

# Evaluation of Digital Identity using Windows CardSpace

Antonio J. Fernandez Sepulveda



Submitted in partial fulfillment of the requirements of  
Napier University for the degree of Master of Science  
in Advanced Software Engineering

School of Computing  
October 2008

## **Authorship declaration**

I, Antonio J. Fernandez Sepulveda, confirm that this dissertation and the work presented in it are my own achievement.

1. Where I have consulted the published work of others this is always clearly attributed;
2. Where I have quoted from the work of others the source is always given. With the exception of such quotations this dissertation is entirely my own work;
3. I have acknowledged all main sources of help;
4. If my research follows on from previous work or is part of a larger collaborative research project I have made clear exactly what was done by others and what I have contributed myself;
5. I have read and understand the penalties associated with Academic Misconduct.
6. I also confirm that I have obtained **informed consent** from all people I have involved in the work in this dissertation following the School's ethical guidelines

**Signed:**

**Date:**

**Matriculation no: 06016366**

## **Data Protection declaration**

Under the 1998 Data Protection Act, we cannot disclose your grade to an unauthorised person. However, other students benefit from studying dissertations that have their grades attached.

**Please sign your name against *one* of the options below to state your preference.**

The University may make this dissertation, with indicative grade, available to others.

The University may make this dissertation available to others, but the grade may not be disclosed.

The University may not make this dissertation available to others.

## Abstract

The Internet was initially created for academic purposes, and due to its success, it has been extended to commercial environments such as e-commerce, banking, and email. As a result, Internet crime has also increased. This can take many forms, such as: personal data theft; impersonation of identity; and network intrusions. Systems of authentication such as username and password are often insecure and difficult to handle when the user has access to a multitude of services, as they have to remember many different authentications. Also, other more secure systems, such as security certificates and biometrics can be difficult to use for many users. This is further compounded by the fact that the user does not often have control over their personal information, as these are stored on external systems (such as on a service provider's site).

The aim of this thesis is to present a review and a prototype of Federated Identity Management system, which puts the control of the user's identity information to the user. In this system the user has the control over their identity information and can decide if they want to provide specific information to external systems. As well, the user can manage their identity information easily with Information Cards. These Information Cards contain a number of claims that represent the user's personal information, and the user can use these for a number of different services. As well, the Federated Identity Management system, it introduces the concept of the Identity Provider, which can handle the user's identity information and which issues a token to the service provider. As well, the Identity Provider verifies that the user's credentials are valid.

The prototype has been developed using a number of different technologies such as .NET Framework 3.0, CardSpace, C#, ASP.NET, and so on. In order to obtain a clear result from this model of authentication, the work has created a website prototype that provides user authentication by means of Information Cards, and another, for evaluation purposes, using a username and password. This evaluation includes a timing test (which checks the time for the authentication process), a functionality test, and also quantitative and qualitative evaluation. For this, there are 13 different users and the results obtained show that the use of Information Cards seems to improve the user experience in the authentication process, and increase the security level against the use of username and password authentication.

This thesis concludes that the Federated Identity Management model provides a strong solution to the problem of user authentication, and could protect the privacy rights of the user and returns the control of the user's identity information to the user.

