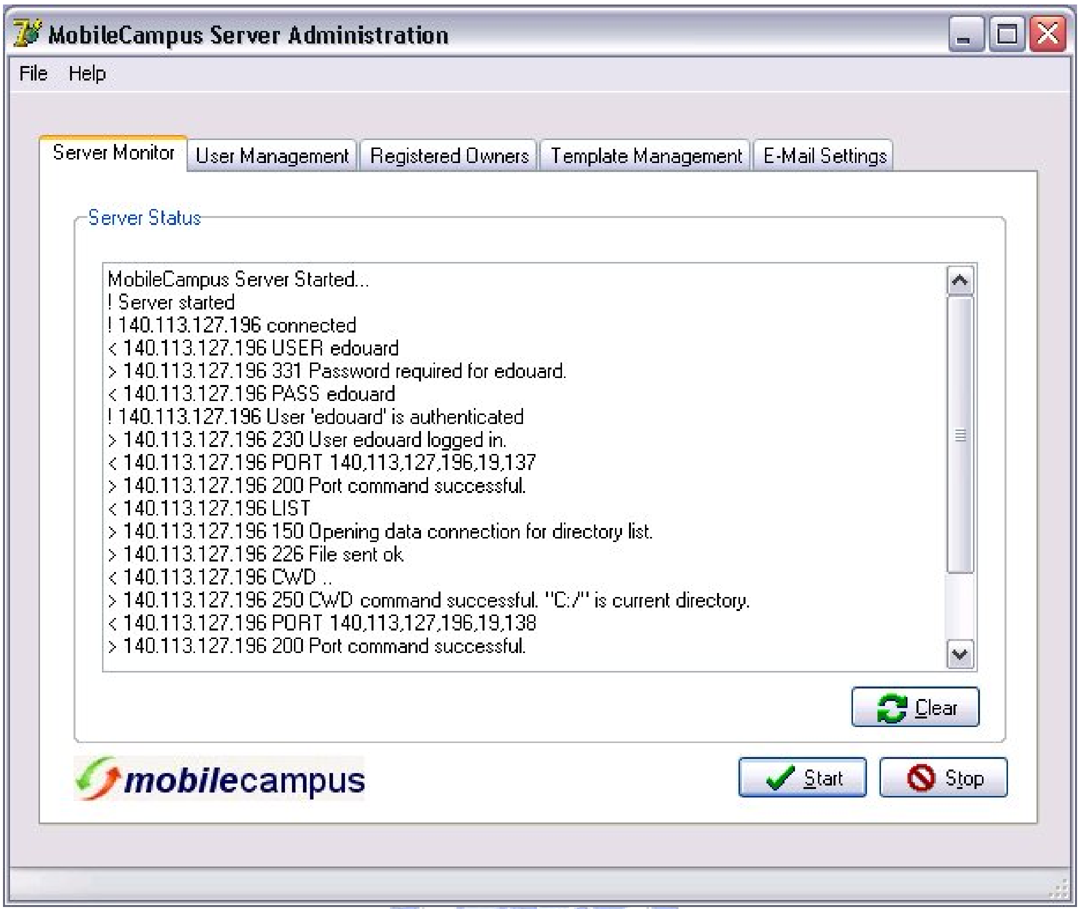

Figure 14. **File Transfer Service Server screenshots**

3.2.5 The Template Creation Service

The Template Creation Service is directly configured under the Multimedia Visual Authoring Tool. It handles the creation of templates and the translation process from the template specification to XML and XSLT for support on pervasive computing devices. While the Multimedia Visual Authoring Tool has powerful capabilities to create entire multimedia presentations, it is also effective for creating basic templates that can be exported into various formats, including formats supported by pervasive computing devices, in this case XSLT.

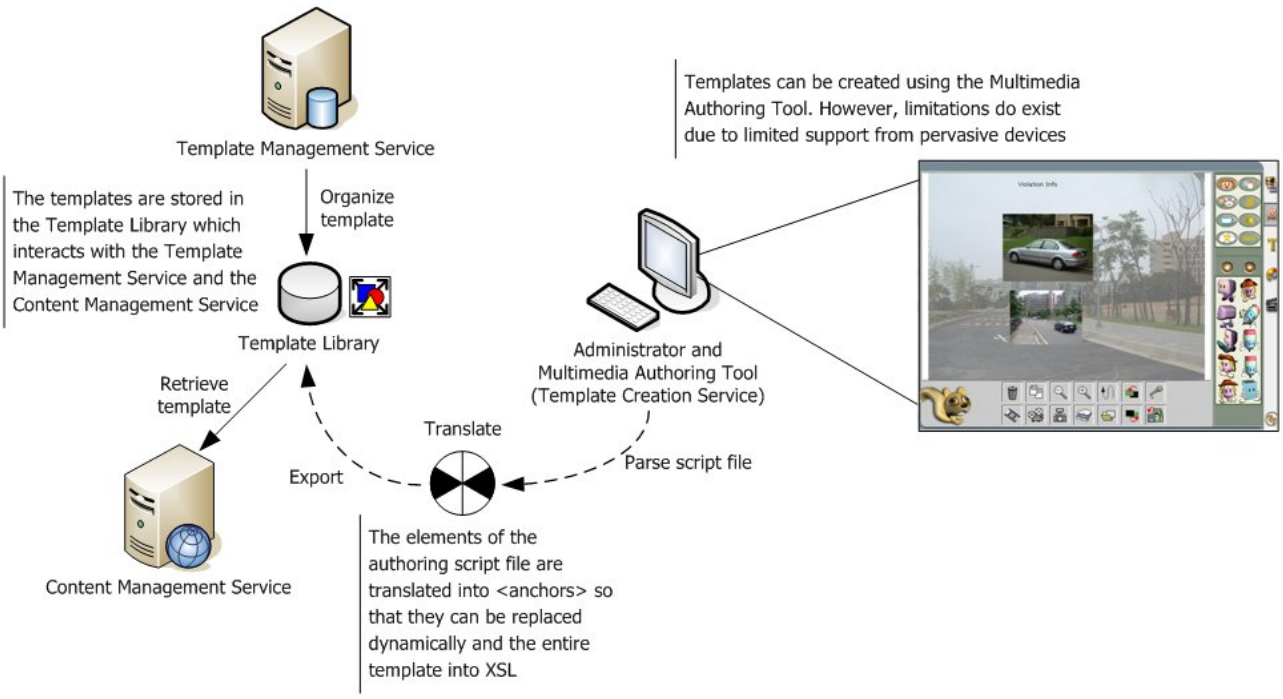

Figure 15. **The Template Creation Service**

Figure 15 illustrates how the Multimedia Visual Authoring Tool exports a script file that is parsed and translated into XML and XSLT format which is in turn imported into the template library for later use.

3.2.6 The Content Management Service

The Content Management Service is called when a user requests to view the presentation. It dynamically assembles the content, layout, style and data and aggregates the content to present a coherent web presentation to the end user.

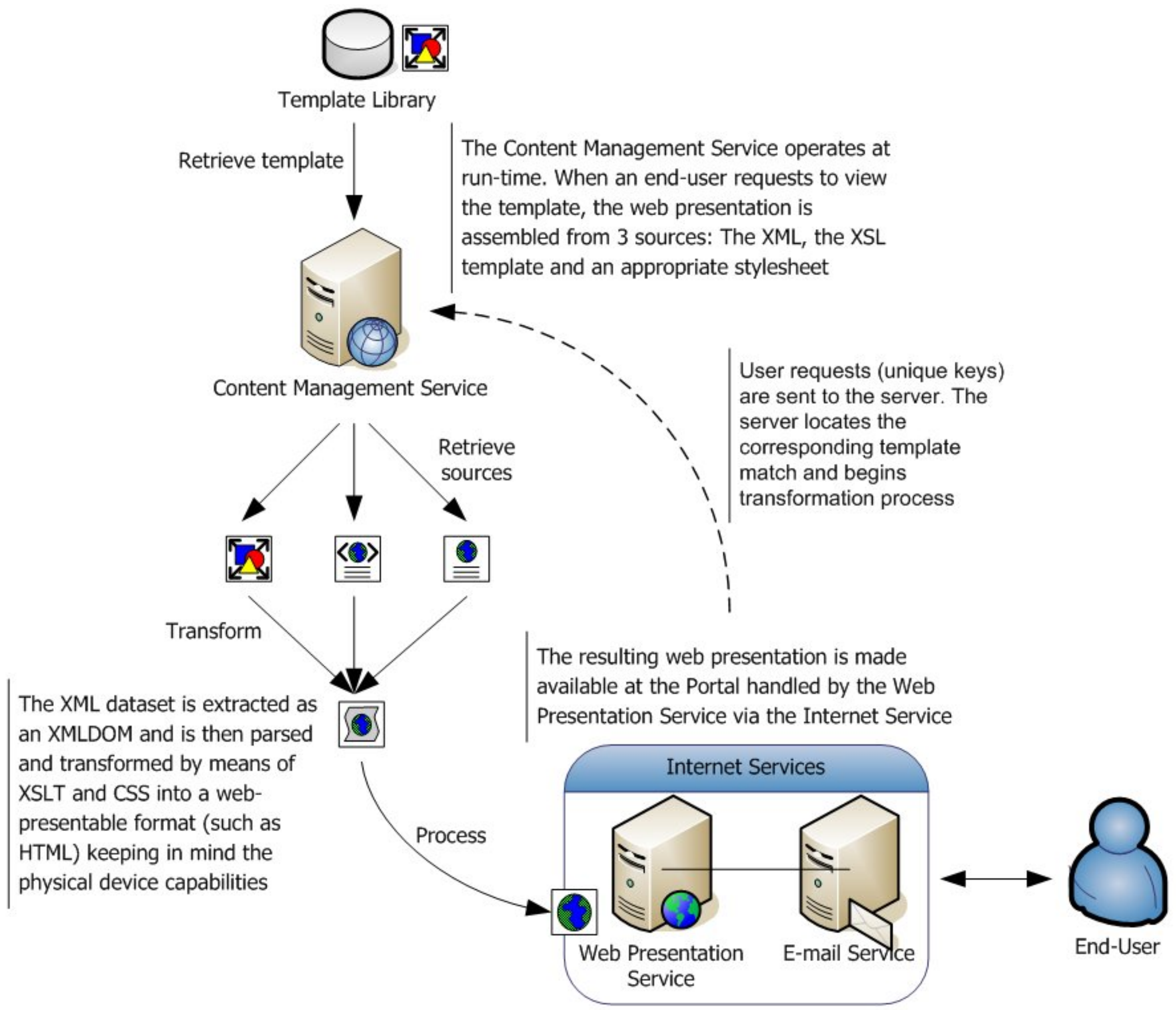

Figure 16. **The Content Management Service**

Figure 16 illustrates the process of gathering information from disparate data sources and dynamically assembling them at run-time and aggregating the content and rendering it for presentation to the end user.

3.2.7 The Internet Service

The Internet Service consists of both the web presentation service and e-mail service. It works at the presentation layer and renders the final presentation format for end users. The web server acts as a portal and calls the content management service when processing a user request. The file transfer service utilizes the e-mail server when needed to communicate with end users.