# Contents

<b>1</b>	<b>INTRODUCTION.....</b>	<b>11</b>
1.1	CONTEXT .....	11
1.2	AIM AND OBJECTIVES .....	11
1.3	BACKGROUND.....	11
1.4	THESIS LAYOUT.....	12
<b>2</b>	<b>LITERATURE REVIEW.....</b>	<b>14</b>
2.1	INTRODUCTION.....	14
2.2	AUTHENTICATION .....	14
2.2.1	<i>Username and Password Authentication.....</i>	<i>15</i>
2.2.2	<i>Biometric .....</i>	<i>15</i>
2.2.3	<i>Digital Certificates .....</i>	<i>17</i>
2.2.4	<i>Security Tokens.....</i>	<i>17</i>
2.3	FEDERATED IDENTITY MANAGEMENT (FIM) .....	18
2.4	CIRCLES OF TRUST IN FEDERATED SYSTEMS.....	19
2.5	OECD DATA PROTECTION PRINCIPLES .....	20
2.6	LAWS OF IDENTITY.....	21
2.6.1	<i>User Control and Consent .....</i>	<i>21</i>
2.6.2	<i>Minimal Disclosure for a Constrained Use .....</i>	<i>21</i>
2.6.3	<i>Justifiable Parties .....</i>	<i>22</i>
2.6.4	<i>Directed Identity.....</i>	<i>23</i>
2.6.5	<i>Pluralism of Operators and Technologies.....</i>	<i>24</i>
2.6.6	<i>Human Integration .....</i>	<i>25</i>
2.6.7	<i>Consistent Experience across Contexts.....</i>	<i>26</i>
2.7	USER-CONTROL IN FEDERATED IDENTITY MANAGEMENT.....	26
2.7.1	<i>Basic Properties .....</i>	<i>27</i>
2.7.2	<i>FIM System Properties .....</i>	<i>27</i>
2.7.3	<i>Transaction Properties.....</i>	<i>28</i>
2.7.4	<i>Identity Information Properties .....</i>	<i>28</i>
2.7.5	<i>Composite Properties .....</i>	<i>29</i>
2.8	FEDERATED IDENTITY MANAGEMENT MODELS .....	30
2.8.1	<i>Relationship-focused Model .....</i>	<i>30</i>
2.8.2	<i>Credential-focused Model .....</i>	<i>31</i>
2.8.3	<i>Relation with the real world.....</i>	<i>31</i>
2.9	STANDARDS FOR FEDERATED IDENTITY MANAGEMENT .....	32
2.9.1	<i>Security Assertion Markup Language.....</i>	<i>32</i>
2.9.2	<i>SAML Components .....</i>	<i>32</i>
2.9.3	<i>WS-Security .....</i>	<i>34</i>
2.9.4	<i>WS-Federation.....</i>	<i>36</i>
2.10	SAMPLE OF FEDERATED IDENTITY MANAGEMENT .....	37
2.11	CONCLUSION.....	38
<b>3</b>	<b>REQUIREMENTS ANALYSIS AND DESIGN.....</b>	<b>40</b>
3.1	INTRODUCTION.....	40
3.2	IDENTITY MANAGEMENT .....	40
3.2.1	<i>Identity Management Requirements .....</i>	<i>40</i>
3.2.2	<i>Identity Management Design.....</i>	<i>41</i>
3.3	DATABASE .....	42
3.3.1	<i>Database Requirements.....</i>	<i>42</i>

3.3.2	<i>Database Design</i> .....	43
3.4	COMMUNICATION.....	46
3.4.1	<i>Communication Requirements</i> .....	46
3.4.2	<i>Communication Design</i> .....	47
3.5	APPLICATION .....	49
3.5.1	<i>Application Requirements</i> .....	50
3.5.2	<i>Application Design</i> .....	53
3.6	IDENTITY PROVIDER.....	58
3.6.1	<i>Identity Provider Requirements</i> .....	58
3.6.2	<i>Identity Provider Design</i> .....	58
3.7	CONCLUSION.....	60
<b>4</b>	<b>IMPLEMENTATION</b> .....	<b>62</b>
4.1	INTRODUCTION.....	62
4.2	IMPLEMENTATION GRAPHIC SCHEMA .....	62
4.3	DATABASE IMPLEMENTATION .....	63
4.3.1	<i>SQL Query</i> .....	63
4.3.2	<i>SQL Stored Procedure</i> .....	64
4.4	MANAGED CARD IMPLEMENTATION .....	65
4.5	REQUEST SECURITY TOKEN IMPLEMENTATION .....	66
4.6	WEBSITE IMPLEMENTATION.....	67
4.7	REQUEST IDENTITY SELECTOR IMPLEMENTATION .....	68
4.8	LOGIN IMPLEMENTATION .....	69
4.9	TIMING TEST QUERY.....	70
4.10	CONCLUSION.....	71
<b>5</b>	<b>EVALUATION</b> .....	<b>73</b>
5.1	INTRODUCTION.....	73
5.2	EVALUATION TEST.....	73
5.2.1	<i>Login to the website using an Information Card</i> .....	73
5.2.2	<i>Register into the website using an Information Card</i> .....	74
5.2.3	<i>Login to the website using an username and password</i> .....	74
5.2.4	<i>Register into the website using an username and password</i> .....	74
5.2.5	<i>Login to the website using an managed card</i> .....	75
5.2.6	<i>Register into the website using an managed card</i> .....	76
5.2.7	<i>Request a service from the website</i> .....	76
5.3	QUANTITATIVE EVALUATION.....	77
5.3.1	<i>Information Card Requirements</i> .....	77
5.3.2	<i>Comparative Test</i> .....	77
5.4	QUALITATIVE EVALUATION .....	78
5.4.1	<i>Identity Management Systems Evaluation</i> .....	78
5.4.2	<i>Information Card Technology Evaluation</i> .....	82
5.5	CONCLUSIONS .....	85
<b>6</b>	<b>CONCLUSION</b> .....	<b>86</b>
6.1	ACHIEVEMENT OF AIM AND OBJECTIVES .....	86
6.2	GENERAL CONCLUSION .....	87
6.3	FUTURE WORK.....	87
<b>7</b>	<b>REFERENCES</b> .....	<b>89</b>
<b>8</b>	<b>APPENDIX A</b> .....	<b>92</b>
8.1	HARDWARE AND SOFTWARE REQUIREMENTS .....	92