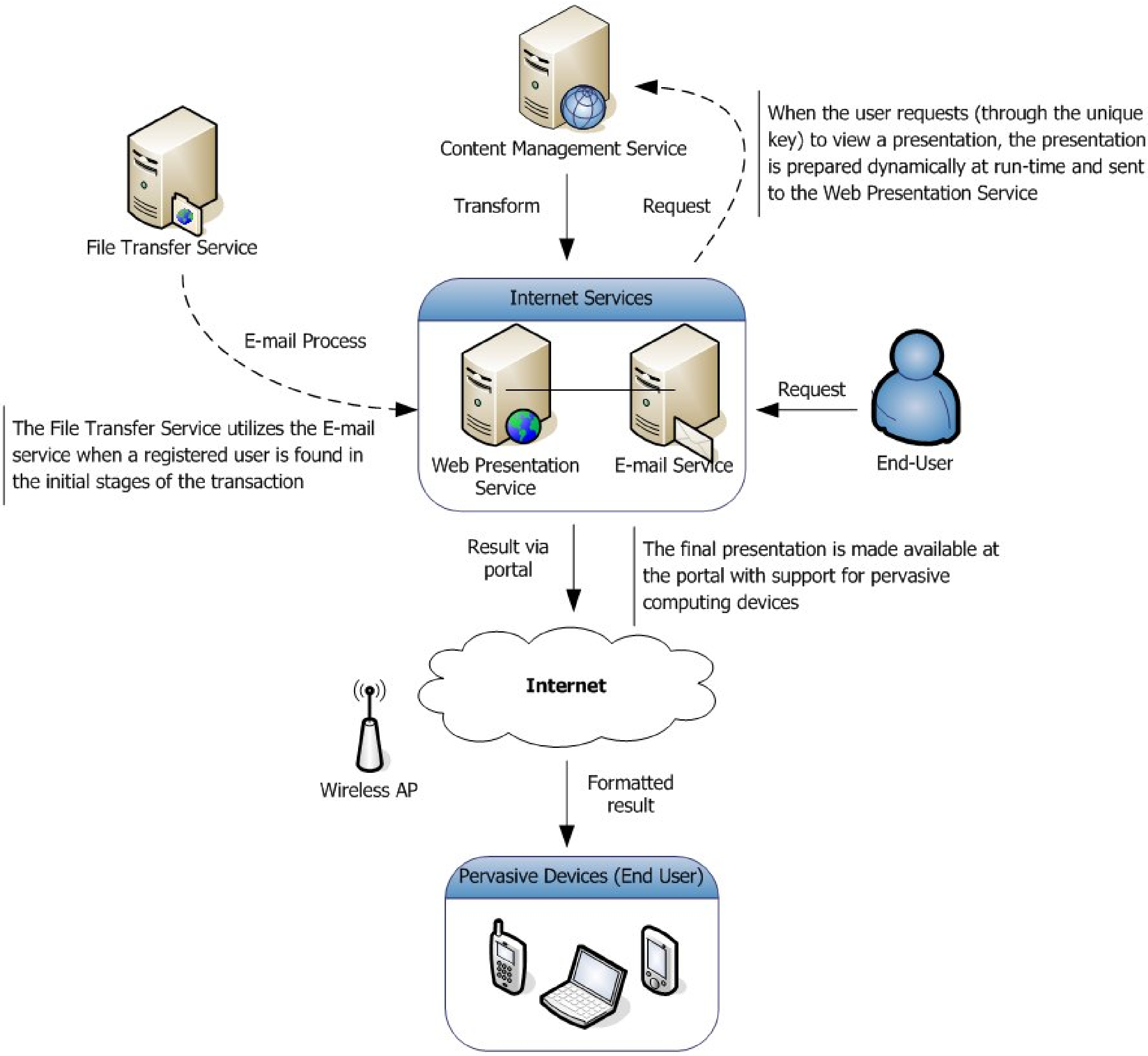

Figure 17. **The Internet Service**

Figure 17 illustrates the request from the end user to the portal. This request is passed on to the Template Management Service where the presentation is assembled and presented to the end user via the web presentation service.

Figure 18 contains a screenshot of the portal. End users enter the key received through e-mail or written on their violation tickets to view their violation online.

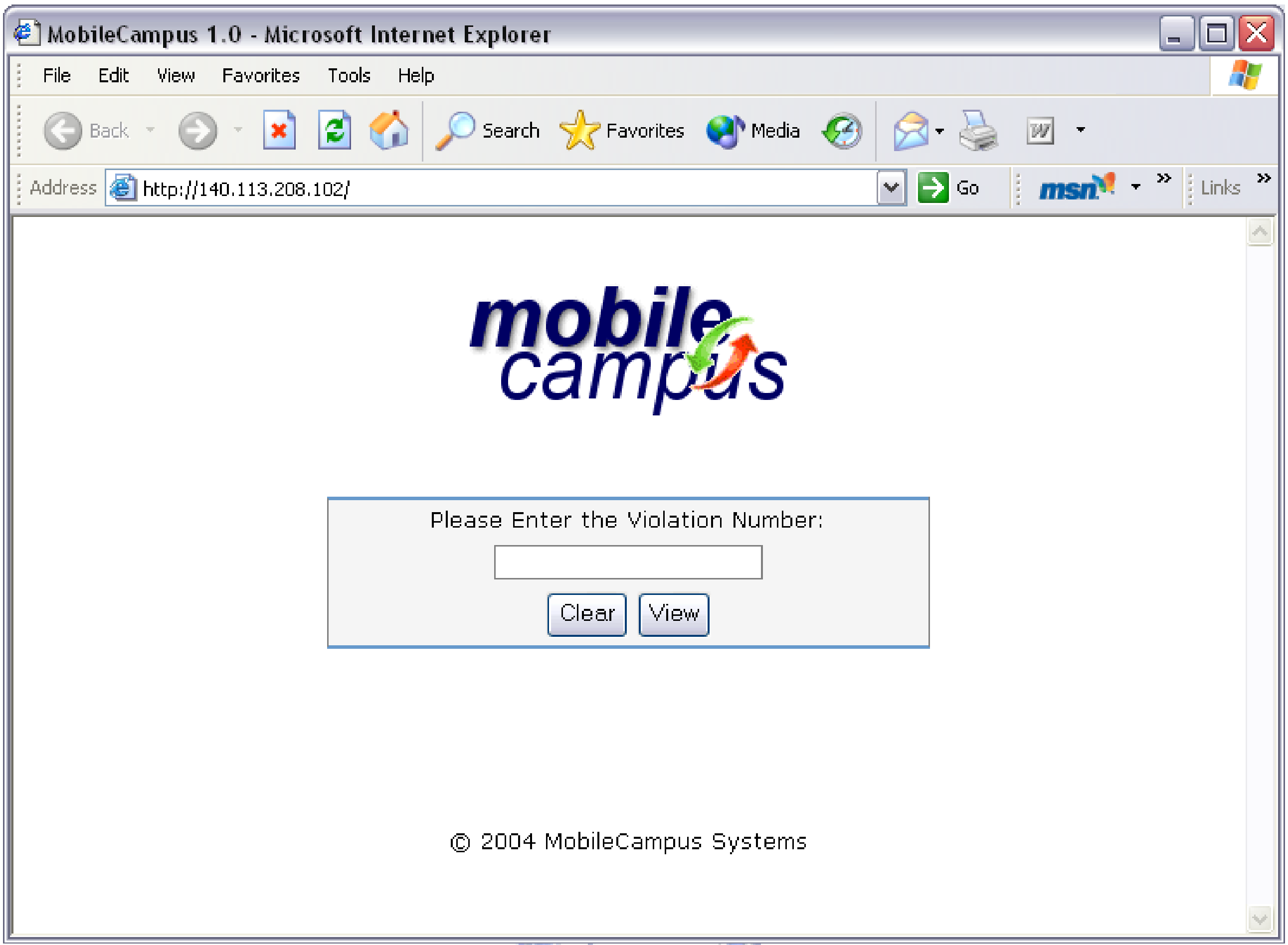

Figure 18. **The web portal (desktop view)**

Figure 19 illustrates a view of the final presentation using templates from the device perspective.

Figure 19. **Viewing the violation (device view)**

3.2.8 Other System Functionality

The proposed system contains other general functionality used to manage the system. These include:

- **User Management**. User management contains campus officer information as well as authentication information needed to access templates and transfer files.
- **Registered Owner Management**. This is the database containing registered vehicle owner information. If the vehicle's owner is registered in the database, an immediate e-mail notification is sent to the owner to inform him or her that their vehicle has received a violation notice.
- **Template Management**. This functionality deals with importing templates from the Multimedia Visual Authoring Tool into the template and media database. It is made available to end users to download and preview.
- **E-Mail Settings**. This functionality contains the relevant settings for outgoing e-mail.

3.3 Dynamic Web-Based Multimedia Templates (DWMT)

In Chapter One, we briefly discussed DWMT as the focus of this research and how it could be used to assist users to create powerful multimedia presentations from their devices. In this section, we will discuss, in detail, our approach to dynamic web-based multimedia templates and how they might assist users and address device limitations, the advantages and disadvantages of static and dynamic templates, and finally the translation and transformation process.

3.3.1 Approaching Templates

In this study, we propose using templates as a method of assisting users and addressing device limitations. Templates are particularly useful because they provide a starting point and create options for the user to create powerful multimedia presentations without needing much effort, knowledge or skills about multimedia presentations. In addition, the dynamic approach of creating templates helps address device limitations such as screen size and supported color depth. Templates contain a specific set of criteria that describe the structure of the template, the semantic relationships between multimedia elements and rules for customization. In this study, templates have the following properties:

- They are dynamic;
- They are web-based; and
- They support multimedia

Templates are dynamic because it was found that dynamically assembled templates are better suited for content aggregation, scalability, and better device support. They are web-based so they are able to reach a larger device audience through standard formats such as HTML. And finally, they support multimedia to enrich and enhance the user experience.

3.3.2 Static vs. Dynamic Templates and Device Independence

Templates exist either statically or dynamically. Static templates are configured manually whereas dynamic templates are configured at run-time. Static templates are particularly useful if the presentation does not depend on on-demand information at run-time. A comparison of static and dynamic templates is summarized in Table 1.

Table 1. **A comparison of static and dynamic templates**

Criteria	Static Templates	Dynamic Templates
Supports real-time, on-demand content delivery	No	Yes, adaptable to real-time, on-demand content delivery
Supports heterogeneous pervasive computing devices	No	Yes, device profiles can be stored and loaded dynamically depending on the device
Supports customization and personalization	No	Yes, use of style sheets offers customization opportunities
Supports content aggregation	No	Yes
Applicability towards proposed system	Applicable, but difficult to manage	Applicable and scalable to support new devices
Rendering issues	Fast, no rendering involved	Slower than static templates

Device heterogeneity poses presentation formatting problems. A presentation created for a large-screen device, should, theoretically be displayed and interacted with a small-screen device because they have the same mark-up [37]. However, this is not always the case because of the small form factor issues. Currently there are several attempts [37, 38, 39, 40] to handle device profiling. Device profiling is the process whereby certain specifications of the device are stored in a repository and called when a device connects so that the content is formatted correctly for that specific device. Ideally, content authors would need to create only one version of their Web content for delivery on multiple devices. During the delivery and rendering process, the transformation would create a presentation to match the delivery device's capabilities. Device profiling is still in its infancy, and like ISAPI [38] and MMIT [41], are platform dependent. Cocoon and HP's CC/PP [40], are open standard attempts and both make extensive use of XML and XSLT for device customization, but are complex to implement.

When developing for pervasive computing devices, one must ask some key questions [37]:

- How can we express a Web application independent of the delivery device?
- How can we adapt device-independent applications to suit delivery device capabilities?
- How can authors retain some control over the final presentation of their content?

Device profiling and device independence is important, because creating separate versions for each kind of device is both economically and administratively impractical. However, there are currently no standards supporting this, despite efforts to create one. We chose XML/XSLT because they are two open standard technologies that can be configured dynamically and particularly suited to handling the translation and transformation process respectively. XML is simple and lightweight for passing data. It separates content from business logic and presentation and supports content aggregation. XSLT is ideal for transforming from multiple data outputs to a single source for different devices. In addition, for web-based multimedia presentation templates, we found XSLT extremely robust and an ideal technology for holding information on the criteria of the structure of the template, the semantic relationships between elements and rules for customization.

3.3.3 The Translation and Transformation Process

As mentioned earlier in this chapter, the proposed system will be integrated with the Multimedia

Visual Authoring Tool. There are several benefits associated with this, as users will be able to create customized templates without prior knowledge or understanding of how the templates are coded and created. This supports our vision of minimizing and hiding the complexity of the system to the user. In this section, we discuss the translation and transformation process – two important processes that deal with the construction and delivery of the templates. These processes are depicted in Figure 20.

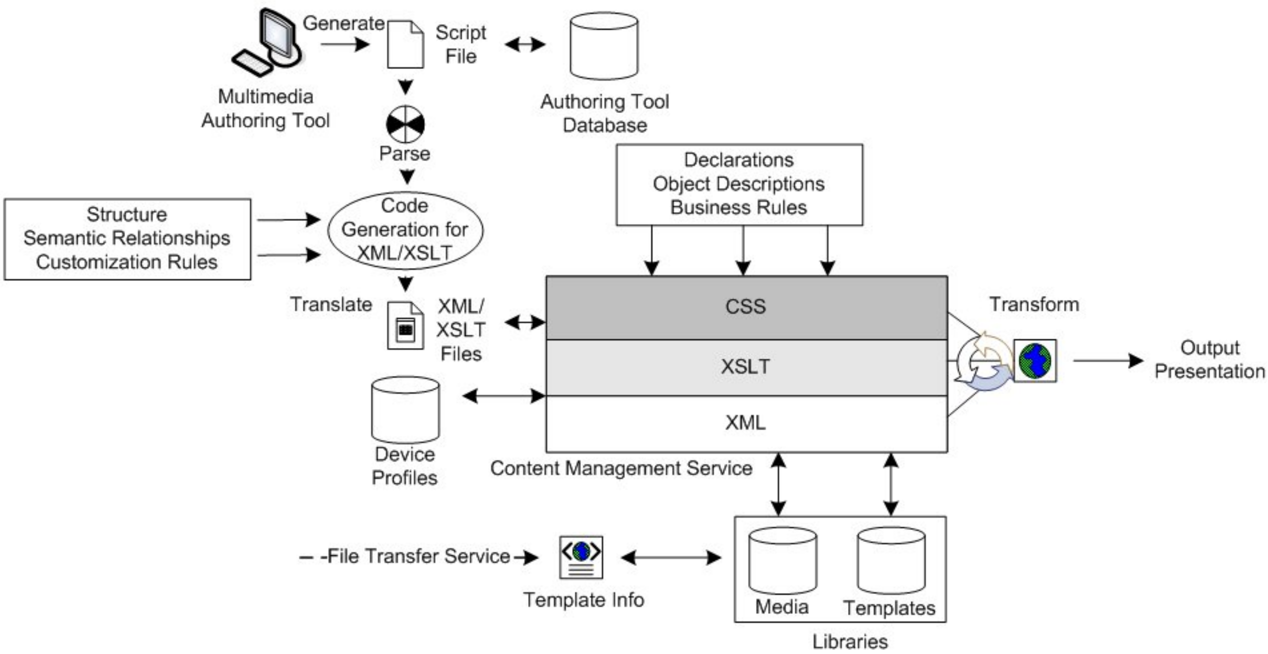

Figure 20. **The translation and transformation process**

The Translation Process

To integrate the proposed system with the Multimedia Visual Authoring Tool, we modified the translator to support the conversion from the Authoring Tool's multimedia script file to XML and XSLT supported by the resultant device.

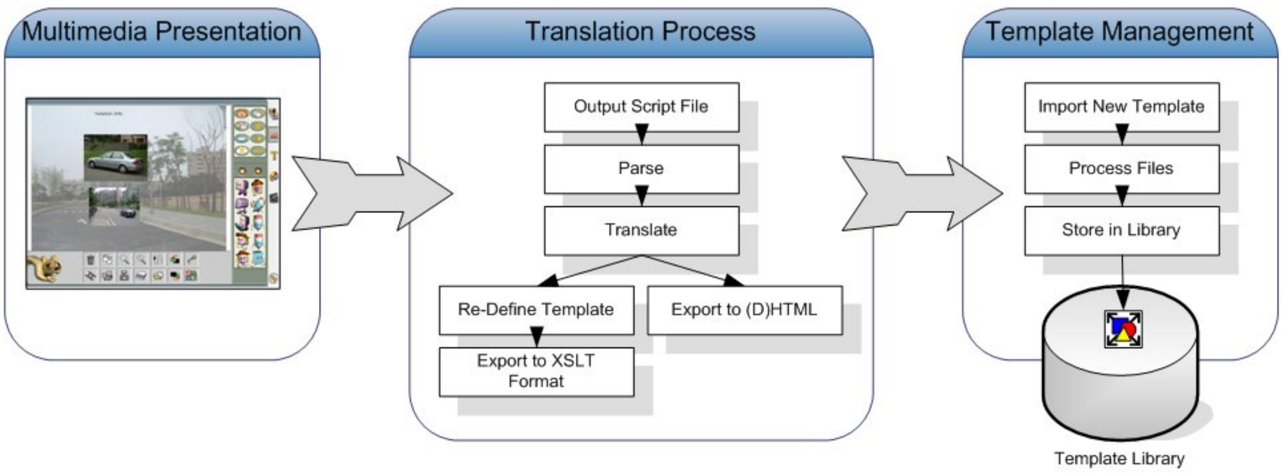

Figure 21. **The translation process**

Figure 21 shows how the translation process interacts with the Template Management Service. The output script file (depicted in Figure 22) contains general information about the template such as typical scene information, title, background image, animation effects, background music, background color, dimensions and so forth. Multimedia elements are listed as actors, actors have a specific type (animation, images, video, audio and text) as well as several attributes related to the type, for example, text actors contain font size, font style, font encoding, font effect attributes and so on. This script file is parsed and translated to support XML and XSLT tags. They are then re-defined, by replacing actual multimedia elements created in the Authoring Tool with *anchors*. Anchors are defined as placeholders for customization or replacement. In addition, the process embeds the structure of the template, the semantic relationships between multimedia elements and rules for customization.

```
[SystemInfo]
  ScriptVersion = "2.0"
  Application = "Multimedia Authoring Tool"
[End_SystemInfo]

[SCENEINFO]
  Title = "MobileCampus"
  BackgroundImage = "@Sc550.jpg"
  BackgroundMusic = ""
  PreludeEffect = 0
  PlayMode = 2
  Width = 640
  Height = 460
  TextureImage = ""
  BackgroundColor = 0
[END_SCENEINFO]

[CASTDEFINE]
[CAST] MCText
Begin
  {...}
End

[CAST] McMovie
Begin
  {...}
End

[CAST] MCAnim
Begin
  {...}
End
[END_CASTDEFINE]
```