<b>9</b>	<b>APPENDIX B</b> .....	<b>93</b>
9.1	PROJECT MANAGEMENT (GANTT CHART) .....	93
9.2	PROJECT MANAGEMENT (PROJECT DIARIES) .....	100
<b>10</b>	<b>APPENDIX C</b> .....	<b>105</b>
10.1	EVALUATION QUESTIONNAIRES .....	105

# List of Figures

<b>FIGURE 2-1: BIOMETRICS CLASSIFICATION .....</b>	<b>16</b>
<b>FIGURE 2-2: DIGITAL CERTIFICATE SCHEMA .....</b>	<b>17</b>
<b>FIGURE 2-3: SINGLE SIGN-ON MODEL (OLSEN, 2007).....</b>	<b>19</b>
<b>FIGURE 2-4: CIRCLE OF TRUST BETWEEN DIFFERENT SERVICES (SULLIVAN, 2005) ..</b>	<b>20</b>
<b>FIGURE 2-5: IDENTITY MANAGEMENT PARTIES (OLSEN, 2007) .....</b>	<b>23</b>
<b>FIGURE 2-6: MULTIPLE IDENTITIES FOR A USER (CLAUS, 2001) .....</b>	<b>25</b>
<b>FIGURE 2-7: GROUP OF DIFFERENT IDENTITIES AREAS (HANSEN, 2008).....</b>	<b>26</b>
<b>FIGURE 2-8: PROPERTIES OF USER-CENTRIC FIM SYSTEMS (BHARGAV , 2007) .....</b>	<b>27</b>
<b>FIGURE 2-9: SAML AUTHENTICATION STATEMENT .....</b>	<b>33</b>
<b>FIGURE 2-10: SAML MESSAGE.....</b>	<b>33</b>
<b>FIGURE 2-11: WS-SECURITY SAML TOKEN PROFILE .....</b>	<b>35</b>
<b>FIGURE 2-12: WEB SERVICE LAYER SECURITY (LO IACONO, 2008) .....</b>	<b>36</b>
<b>FIGURE 2-13: SAMPLE OF FIM (DEMCHENKO, 2004) .....</b>	<b>38</b>
<b>FIGURE 3-1: LOGIN PAGE WITH CARDSPACE SUPPORT.....</b>	<b>42</b>
<b>FIGURE 3-2: THE DATABASE RELATIONSHIP.....</b>	<b>44</b>
<b>FIGURE 3-3: DATABASE STORED PROCEDURES .....</b>	<b>45</b>
<b>FIGURE 3-4: DATABASE TABLES .....</b>	<b>46</b>
<b>FIGURE 3-5: INTERNET INFORMATION SERVICE .....</b>	<b>47</b>
<b>FIGURE 3-6: CREATION OF THE SECURITY CERTIFICATE.....</b>	<b>48</b>
<b>FIGURE 3-7: MICROSOFT MANAGEMENT CONSOLE .....</b>	<b>48</b>
<b>FIGURE 3-8: THE WEBSITE SECURITY CERTIFICATE.....</b>	<b>49</b>
<b>FIGURE 3-9: MAIN PAGE DESIGN.....</b>	<b>51</b>
<b>FIGURE 3-10: THE LOGIN AND REGISTRATION MENU .....</b>	<b>51</b>
<b>FIGURE 3-11: THE INFORMATION CARD INTERFACE .....</b>	<b>52</b>
<b>FIGURE 3-12: LOGIN WITH USERNAME AND PASSWORD.....</b>	<b>52</b>
<b>FIGURE 3-13: USERNAME AND PASSWORD REGISTER INTERFACE.....</b>	<b>53</b>
<b>FIGURE 3-14: MAIN PAGE WEBSITE .....</b>	<b>54</b>
<b>FIGURE 3-15: INFORMATION CARD LOGIN PAGE.....</b>	<b>55</b>
<b>FIGURE 3-16: CARDSPACE IDENTITY SELECTOR .....</b>	<b>55</b>
<b>FIGURE 3-17: USERNAME AND PASSWORD LOGIN PAGE .....</b>	<b>56</b>
<b>FIGURE 3-18: USERNAME AND PASSWORD REGISTER PAGE.....</b>	<b>57</b>
<b>FIGURE 3-19: ASSOCIATE AN INFORMATION CARD PAGE.....</b>	<b>57</b>
<b>FIGURE 3-20: IDENTITY PROVIDER .....</b>	<b>59</b>
<b>FIGURE 3-21: MANAGED CARD .....</b>	<b>60</b>
<b>FIGURE 4-1: IMPLEMENTATION GRAPHIC SCHEMA .....</b>	<b>62</b>
<b>FIGURE 4-2: DATABASE IMPLEMENTATION .....</b>	<b>63</b>
<b>FIGURE 4-3: WEBSITE SOLUTION.....</b>	<b>68</b>
<b>FIGURE 4-4: DATABASE TIMER .....</b>	<b>71</b>
<b>FIGURE 5-1: RESULTS FROM THE IDENTITY MANAGEMENT SYSTEMS EVALUATION .</b>	<b>81</b>
<b>FIGURE 5-2: INFORMATION CARDS VERSUS USERNAMES/PASSWORDS EVALUATION .</b>	<b>85</b>