Figure 22. **Sample multimedia script file**

The result of the translation process is an XSLT template that is specifically designed to support the proposed system’s template format. Figure 23 depicts a blueprint of the resultant template.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" >
  <xsl:template match="usertemplate/template">
            {…}
   <!-- image anchor definition -->
   <xsl:value-of select="imagelocation"/>
            {…}
   <!-- unique id -->
   <xsl:value-of select="@id"/
            {…}
   <!-- text anchors -->
   <xsl:value-of select="registration"/>
   <xsl:value-of select="date"/>
   <xsl:value-of select="time"/>
   <xsl:value-of select="location"/>
   <xsl:value-of select="comments"/>
   <xsl:value-of select="officername"/>
   <xsl:value-of select="officerphone"/>
   <xsl:value-of select="officeremail"/>
            {…}
   <!-- video anchors (AVI, WMV) -->
   <xsl:value-of select="videolocation1"/>
   <xsl:value-of select="videolocation2"/>
            {…}
   <!-- sound anchor (WAV) -->
   <xsl:value-of select="voicelocation"/>
            {…}
  </xsl:template>
</xsl:stylesheet>
```

Figure 23. **Sample XSLT blueprint**

The Transformation Process

The transformation process or content aggregation process describes how the data, content, layout and style are dynamically aggregated at run-time from disparate data sources and rendered into a single web presentation.

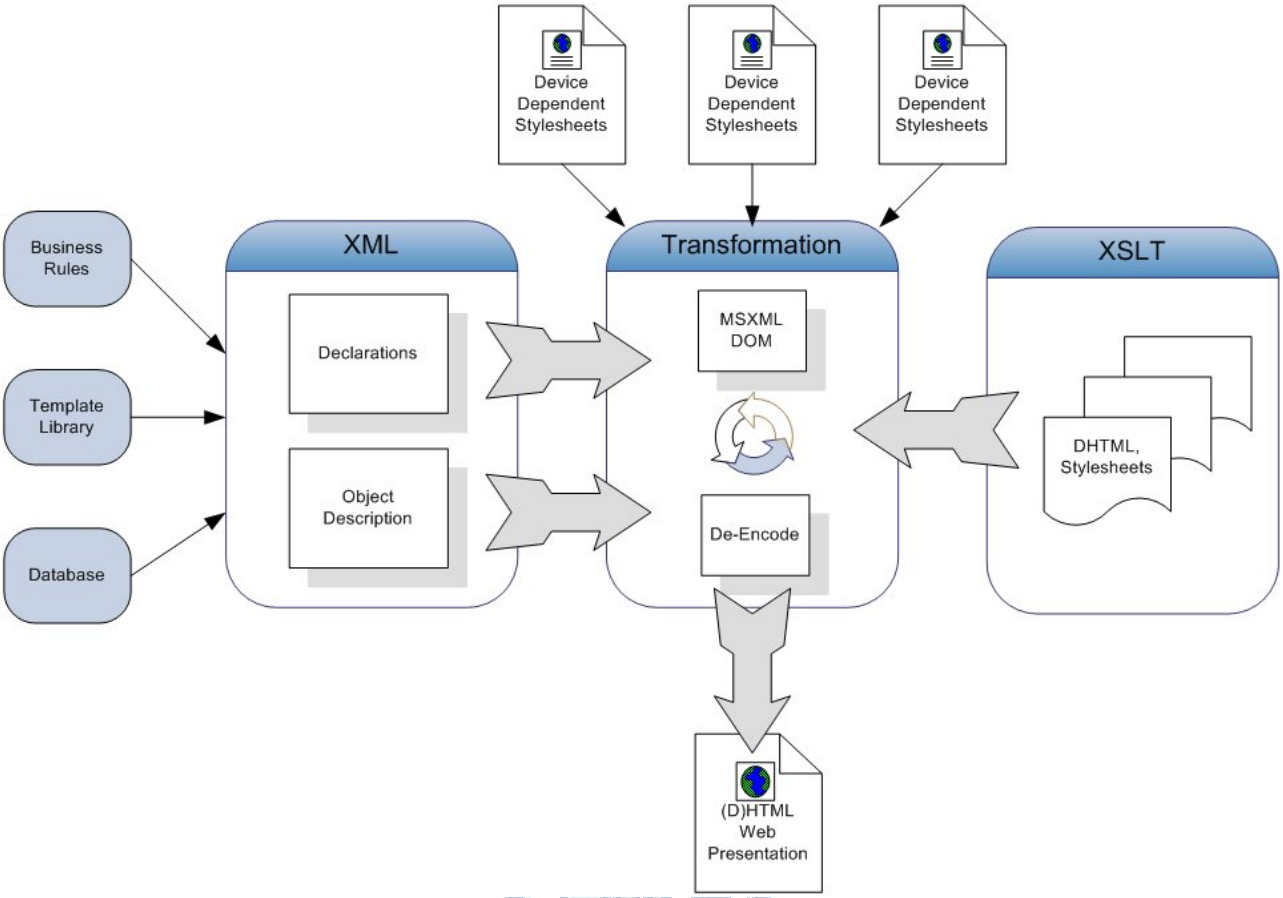

Figure 24. **The transformation or content aggregation process**

As depicted in Figure 24, the XML contains information such as the business rules, information about the templates and media. These are transformed using the XSLT template and CSS (using the Mobile CSS 1.0 standard) containing device profile information and presented on the Web as HTML.

3.4 Scenario Demonstrations

In this section we present two scenarios to demonstrate the practicability of the proposed system. We describe the process of each step from a high and low level view, and depict the process with screenshots where applicable.

3.4.1 Scenario One

In the first scenario, we demonstrate the use of pre-defined templates. In this scenario, we envision the following situation:

A car parked in a "no-parking" zone outside the library is spotted by a campus officer. He takes out his device, attaches his camera and records a video, takes a snapshot and makes a voice recording. In addition, he writes down some comments about the violation. Satisfied, he selects a pre-defined template, connects to the server, and transfers the files. He receives confirmation once his files have been uploaded. Because the user is a registered vehicle owner, the campus officer receives his contact details and violation number and issues the ticket, at the same time, the server automatically e-mails the vehicle's owner to inform him or her that their vehicle is in violation. The owner then views his or her violation on the MobileCampus portal

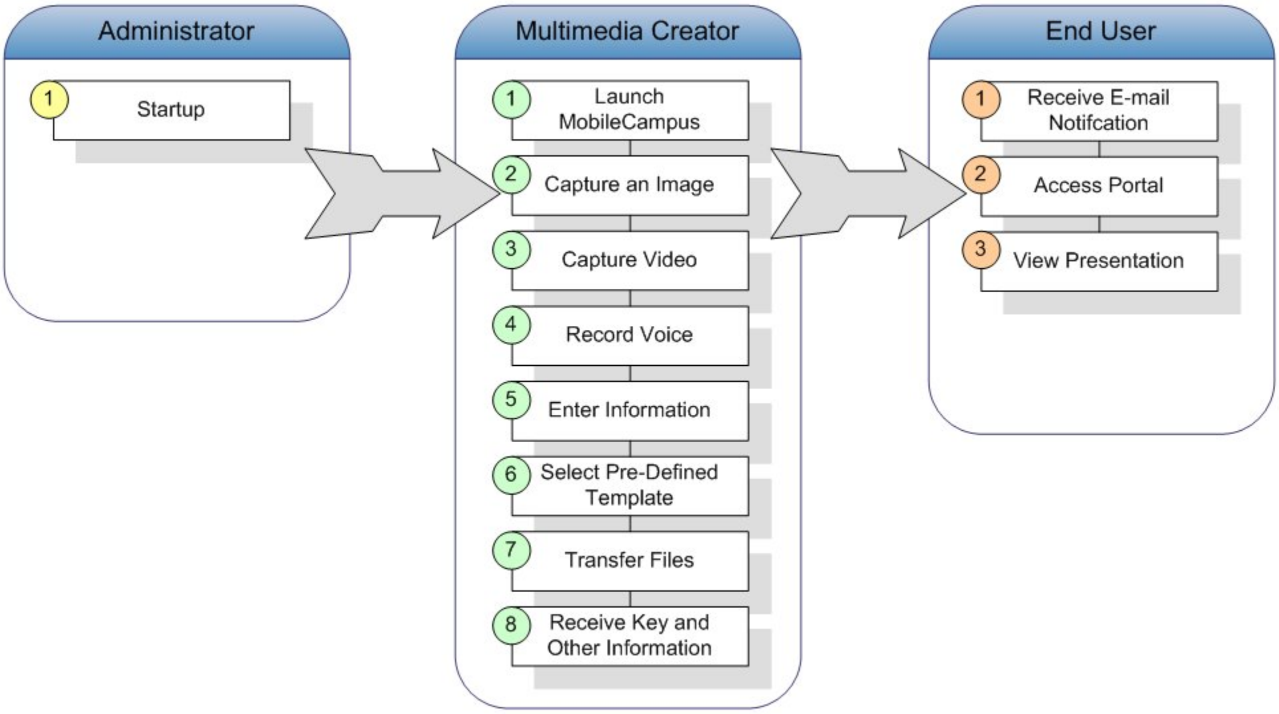

Figure 25. **Scenario One**

An outline of this scenario is depicted above in Figure 25. A detailed description of this transaction is elaborated as follows:

Table 2. **The first four steps of the transaction**

Steps	High-Level View	Low-Level View
Launch MobileCampus	The campus officer launches MobileCampus and accesses the Multimedia Capturing Service	None
Capture an Image	The “still image” radio button is selected and the campus officer clicks the “capture” button	The image is stored locally as a JPEG image
Capture Video	The “movie” radio button is selected and the campus officer clicks the “record” button. When the officer has completed recording the video, he clicks the “stop” button. Any adjustments to lighting, brightness and size are made here	Because the current camera API does not support built-in video compression, images are saved as JPEGS at a rate of five images per second. Images are compressed and packed into ZIP format in the background
Record Voice	The officer uses the voice recorder to record voice	The voice recording is stored locally as a wave file and compressed and packed into ZIP format in the background

Table 2 and Table 3 describe the first five steps of the transaction (The Multimedia Capturing Service). Figure 26 and Figure 27 illustrate these steps from a high level viewpoint.

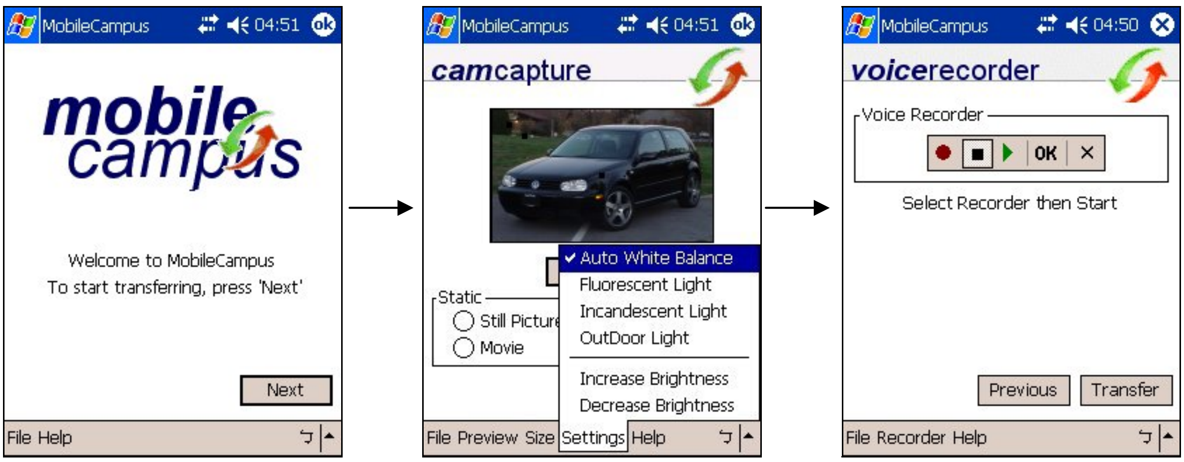

Figure 26. **The first four steps of the transaction (screenshots)**

Table 3. **The fifth step of the transaction**

Steps	High-Level View	Low-Level View
Enter Information	The officer enters the necessary information such as the time, location, nature of violation, comments and so forth	This information is stored locally as text

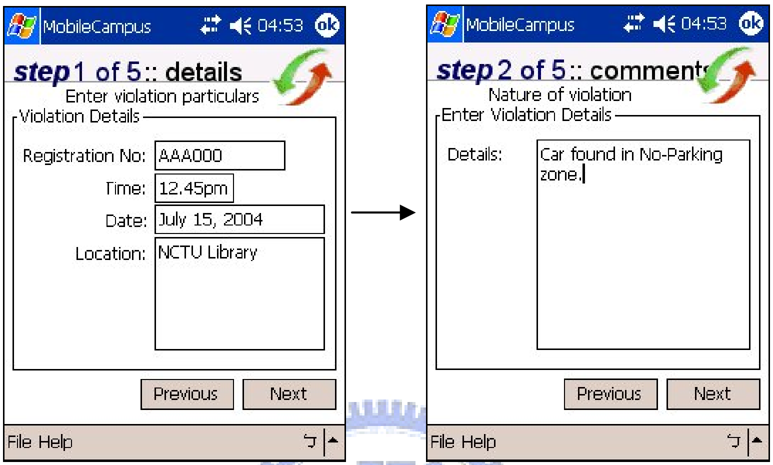

Figure 27. **The fifth step of the transaction (screenshots)**

Table 4 and Figure 28 describe and illustrate steps six to eight respectively.

Table 4. **Steps six to eight of the transaction**

Steps	High-Level View	Low-Level View
Select Pre-Defined Template	The officer selects a pre-defined template	Information about the template selection is stored locally
Transfer Files	Once the officer has completed acquiring the multimedia and is satisfied with his template selection, he connects to the server. The file transferring process is automatic	The client connects to the server via 802.11b/g using the File Transfer Protocol (FTP). The user is first authenticated and verified. If the verification is successful, a unique key is generated for the transaction. All acquired compressed multimedia information is then transferred to the server. During this time, information about the transaction is stored in XML. Acquired multimedia files are

		unpacked and stored on the server in the media library (via the Template Management Service). The server also connects to the vehicle register to check if the owner exists in the database. If the owner exists, an e-mail is generated and sent automatically to the owner
Receive Key and Other Information	When the transaction is complete, the officer will receive confirmation. All information about the owner, details about the transaction and the unique key are sent back to the officer for reference	If queries to the vehicle register are successful, this information is sent back to the officer

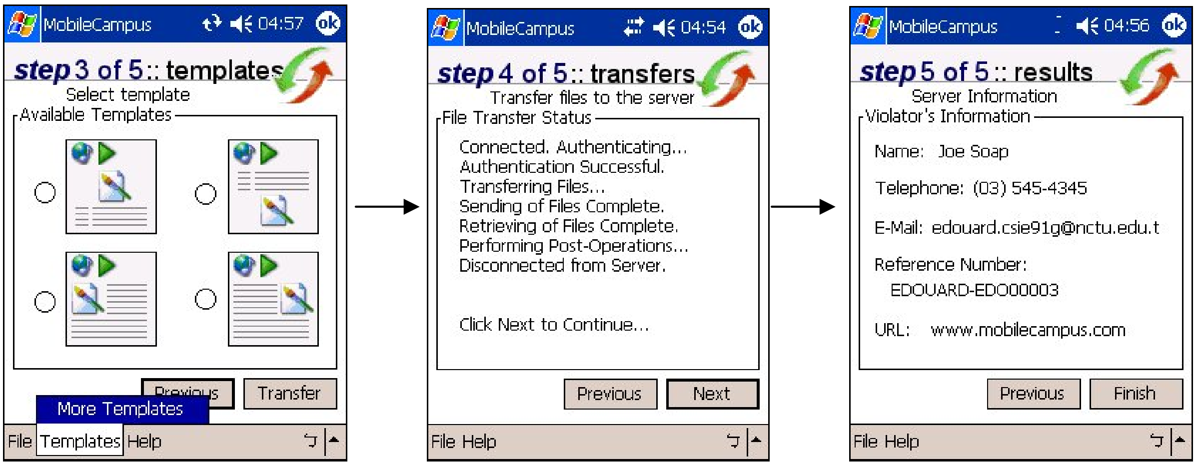

Figure 28. **Steps six to eight of the transaction (screenshots)**

The end user (vehicle owner) steps are described in Table 5 and depicted in Figure 29 respectively.

Table 5. **The end user steps of the transaction**

Steps	**High-Level View**	**Low-Level View**
Receive E-mail notification	The end user receives an e-mail on his or her pervasive computing device (or desktop PC)	The E-mail Service, when called by the File Transfer Service sends the e-mail to the recipient
Access Portal	The end user accesses the MobileCampus portal	The Web Presentation Service loads the portal. Depending on the device accessing the portal, the Web Presentation Service

		formats the portal accordingly using CSS and device profiling
View Presentation	The end user enters the key sent to him or her via e-mail, and views the presentation	Once the server receives the key, the transformation process and device customization process takes place for presentation to the end user

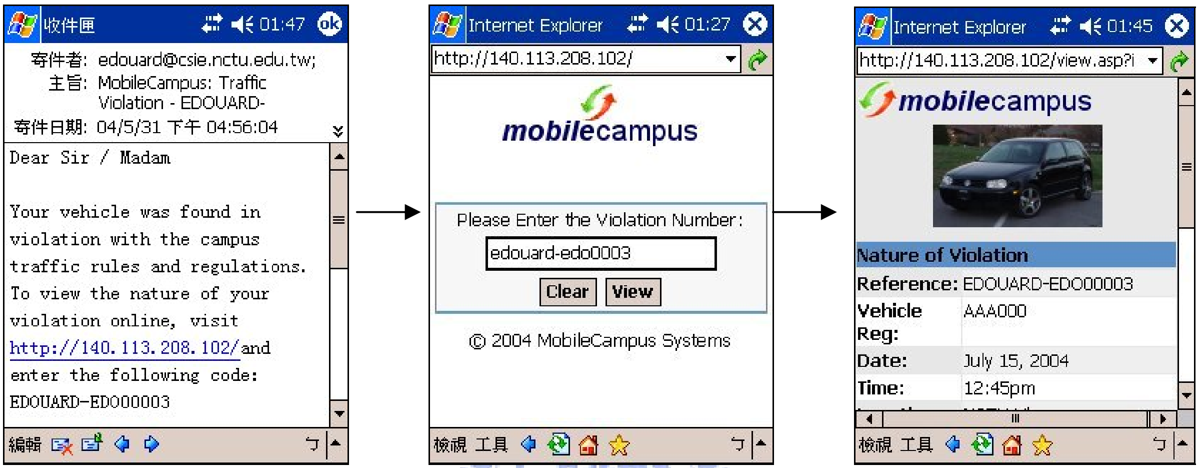

Figure 29. **The end user steps of the transaction (screenshots)**

3.4.2 Scenario Two

The second scenario is similar to the first with one difference. We demonstrate the integration of the Multimedia Visual Authoring Tool with the proposed system to create customized templates. In this scenario, we envision the following situation:

The system administrators and campus officers decided to add additional templates to facilitate more choice. They open the Multimedia Visual Authoring Tool to design a few templates and import them into the Template Management System. A campus officer on the field spots a speeding car. He takes out his device, attaches his camera and records a video, takes a snapshot and makes a voice recording. In addition, he writes down some comments about the violation. Satisfied, he decides to see what other templates are available, so he connects to the server and previews some of the *templates. He selects a template, connects to the server, and transfers the files. He receives confirmation once his files have been uploaded. Because the user is a registered vehicle owner, the campus officer receives his contact details and violation number for reference, at the same time, the server automatically e-mails the vehicle's owner to inform him or her that their vehicle is in violation. The owner then views his or her violation on the MobileCampus portal*

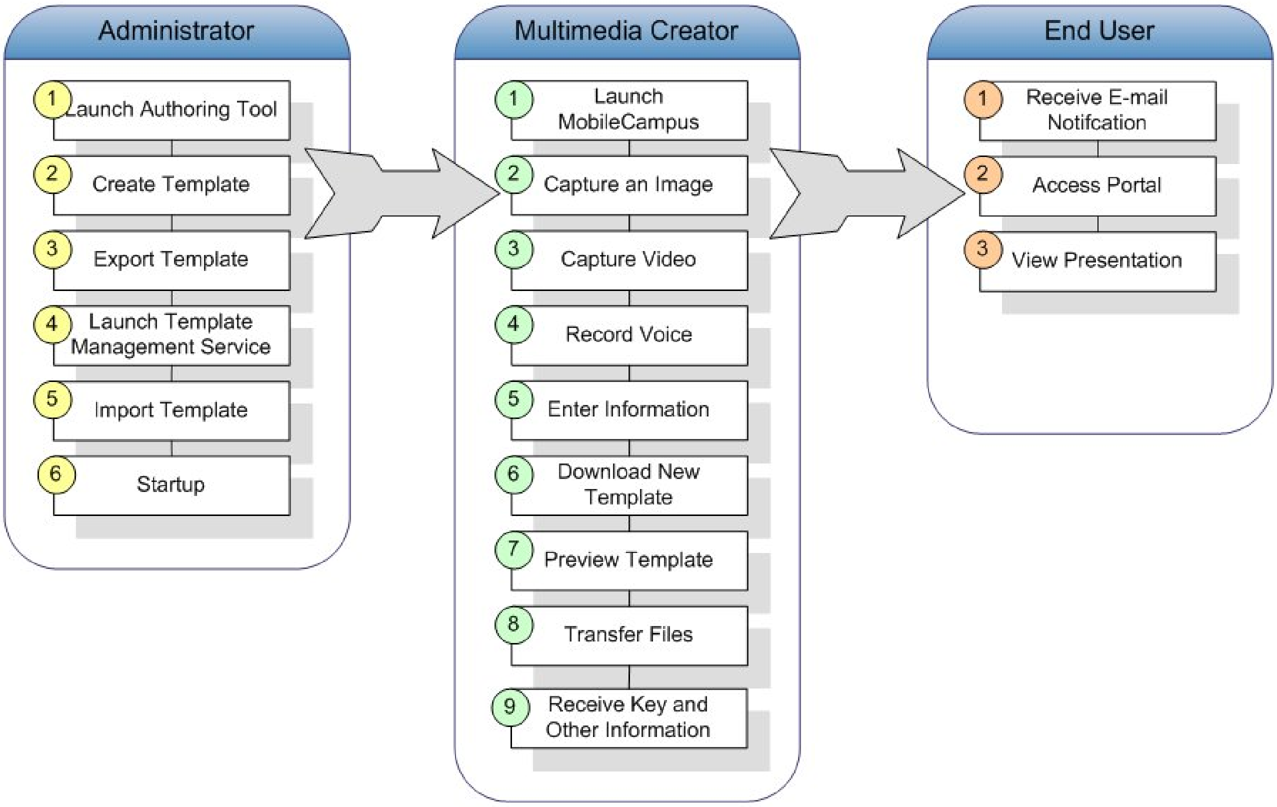

Figure 30. **Scenario Two**

Figure 30 outlines this scenario. A detailed description of this transaction is elaborated as follows:

Table 6. **The administrator steps of the transaction**

Steps	**High-Level View**	**Low-Level View**
Launch the Multimedia Visual Authoring Tool	The campus officer or system administrator launches the Multimedia Visual Authoring Tool	None
Create Template	The user designs and creates a	The multimedia template is

	template and saves it	saved into a script file that contains information about the template
Export Template	Along with other types of exporting facilities, one functionality integrated into the Multimedia Visual Authoring Tool is support for XSLT, the user selects this format for device support	The script file is parsed and translated into XSLT. Multimedia tags are replaced with anchors for customization, and additional criteria is embedded into the template during this process
Launch the Template Management Service	The user accesses the Template Management Service	None
Import Template	The user imports the template into the template library	The template is copied and stored in the template library

Table 6 (above) describes the administrator process.

Table 7. **The first four steps of the transaction**

Steps	**High-Level View**	**Low-Level View**
Launch MobileCampus	The campus officer launches MobileCampus and accesses the Multimedia Capturing Service	None
Capture an Image	The “still image” radio button is selected and the campus officer clicks the “capture” button	The image is stored locally as a JPEG image
Capture Video	The “movie” radio button is selected and the campus officer clicks the “record” button. When the officer has completed recording the video, he clicks the “stop” button. Any adjustments to lighting, brightness and size are made here	Because the current camera API does not support built-in video compression, images are saved as JPEGS at a rate of five images per second. Images are compressed and packed into ZIP format in the background
Record Voice	The officer uses the voice recorder to record voice	The voice recording is stored locally as a wave file and compressed and packed into ZIP format in the background

Table 7 and Table 8 describe the first five steps of the transaction which are both very similar to the first scenario. Figure 31 and Figure 32 illustrate these steps from a high level viewpoint.

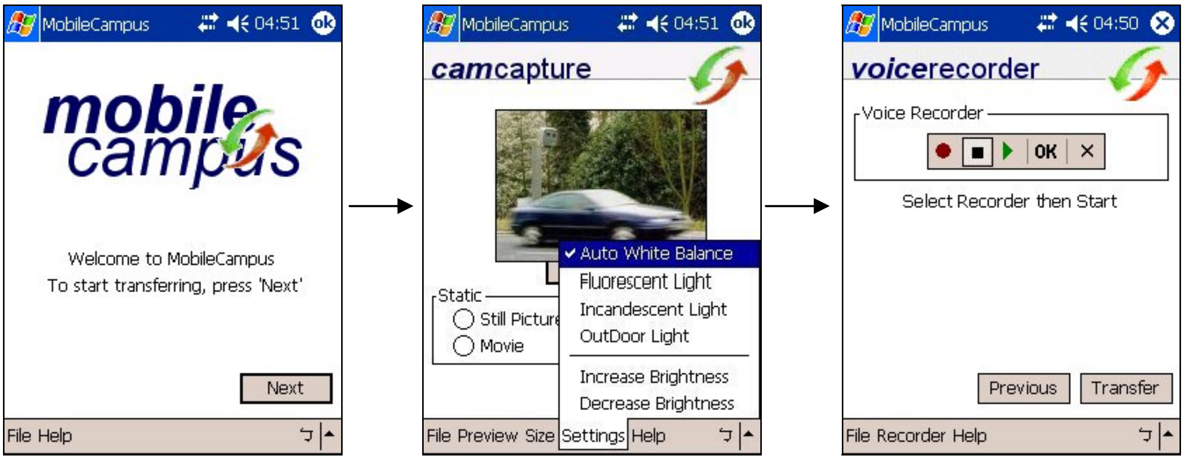

Figure 31. **The first four steps of the transaction (screenshots)**

Table 8. **The fifth step of the transaction**

Steps	High-Level View	Low-Level View
Enter Information	The officer enters the necessary information such as the time, location, nature of violation, comments and so forth	This information is stored locally as text

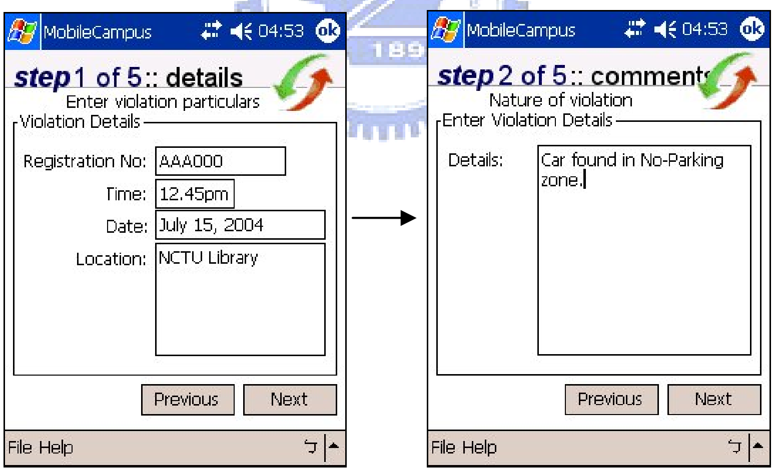

Figure 32. **The fifth step of the transaction (screenshots)**

Table 9. **Steps six to nine of the transaction**

Steps	High-Level View	Low-Level View
Download a New Template	The officer connects to the server to download new templates	The File Transfer Service is called which connects to the Server to check for new templates. A list of templates is

		downloaded and stored on the device
Preview Template	The officer selects a template and previews it using the built-in web browser	The Web Presentation Service identifies the template and dynamically assembles the template for preview through the client web browser
Transfer Files	Once the officer has completed acquiring the multimedia and is satisfied with his template selection, he connects to the server. The file transferring process is automatic	The client connects to the server via 802.11b/g using the File Transfer Protocol (FTP). The user is first authenticated and verified. If the verification is successful, a unique key is generated for the transaction. All acquired compressed multimedia information is then transferred to the server. During this time, information about the transaction is stored in XML. Acquired multimedia files are unpacked and stored on the server in the media library (via the Template Management Service). The server also connects to the vehicle register to check if the owner exists in the database. If the owner exists, an e-mail is generated and sent automatically to the owner
Receive Key and Other Information	When the transaction is complete, the officer will receive confirmation. All information about the owner, details about the transaction and the unique key are sent back to the officer for reference	If queries to the vehicle register are successful, this information is sent back to the officer

Table 9 describes steps six to nine. Figure 33 and Figure 34 illustrate steps six and seven and steps eight and nine respectively.

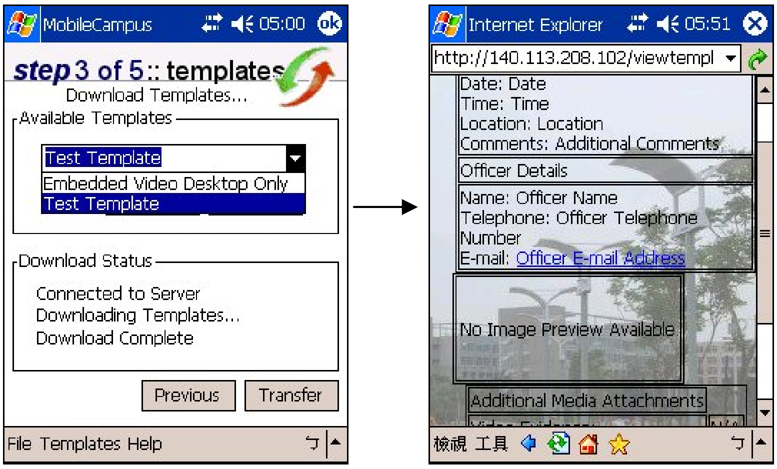

Figure 33. **Steps six and seven of the transaction (screenshots)**

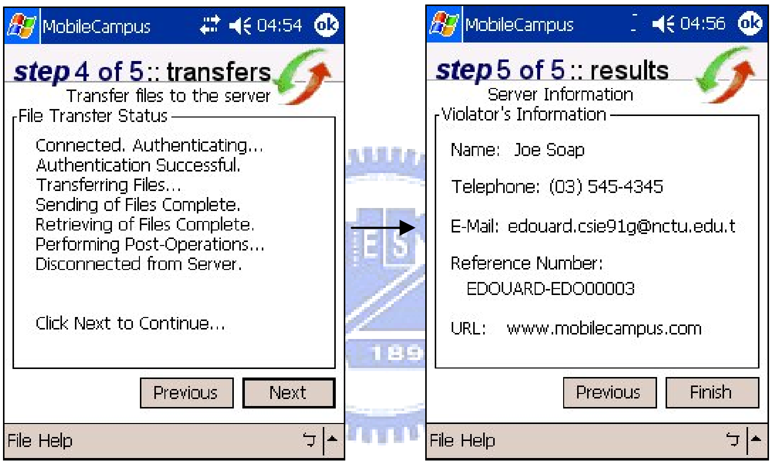

Figure 34. **Steps eight and nine of the transaction (screenshots)**

The end user process is identical to the first scenario. The end user (vehicle owner) steps are described in Table 10 and depicted in Figure 35 respectively.

Table 10. **The end user steps of the transaction**

Steps	**High-Level View**	**Low-Level View**
Receive E-mail notification	The end user receives an e-mail on his or her pervasive computing device (or desktop PC)	The E-mail Service, when called by the File Transfer Service sends the e-mail to the recipient
Access Portal	The end user accesses the MobileCampus portal	The Web Presentation Service loads the portal. Depending on

		the device accessing the portal, the Web Presentation Service formats the portal accordingly using CSS and device profiling
View Presentation	The end user enters the key sent to him or her via e-mail, and views the presentation	Once the server receives the key, the transformation process and device customization process takes place for presentation to the end user

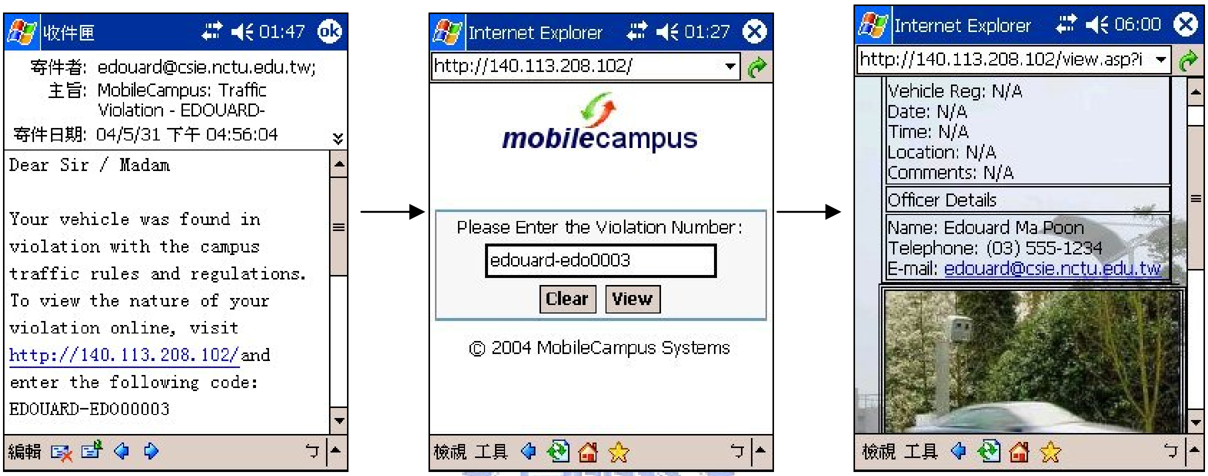

Figure 35. **The end user steps of the transaction (screenshots)**

3.5 Summary

MobileCampus can be divided into six services that communicate dependently and independently of one another: The Multimedia Capturing Service, The File Transfer Service, The Template Management Service, The Template Creation Service, The Content Management Service and The Internet Service. This chapter discussed two important processes, the translation process and transformation process that facilitate the integration of the Multimedia Visual Authoring Tool and the delivery of presentations to devices respectively. Two scenarios were constructed to demonstrate the practicability of MobileCampus and support its applicability.

The following chapter presents an analysis and evaluation of the proposed system against the specific criteria set out in Chapter Two.

CHAPTER FOUR

Discussion and Evaluation

4.1 Introduction

The previous chapter discussed the proposed system, MobileCampus, in detail. This chapter presents a discussion and evaluation of the proposed system. We attempt to critically evaluate the system against the specific criteria identified earlier in Chapter Two, particularly addressing the application challenges, the characteristics of mobile users and the limitations imposed by devices as well as how the proposed system contributes towards the vision of pervasive computing and enhances the user experience. Based on this evaluation, some limitations are identified and discussed. This evaluation forms the basis for a discussion and major contributions of the project and opportunities for future work, which is handled in the following chapter.

4.2 Criteria for Evaluation

Earlier, in Chapters One and Two, we laid the foundation, scope and criteria from which we intended to evaluate the proposed system. In three general aspects, we qualitatively evaluate the proposed system in terms of:

- How it addresses specific application challenges and supports the vision of pervasive computing

- How it enhances the user experience by taking into account the characteristics of users that will use such an application and the device limitations by working around them to achieve desired outcomes

We believe the proposed system achieved or partially achieved the vision of pervasive computing and enhanced the user experience, not only in terms of meeting the specific challenges and taking into account user characteristics and device limitations, but also in terms of allowing users to access information anytime, anywhere and on-demand within the computing environment. The proposed system particularly contributes to pervasiveness in terms of:

- **Productivity**. Pervasiveness can turn able to turn idle time into productive time by allowing mobile users to access timely resources while reducing paperwork.
- **Quality**. Access to resources reduces the inconvenience to users.
- **Portability**. The proposed system allows users to connect to resources from anywhere at anytime and on-demand.
- **Centralized Data**. By centralizing data on back-end servers, mobile users are reduced of the burden of unsynchronized data, and offered accurate, up-to-the-minute analysis of information.

Table 11 summarizes the system evaluation in terms of the application challenges set out in Chapter Two.

Table 11. **Summary of evaluation against the application challenges**

Challenge	System Evaluation
Automation	We emphasize a thin-client approach by transferring intensive processing duties to backend services (the File Transfer Service, Template management Service, Template Creation Service, Content Management Service and Internet Service) to assist with automation responsibilities to relieve end users of several tasks and saving processing power on devices (addressing device limitations). It also assists users with little or no experience working with multimedia presentations and hides system

	complexity
Content Aggregation	We support content aggregation using XML/XSLT. Content aggregation works well in a pervasive computing environment that demands information at run-time and supports heterogeneous devices.
Customization and personalization	We allow users to customize their experience. The integration of the Multimedia Visual Authoring Tool allows users to customize content and import them into the template library. In a pervasive computing environment, tasks should be completed minimally with little effort and time required.
Multi-device capability	The dynamic nature of templates reduced the problem of device heterogeneity by allowing the storing of device profiles and catering for future devices.
Web content and custom applications	We distributed our applications and converged web content with custom applications taking advantage of both browser facilities and custom applications on the pervasive device to offer access to a rich application and content delivery environment.
Integration of Services	MobileCampus is a system that is divided into several services that communicate together to achieve a common system goal across different platforms and protocols.

We found that the implementation of templates addresses the mobile users' characteristics by assisting users to create powerful multimedia presentations from their devices, relieving them of intensive task duties and limitations posed by these devices. In particular, we found that templates enhance the user experience in the following ways:

- They empower users by allowing them to perform tasks efficiently and effectively because they act as starting points for building powerful multimedia presentations minimizing the time needed to perform tasks. Minimum navigation and interaction with the device and real-time access to information creates fewer steps and fewer problems while minimizing errors and increasing data accuracy.
- They help reduce errors by transferring intensive tasks to back-end servers
- They help reduce paperwork and increase accuracy
- They address device limitations such as screen size, screen resolution and color depth by storing device profiles
- They assist preserving power and power consumption on devices by transferring demanding

tasks to back-end servers

- The choice of an open standards approach allows easy deployment, support of multiple devices, networks and back-end systems.