## List of Tables

<b>TABLE 4-1: CREATING PROCESS OF THE USERINFORMATIONCARDS TABLE .....</b>	<b>64</b>
<b>TABLE 4-2: STORED PROCEDURE GET USER BY INFORMATIONCARD.....</b>	<b>65</b>
<b>TABLE 4-3: MANAGED CARD INFORMATION.....</b>	<b>66</b>
<b>TABLE 4-4: SECURITY TOKEN SERVICE.....</b>	<b>67</b>
<b>TABLE 4-5: REQUEST IDENTITY SELECTOR.....</b>	<b>69</b>
<b>TABLE 4-6: LOGIN WITH INFORMATION CARD PROCESS.....</b>	<b>70</b>
<b>TABLE 4-7: TIMING CONTROL .....</b>	<b>70</b>
<b>TABLE 5-1: LOGIN PROCESS WITH INFORMATION CARDS .....</b>	<b>73</b>
<b>TABLE 5-2: REGISTER PROCESS WITH INFORMATION CARDS.....</b>	<b>74</b>
<b>TABLE 5-3: LOGIN PROCESS WITH USERNAME AND PASSWORD .....</b>	<b>74</b>
<b>TABLE 5-4: REGISTER PROCESS WITH USERNAME AND PASSWORD .....</b>	<b>75</b>
<b>TABLE 5-5: LOGIN PROCESS WITH MANAGED CARDS .....</b>	<b>75</b>
<b>TABLE 5-6: REGISTER PROCESS WITH MANAGED CARDS .....</b>	<b>76</b>
<b>TABLE 5-7: REQUEST SERVICE FROM WEBSITE .....</b>	<b>77</b>
<b>TABLE 5-8: COMPARATIVE TABLE OF THE MOST IMPORTANT BROWSERS.....</b>	<b>77</b>
<b>TABLE 5-9: COMPARATIVE TABLE OF IDENTITY MANAGEMENT SYSTEMS.....</b>	<b>78</b>
<b>TABLE 5-10: ACCEPTANCE LEVEL INFORMATION CARDS AUTHENTICATION.....</b>	<b>82</b>
<b>TABLE 5-11: INFORMATION CARDS VERSUS USER NAMES AND PASSWORDS.....</b>	<b>83</b>

# Acknowledgments

I would like to express my thanks and gratitude to my girlfriend Elina who supported me all the time. As well many thanks go to Scott Henderson who advised and helped me with the document. Also many thanks go to Bill Buchanan who provided the idea of this thesis and led me in the development of this. Last, but not least, the rest of the people including my family, who disinterestedly helped me in completing this thesis.