In addition, MobileCampus' enables a simplified, independent client representation layer that optimizes the user experience with minimal keystrokes and providing step-by-step navigation. Back-end integration hides the complexity of the process simplifying the user task.

4.3 Limitations

The translation and transformation process is a complicated process, not only in terms of supporting multiple pervasive computing device constraints and limitations, but also in terms of supporting an entire presentation script file (i.e. more than one scene). Currently, the template translation is limited to one scene and single multimedia elements of each type.

In addition, resulting formatting of presentations is not always ideal and needs to be addressed carefully.

4.4 Summary

After an extensive qualitative evaluation, it was established that the proposed system generally achieves its goals well, in terms of supporting the vision of pervasive computing and enhancing the user experience by addressing the application challenges, mobile user characteristics and device limitations. However, certain limitations were identified such as the inability to support entire multimedia presentations but only a single scene and the formatting results of presentations on different devices.

The following chapter discusses the contributions of this research project and provides a basis for future research work.

CHAPTER FIVE

Conclusions and Future Research

5.1 Research Summary

This research addressed the problem of assisting users perform tasks (such as create powerful multimedia presentations) efficiently and effectively by introducing dynamic web-based multimedia templates (DWMT). In order for a pervasive computing application to be successfully developed and deployed, a number of challenges need to be addressed. From an application-centric viewpoint, we are particularly interested in specific application challenges, characteristics of mobile users and the device limitations. In addition, we are concerned about the contribution towards the vision of pervasive computing as well as enhancing the user experience in such an environment. The proposed system attempts to consider these factors as important to the successful development and deployment of the system. These factors are also used as metrics from which we qualitatively evaluated the system.

5.2 Major Contributions of this Research

The growth of pervasive computing devices has been exponential, and there is little doubt that the need for creating pervasive computing applications that empower mobile users is an important issue in a mobile environment that is scarce of resources, obliterated by heterogeneity and demanding in terms of efficiency and effectiveness.

We investigated and identified certain challenges for developing and deploying successful applications in a pervasive computing environment and established some specific criteria that should be addressed from an application-centric viewpoint that supports the vision of pervasive computing and enhances the user experience. We studied the characteristics of the mobile user, and analyzed the attributes of successful pervasive computing applications. In addition, we investigated some approaches to templates. We then designed and implemented a system that demonstrated the use of dynamic web-based multimedia templates for assisting users of pervasive computing devices. We evaluated the system qualitatively in terms of its performance against the criteria mentioned earlier and how it supports the vision of pervasive computing and enhances the user experience. Major contributions of this project include:

- Assisting mobile users with little time and resources at their disposal to create powerful presentations or reports in a quick and efficient manner directly from their devices with minimal effort and skill required.
- Offering users with little experience or programming ability the opportunity to create powerful multimedia presentations from their devices.
- Contributing towards pervasiveness and enhancing the user experience by addressing specific application challenges associated, mobile user characteristics and device limitations with the vision of pervasive computing and dynamically presenting them to users to support the heterogeneity of devices and future devices

5.3 Future Research

Owing to the dynamic nature of the computing field, especially in wireless computing, the increasing technological advances and increasing demands of users for more effective and efficient interaction will guide future research. The reduction in size of mobile devices and implementation of invisible and embedded devices are becoming a reality. This will introduce a shift in the computing paradigm where the user tasks will ultimately become an important aspect of computing, and several attempts to design devices that meet their needs will proliferate.

This work on this is ongoing and further investigations need to be carried out in both the research and development areas, specifically in the delivery format quality of the presentations and support for new devices as well as a semantic evaluation of the system in a real-world environment. We feel that once the prototypes are considered successful, there is potential for several more applications in this environment and beyond.

5.4 In Closing

The focus of this research has been on proposing the use of dynamic web-based multimedia templates to empower users of pervasive computing devices and how it might fit in a pervasive computing environment by addressing, from an application-centric viewpoint, specific criteria such as the application challenges, mobile user characteristics and device limitations while considering the vision of pervasive computing and enhancing the user experience. The application of MobileCampus is not limited and can be conceived to be a general application for all multimedia data. The successful features of the proposed system rely heavily on its commitment to addressing the set criteria.

DEFINITION OF TERMS

CDMA The US military first use Code Division Multiple Access (CDMA) technology during World Ware II. CDMA encodes radio signals by using a random sequence to define a channel and convert speech into digital signals. It reportedly is more reliable, saves battery life, and is more secure than other wireless transmission technologies. QUALCOMM provided the hardware for the military during World War II and is now applying for patents on the technology that was made public after the war.

CSS An extension to HTML to allow styles, e.g. colors, font, size to be specified for certain elements of a hypertext document. Style information can be included in-line in the HTML file or in a separate CSS file (which can then be easily shared by multiple HTML files). Multiple levels of CSS can be used to allow selective overriding of styles.

GPRS General Packet Radio Service (GPRS) offers high-speed Internet service over a Global System for Mobile Communication (GSM) network. This process sends packets in bursts so that the user experience is instant connectivity, faster data transmission and faster response time for roaming users. It's easy to set up and easy to install.

GSM GSM was introduced in 1991 and came into service sometime in 1997. The packet technology provides high-speed wireless access over a GSM network for access by mobile devices and allows eight simultaneous calls per radio frequency. GSM is available in more than 100 countries, and the default service is available in Europe, Asia and Australia. GSM is also available in the Americas at the 1900mhz frequency.

IEEE 802.11 The IEEE 802.11 protocol is a standard in the wireless industry. It defines a physical layer and a sublayer that manages media access control (MAC). The protocol specifies two authentication methods. Open Systems authentication allows free access to the network and Shared Key authentication provides security through a prearranged signature.

IEEE 802.11b (Wi-Fi) The IEEE 802.11b protocol enhances and standardizes the physical layer so that it can support higher bit-rates, which allows wireless networking at higher speeds. This protocol supports bit-rates of 5.5mbps and 11mbps.

IEEE 802.1x The IEEE 802.1x protocol defines port-based, network access control used to provide authenticated network access for Ethernet (wired) and wireless networks. 802.1x support is included with Windows XP Professional.

IETF Internet Engineering Task Force (IETF) is an open organization designed to promote communication among network developers, architects, designers, and basically anyone with an interest in promoting well-designed Internet applications and efficient development for Internet tools and applications.

ITU International Telecommunication Union (ITU) is located in Geneva, Switzerland, and works with the United Nations to establish standards for global telecommunication networks and services. Its purpose is to act as a free international agent to work with governments to establish telephony and wireless standards worldwide.

Pervasive Computing The idea that technology is moving beyond the personal computer to everyday devices with embedded technology and connectivity as computing devices become progressively smaller and more powerful. Also called ubiquitous computing, pervasive computing is the result of computer technology advancing at exponential speeds -- a trend toward all man-made and some natural products having hardware and software. Pervasive computing goes beyond the realm of personal computers: it is the idea that almost any device, from clothing to tools to appliances to cars to homes to the human body to your coffee mug, can be imbedded with chips to connect the device to an infinite network of other devices. The goal of pervasive computing, which combines current network technologies with wireless computing, voice recognition, Internet capability and artificial intelligence, is to create an environment where the connectivity of devices is embedded in such a way that the connectivity is unobtrusive and always available.

TDMA Time Division Multiple Access (TDMA) technology is used to transmit digital packets from a cell phone to a base station Access Point. TDMA works by breaking transmissions into smaller chunks and then stacking them into shorter time units so that more calls can be sent simultaneously. GSM is using TDMA to provide the eight calls per frequency mentioned earlier.

Wi-Fi Alliance The Wi-Fi Alliance is a non-profit international association formed in 1999 to certify the interoperability of WLAN products based on the IEEE 802.11 specification. The goal of the Wi-Fi Alliance’s members is to enhance the user experience through product interoperability.

WLAN Wireless Local Area Networks (WLANs) provide wireless network access in a corporate environment. With roaming wireless connections, users can move from building to building and from room to room without disruption of service. Two types of WLANs are available: infrastructure and ad-hoc networks. An infrastructure network connects individual PCs (known as stations) to a wired network via a wireless AP. Ad-hoc networks allow individual users to form a temporary wireless network for sharing and collaborating without the need for a wireless AP.

WMV Short for Windows Media Video, it is a Microsoft standard for delivering video over the Internet.

WPAN Wireless Personal Area Networks (WPANs) are designed to provide an individual with wireless connectivity within a personal space. This space surrounds the user up to 10 meters and provides an ad-hoc wireless network. Typically used for cell phones, laptops and PDAs. This ad-hoc network uses either infrared technology to "squirt" data to another device within 1 meter or takes advantage of Bluetooth technology.

WWAN Wireless Wide Area Networks (WWANs) are designed to establish wireless connections over a large geographic area. Due to the size of the areas that a WWAN must transmit, WWAN technologies transfer data by using satellites or multiple antenna sites that wireless service providers maintain. Although wireless manufacturers and developers are working toward a wireless world standard, there isn't one today. GSM is widely predominant throughout the world; however CDMA and its 1xRTT standard are also available.

XML XML is a markup language for documents containing structured information.
Structured information contains both content (words, pictures, etc.) and some indication of what role that content plays (for example, content in a section heading has a different meaning from content in a footnote, which means something different than content in a figure caption or content in a database table, etc.). Almost all documents have some structure. A markup language is a mechanism to identify structures in a document. The XML specification defines a standard way to add markup to documents.

XSLT Short for Extensible Style Language Transformation, the language used in XSL style sheets to transform XML documents into other XML documents. An XSL processor reads the XML document and follows the instructions in the XSL style sheet, then it outputs a new XML document or XML-document fragment. This is extremely useful in e-commerce, where the same data need to be converted into different representations of XML. Not all companies use the exact same programs, applications and computer systems. XSLT Recommendation was written and developed by the XSL Working Group and became ratified by the W3C on November 16, 1999

REFERENCE APPENDIX

[1] G. Cohen, "New Opportunities". International Business Machines. 2003. Available: http://www-1.ibm.com/partnerworld/pwhome.nsf/weblook/news_busopps_adkins_qanda.html

[2] G. Banavar, J. Beck, E. Gluzberg, J. Munson, J. Sussman and D. Zukowski, "Challenges: An Application Model for Pervasive Computing". *In Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom2000)*. Boston, Massachusetts. 2000. pp 266 – 274

[3] G. Couloris, J. Dollimore and T. Kindberg, "Distributed Systems Concepts and Design", 3rd Edition. Addison-Wesley, 2001.

[4] *Introduction to Pervasive Computing*. International Business Machines. 2003. Available: http://www-306.ibm.com/software/pervasive/business_partners/

[5] BestWise International Computing Company. Available: http://www.caidiy.com.tw

[6] Veo PhotoTraveler. Available: http://www.veo.com.tw

[7] Hewlett-Packard iPAQ H5555 PocketPC. Available: http://www.hp.com

[8] D. Saha and A. Mukherjee, "Pervasive Computing: A Paradigm for the 21st Century". *IEEE Computer Society*. March 2003.

[9] S. Acharya, "Application and Infrastructure Challenges in Pervasive Computing". Bell Laboratories, Lucent Technologies, Inc. 2002

[10] J. R. Smith, R. Mohan, and C. S. Li, "Scalable Multimedia Delivery for Pervasive Computing". *In Proceedings of the Seventh ACM International Conference on Multimedia (Part 1)*. Orlando, Florida. 1999. pp 131 – 140

[11] K. Henricksen, J. Indulska and A. Rakotonirainy, "Infrastructure for Pervasive Computing:

Challenges”. Distributed Systems Technology Centre, The Unviersity of Queensland, 2001.

[12] D. Bevis, “Extending Enterprise Applications to Mobile Users”. Websphere Everyplace Access, July 2002.

[13] *Mobilizing the Enterprise: Solutions for Automating the Everyday Tasks of Mobile Teams*. A Business Whitepaper. Orsus Solutions Limited. 2002.

[14] Webster’s Online Dictionary. Available: http://www.websters.com

[15] M. Satyanarayanan, “Pervasive Computing: Vision and Challenges”. *IEEE Personal Communications*. Vol. 8, No. 4. pp 10 – 17. August 2001.

[16] D. Saha, A. Mukherjee and S. Bandopadhyay, *Networking Infrastructure for Pervasive Computing: Enabling Technologies and Systems*. Kluwer Academic, 2002.

[17] E. Brewer *et al*. “A Network Architecture for Heterogeneous Mobile Computing”. *IEEE Personal Communications*. Vol 5, No.5. October, 1998

[18] D. Arnold *et al*. “Discourse with Disposable Computers: How and Why you will talk to your tomatoes”. *Usenix Workshop on Embedded Systems*, March 1999

[19] D. Gribble *et al*. “The Ninja Architecture for Robust Internet-Scale Systems and Services”. *Computer Networks*, *Special Issue on Pervasive Computing*, June 2000.

[20] B. Egan. “Making the Case for Wireless Mobility Investment”. Mobile Competency, 2003.

[21] K. Eustice. “A Universal Information Applicance”. *IBM Systems Journal*, Vol 38, No. 4, 1999.

[22] D. Norman. “The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the solution”. MIT Press, 1998.

[23] *Imagination. Innovation. Results.* IBM’s Enterprise Vision for Pervasive Computing. March 2002.

[24] M. Dertouzos. “The Future of Computing”. Scientific American. August, 1999.

[25] S. Midkiff. “Mobile and Wireless Networks and Applications”. *IEEE Pervasive Computing*.

October – December, 2002.

[26] National Express. Available: http://www.gobycoach.com

[27] The Waterbury Police Department. Available: http://www.wtbypd.org

[28] Lotte Sam Kang. Available: http://www.lotte.co.kr

[29] Kudos. Available: http://www.microsoft.com/australia/business/casestudy/studies/kudos.aspx

[30] Enning GmbH. Available: http://www.enning.de

[31] Carnegie-Mellon University's Aura Project. Available: http://www-2.cs.cmu.edu/~aura

[32] University of Washington's Portolano Project. Available: http://portolano.cs.washington.edu

[33] Hewlett-Packard's CoolTown Project. Available: http://www.cooltown.com

[34] University of California at Berkeley's Endeavour Project. Available: http://endeavour.cs.berkeley.edu

[35] Massachusetts Institute of Technology's Project Oxygen. Available: http://oxygen.lcs.mit.edu

[36] IBM Websphere Anyplace Project. Available: http://webspherеadvisor.com/doc/13254

[37] M. Butler, F. Giannetti, R. Gimson and T. Wiley. "Device Independence and the Web". *IEEE Internet Computing*. September – October 2002. pp 81 – 86.

[38] J-B. Minchelli. "XSL Device Formatting Using the Microsoft XSL ISAPI Filter". March 2002

[39] The Apache Cocoon Project. Available: http://cocoon.apache.org

[40] Composite Capabilities/Preferences Profile. Available: http://www.w3.org/Mobile/CCPP

[41] Microsoft Mobile Internet Toolkit. Available: http://www.microsoft.com

[42] W. Huang, M. O'Dea, and A. Mille, "ConKMeL: A Contextual Knowledge Management Framework to Support Intelligent Multimedia e-Learning". *In Proceedings of the IEEE Fifth International Symposium on Multimedia Software Engineering (ISMSE'03)*. 2003. pp 223 –

231.

[43] L. Bouthilher, “Synchronized multimedia on the Web”. *Web Techniques Magazine*, 3(9), September 1998

[44] Unwired Planet. Handheld device markup language specification. Technical Report Version 2.0, Unwired Planet, Inc., April 1997.

[45] O. Avaro, P. Chou, A. Eleftheriadis, C. Herpel, and C. Reader, “The MPEG-4 systems and description languages: A way ahead in audio visual information representation”. *Signal Processing: Image Communication, Special Issue on MPEG-4*, 4(9):385-431, May 1997

[46] J. Honeycutt and R. Lutsch. “Wireless Network for Mobility and Collaboration”. *Microsoft Corporation*. December 2002.

[47] V. Stanford. “Pervasive Computing Puts Food on the Table”. *IEEE Pervasive Computing*. January – March, 2003.

[48] P. Lipton. “Making your Mobile e-Business a Success using XSL and Web Services”. *Computer Associates*. October 2002.

APPENDIX I

Sample XML Transaction File

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- DTD Declarations-->
<!DOCTYPE usertemplate [
    <!ELEMENT usertemplate (template)*>
    <!ELEMENT template (registration, time, date, location, comments, officername,
officeremail, officerphone, imagelocation, videolocation1, videolocation2,
voicelocation, templatechoice)>
        <!ATTLIST template id CDATA #REQUIRED>
        <!ELEMENT registration (#PCDATA)>
        <!ELEMENT time (#PCDATA)>
        <!ELEMENT date (#PCDATA)>
        <!ELEMENT location (#PCDATA)>
        <!ELEMENT comments (#PCDATA)>
        <!ELEMENT officername (#PCDATA)>
        <!ELEMENT officeremail (#PCDATA)>
        <!ELEMENT officerphone (#PCDATA)>
        <!ELEMENT imagelocation (#PCDATA)>
        <!ELEMENT videolocation1 (#PCDATA)>
        <!ELEMENT videolocation2 (#PCDATA)>
        <!ELEMENT voicelocation (#PCDATA)>
        <!ELEMENT templatechoice (#PCDATA)>
]>
<usertemplate>
    <template id="EDOUARD-EDO00001">
        <registration>aaa000</registration>
        <time>N/A</time>
        <date>N/A</date>
        <location>N/A</location>
        <comments>N/A</comments>
        <officername>Edouard Ma Poon</officername>
        <officeremail>edouard@csie.nctu.edu.tw</officeremail>
        <officerphone>NA</officerphone>
        <imagelocation>./users/edouard/EDO00001/image.jpg</imagelocation>
        <videolocation1>./users/edouard/EDO00001/video.avi</videolocation1>
        <videolocation2>./users/edouard/EDO00001/video.wmv</videolocation2>
        <voicelocation>./users/edouard/EDO00001/voice.wav</voicelocation>
        <templatechoice>1</templatechoice>
    </template>
</usertemplate>
</xml>
```