# 1 Introduction

## 1.1 Context

---

The use of the Internet has been increasing each year, providing services like email and online purchases. Since the Internet was designed without an Identity layer, this is a perfect place for organized crime that tries to steal user digital identities (Coyle, 2007). Due to the fact that there is a fast-growing number of services and online activities on the Internet like e-commerce, bank details, book tickets, there has to be an alternative way to identify the users and the organization that provides the service, in order to protect them against identity theft or identity crime.

Phishing is a new term that is more frequently associated with Internet attacks and are an attempt to criminally and fraudulently acquire personal information, such as usernames, passwords, bank details, by masquerading as a trustworthy entity in an electronic communication. Another related problem is that the user has to identify themselves with a username and password. This method forces the user to remember different usernames and passwords, but most of the time they repeat the same identification for different websites. These identification methods increase the risk that user information is revealed by means of phishing attacks.

## 1.2 Aim and Objectives

---

The aim of this thesis is to investigate new identity management system that protect the user against identity theft, spoofing, phishing, fraud and any other weakness in the current models of authentication. Objectives supporting the aim are to:

- Conduct a critical review of existing literature relating to Identity Management systems.
- Review the Federated Identity Management system as a possible solution against the weakness in the current models of authentication, and review a series of standards for Federated Identity Management which could improve the security on the communication between different parties.
- Design and implement a prototype of Federated Identity Management system which uses security tokens in order to establish a communication between two different parties.
- Provide an evaluation of this model of authentication versus the use of username and password authentication.

## 1.3 Background

---

Since the Internet provides a tool for organizations and personal users, its use has expanded. As a negative effect, the information circulating on the Internet is vulnerable to

attacks and existing security systems do not provide all the necessary security constraints (Network Security, 2007). These problems can create serious economic damage to businesses and frustration to users, who feel unprotected with existing security systems.

This thesis aims to provide a possible solution to combat these problems, as well to provide the user with an easy way to handle their authentication. The solution is based in the use of software tokens of information that provide the user's identity information. These tokens are then protected with strong encryption, in order to protect them against attacks like the man-in-the-middle (Bhargav, 2007). They contain identity information about the user and these can be used to authenticate them for different services. When the user has to identify themselves to obtain a service from Internet, they can select an information token that satisfies the service policy and use this to send their identity information.

As well, this thesis introduces the concept of Identity Provider (IP), which can prove that an identity information belongs to the user that is trying to obtain a service. They can thus be a recognized organization such as the Government that provides identity information to authenticate different users against service providers. If the user has to identify themselves to obtain a service from Internet, they can select a managed card that satisfies the service policy and use this in order to request to the IP their identity information. Then the IP return a security token with the user information and it is sent to the service provider to obtain the service. A managed card is an information card that is provided by the IP. This card contains the claims requested by the service.

## 1.4 Thesis layout

---

A brief overview of each of the chapters is:

- **Chapter 2 (Literature Review).** This presents a critical analysis of the issues associated with existing authentication systems. It describes the current Identity Management systems such as the traditional username/password model, biometrics, security certificates and security tokens. Along with this it presents a review of the Federated Identity Management system as an alternative solution to the problems related with the Identity Management systems.
- **Chapter 3 (Requirements Analysis and Design).** This presents a solution to the issues outlined in the literature review, and covers the hardware and software used in the software development. The solution has been the set up of a website with CardSpace support, in order to evaluate different requirement such as performance, functionality and usability. The system provides two different types of authentication systems; the first one is based on the use of Information Cards and the second one is the traditional Identity Management based on the use of username and password.
- **Chapter 4 (Implementation).** This presents how the solution has been implemented, showing specific parts of code in order to develop a prototype of Federated

Identity Management system. Some of the parts covered in this chapter are database implementation, Managed Card implementation, login Implementation and so on.

- **Chapter 5 (Evaluation).** This presents an evaluation of the developed prototype. The evaluation process will cover the most important aspects of testing for the Identity Management system. As well, this chapter presents an evaluation between Information Cards authentication and username and password authentication. A group of different users have carried out this evaluation in order to provide a clear conclusion about the system.
- **Chapter 6 (Conclusions).** Finally the conclusion returns to the aims and objectives of this thesis, highlighting the benefits of the model proposed and the obtained results for the developed prototype. Also future direction of this research is presented.