APPENDIX II

Sample XSLT Template File

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform" >
  <xsl:template match="usertemplate/template">
    <center>
    <table>
      <tr>
        <td align="center">
          <xsl:choose>
          <xsl:when test="imagelocation='N/A'">
          <br/><br/>No Image Preview Available<br/><br/><br/>
          </xsl:when>
          <xsl:otherwise>

            <img>
            <xsl:attribute name="src"><xsl:value-of
      select="imagelocation"/></xsl:attribute>
            <xsl:attribute name="border">1</xsl:attribute>
            </img>
            </xsl:otherwise>
            </xsl:choose>
        </td>
      </tr>
    </table>
   <table>
      <tr><td class="head">Nature of Violation</td></tr>
    </table>
    <table cellspacing="1">
      <tr>
        <td class="cellhead">Reference:</td>
        <td class="cellcontents"><xsl:value-of select="@id"/></td>
      </tr>
      <tr>
        <td class="cellhead2">Vehicle Reg:</td>
        <td class="cellcontents2"><xsl:value-of select="registration"/></td>
      </tr>
      <tr>
        <td class="cellhead">Date:</td>
        <td class="cellcontents"><xsl:value-of select="date"/></td>
      </tr>
      <tr>
        <td class="cellhead2">Time:</td>
        <td class="cellcontents2"><xsl:value-of select="time"/></td>
      </tr>
      <tr>
        <td class="cellhead">Location:</td>
        <td class="cellcontents"><xsl:value-of select="location"/></td>
      </tr>
      <tr>
```

```
    <td class="cellhead2">Comments:</td>
    <td class="cellcontents2"><xsl:value-of select="comments"/></td>
  </tr>
</table>
   <table>
  <tr><td class="head">Officer Details</td></tr>
</table>
<table>
  <tr>
    <td class="cellhead">Name:</td>
    <td class="cellcontents"><xsl:value-of select="officername"/></td>
  </tr>
  <tr>
    <td class="cellhead2">Telephone:</td>
    <td class="cellcontents2"><xsl:value-of select="officerphone"/></td>
  </tr>
  <tr>
    <td class="cellhead">E-Mail:</td>
    <td class="cellcontents">
      <xsl:choose>
      <xsl:when test="officeremail='N/A'">
      <xsl:value-of select="officeremail"/>
      </xsl:when>
      <xsl:otherwise>
      <a><xsl:attribute name="href">mailto:<xsl:value-of
  select="officeremail"/></xsl:attribute><xsl:value-of
  select="officeremail"/></a>
      </xsl:otherwise>
      </xsl:choose>
    </td>
  </tr>
</table>
<table>
  <tr><td class="head">Additional Media Attachments</td></tr>
</table>
<table>

<tr>
<td class="cellhead">Video Evidence:
</td>
<td class="cellcontents">
  <xsl:choose>
  <xsl:when test="videolocation1='N/A'">
  <xsl:value-of select="videolocation1"/>
  </xsl:when>
  <xsl:otherwise>
  [ <a><xsl:attribute name="href"><xsl:value-of
  select="videolocation1"/></xsl:attribute>AVI (MPEG4)</a> | <a><xsl:attribute
  name="href"><xsl:value-of select="videolocation2"/></xsl:attribute>WMV</a> ]
  Format
  </xsl:otherwise>
  </xsl:choose>
</td>
</tr>
<tr>
<td class="cellhead2">Voice Commentary:
</td>
<td class="cellcontents2">
  <xsl:choose>
  <xsl:when test="voicelocation='N/A'">
  <xsl:value-of select="voicelocation"/>
```

```
      </xsl:when>
      <xsl:otherwise>
      [<a><xsl:attributename="href"><xsl:value-of
      select="voicelocation"/></xsl:attribute>Click</a>] to play
      </xsl:otherwise>
      </xsl:choose>
    </td>
    </tr>
    </table>
    </center>
  </xsl:template>
</xsl:stylesheet>
```

APPENDIX III

Sample CSS Device Profile File

```
body {
  font-family: verdana;
  font-size: 10pt;
}

table {
  font-family: verdana;
  font-size: 10pt;
  width: 250px;
  table-layout:fixed;
  cellspacing: 1;
  background: #E7E7E7
}

.tablecolumn {
  font-family: verdana;
  font-size: 10pt;
  width: 95px;
  table-layout:fixed;
  cellspacing: 1;
  background: #E7E7E7
}

img.imagesize {
  width: 90px;
}

.cellcontents2 {
  width: 184px;
  align: left;
  vertical-align: top;
  background: #FFFFFF
}

.cellhead2 {
  vertical-align: top;
  font-family: verdana;
  font-size: 10pt;
  font-weight: bold;
  width: 56px;
  background: #FFFFFF
}

.cellcontents {
  vertical-align: top;
  width: 184px;
  align: left;
  background: #EFEFEF
}

.cellhead {
  font-family: verdana;
```

```
  font-size: 10pt;
  font-weight: bold;
  width: 56px;
  background: #EFEFEF
  vertical-align: top;
}

.maincolumn {
  width: 240px;
}

.head {
  font-family: Verdana;
  font-size: 10pt;
  font-weight: bold;
  background: #5E8FC4
 }
```