## 2 Literature Review

### 2.1 Introduction

---

This literature review looks at the various Identity Management (IM) systems. It will firstly examine the problems related with the current systems of authentication examining models of authentication, such as username and password, biometrics, Digital Certificates and Security Tokens. The second part will present a Federated Identity Management (FIM) model, which has three different entities: **the user**; **the Identity Provider (IP)**; and the **Service Provider (SP)**. The user is the person that wants to receive the service, whereas the IP is in charge of managing and issuing user credentials, and the SP, or the Relying Party (RP), provides the service. The SP will then provide the service to the user based on their credentials. In this model, the user has control over their credentials. When a user wants to obtain a service from the SP, they can request a Security Token from the IP that contains their identity information, and provide this to the SP.

As well, this chapter reviews the different models of FIM and a series of standard languages that facilitate the creation of the different elements involved in the FIM model such as the IP, the Security Tokens, the communication protocols, and so on.

### 2.2 Authentication

---

Many applications and services used today are located on remote systems. For a user to can gain access to these systems, they should authenticate themselves to the system. Authentication can therefore be defined as the process of validating a user before allowing the user access. It thus provides a method to prove that a user (or entity) is trusted, and that the supplied details are valid, and it must avoid the use of plagiarism, which means that a user provides information belonging to another user.

Some of the problems relating to the traditional identity management are due to the fact this was designed for the SPs and not for the users. One of the most common problems is that the user is required to memorise multiple passwords for using different services. Due to the fact that the number of online services available is increasing, the use of passwords is untenable for many users. Another problem is that the user sometime provides the same identity credentials for different services, because the user cannot remember all their credentials for every different service. All these problems result in a poor identity management system that cannot meet current requirements.

The proposed solution in this thesis is based on user-centric identity management, where an individual's identity is exposed by a set of attributes. These attributes can represent a group of claims made by the user without being certified by a third party, or attributes that have been certificated by a third party. This is in the same way that, in the

real world, a user can handle a collection of different identities, belonging to different sets of attributes. The following sections discuss authentication models.

### 2.2.1 Username and Password Authentication

Username and password authentication is a simple authentication method that is used to gain access to a service, for example in a website. This method is known as Password Authentication Protocol (PAP), and is also used by point-to-point protocols. It is composed of a username or login and a password (Cheng-Chi, 2005), where the username is a unique name that identifies each user, and are chosen by the user, or by the system administrator. The name of user is generally set up as a combination of their name or their initials along with some arbitrary numbers.

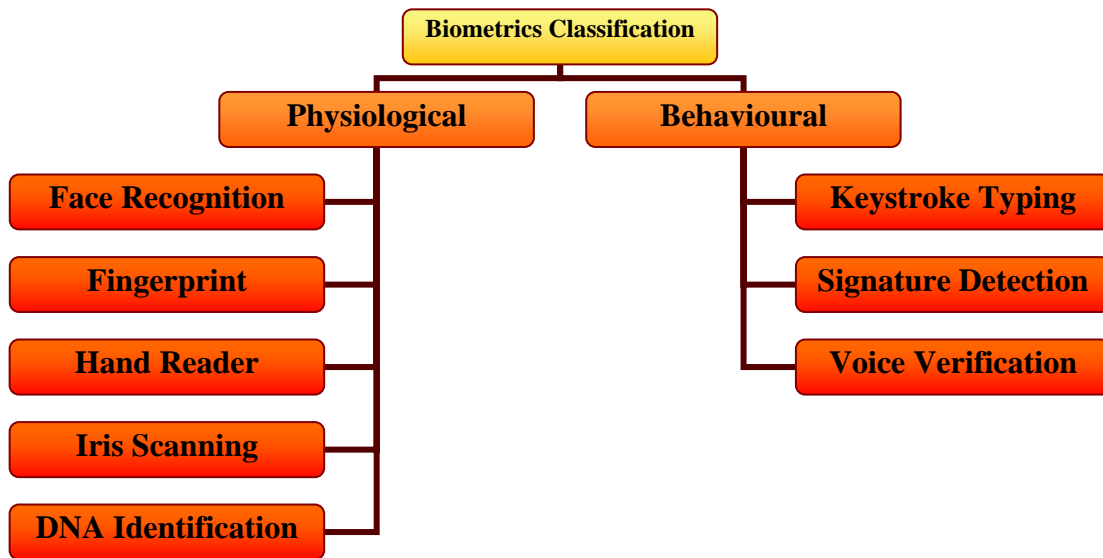
The password is chosen by the user, or it can be created arbitrarily, in some cases. For security reasons, some systems require a password that contains digits and/or symbols, so that it cannot be easily revealed. The advantage of usernames and passwords is that the user can choose the username and password, and that most users know how to use this authentication process. With username and password authentication there is also no need to install any extra software or extra hardware device, and it is often seen as the least expensive authentication to use. The disadvantage with the security of the username and password authentication technique depends on the user's capability of retaining the username and password in a secret manner. Another disadvantage is that it could be captured when it is sent through the network. Also the complexity of the system can increase over time, as support has to be added to handle resetting passwords, handling locked accounts, and re-sending passwords. The system is also susceptible to numerous attacks such as the man-in-the-middle.

### 2.2.2 Biometric

Figure 2.1 shows a classification for the different biometric authentication protocols. Biometric authentication is divided in two different classes which are discussed in the following sections. Biometric authentication protocol is based on the use of physical aspects or behaviour qualities of the users. This can include biometrics behaviours such as typing rhythm, behaviour footprints, and so on, and these can be used to identify the user without having to interfere or interrupt with the user's activities (Vildjiounaite, 2006). The process of authentication is carried out by means of comparison between the input and previous stored information. Physiological aspects are related to the physical features such as:

- **Face Recognition.** Face recognition is an authentication method based on the recognition of the users face by means of a digital image. The system compares the image obtained as input with an image of the user recorded on a database.
- **Fingerprint.** Fingerprint is an authentication method based on the impression of the bottom of the user's finger. The authentication system scans the fingerprint of the user and compares this with the user's fingerprint stored on a database.

- **Hand Reader.** Hand Reader is an authentication method based on the geometry of the user's hand. The authentication system scans the unique geometry of the user's hand and compares this with the template stored on a database.



**Figure 2-1: Biometrics Classification**

- **Iris Scanning.** Iris Scanning is an authentication method based on a high-resolution image of the user's iris. The authentication system uses a camera with infrared illumination to obtain an image of the user's iris, and then this image is converted to digital template and is compared with the previous template of the user stored on a database.
- **DNA Identification.** DNA Identification is an authentication method based in the structure present in every human cell. The authentication system obtains a sample of the user's DNA (the sample can be obtained from blood, saliva, hair, semen, or tissue) and compares with the previous template of the user stored in the database.

Behavioural aspects are related to the behaviour of a user such as:

- **Signature Detection.** Signature Detection is an authentication method based on the user's handwritten signatures. The authentication system scans the user's signature compares this with the image signature stored in the database. Advanced Signature Detection systems can check rhythm, acceleration and pressure of the user.
- **Voice Recognition.** Voice Recognition is an authentication method based on the user's voice tone. The authentication system compares the user's voice with the pattern stored on a database.
- **Keystroke Typing.** Keystroke Typing is an authentication method based on the measuring of the time that the key is hold down and duration between taps when the user writes their authentication. When the user writes their authentication, the system measures its result with the pattern stored on a database



### 2.2.3 Digital Certificates

Public key cryptography is based on the use of one public key and one private key. The two keys are linked together by means of a complex mathematical equation (Galindo, 2008). Figure 2.2 shows the encryption process. When User2 wants to send a message to User1, User2 uses User1's public key to encrypt the message. When User1 receives the message, he/she can use their private key to decrypt the message. The problem with public key cryptography is that anyone can create the pair of keys using an identity that does not belong to them, also it is difficult to distribute the public key of the user. Digital Certificates provide a solution to this problem based on the use of a public key certificate. This identity certificate provides a digital signature that binds an identity with its public key. The certificate is then used to check that the public key belongs to the identity. Usually the signature is provided by a certificate authority (CA) or by the user in the case of a self-signed certificate (Claus, 2001). Figure 2.2 shows the process of using a digital certificate when the owner of the certificate sends a message to another person.

The public key is published to a key storage place, such as for a PKI (Public Key Infrastructure) server, so any user has access to it. In order to provide the validity of the certificate, the user provides the certificate signed with the private key of a credible CA. The CA then provides a digital certificate, which contains the public key and the identity of the issuer. An example of digital certificate is the X.509 certificate.

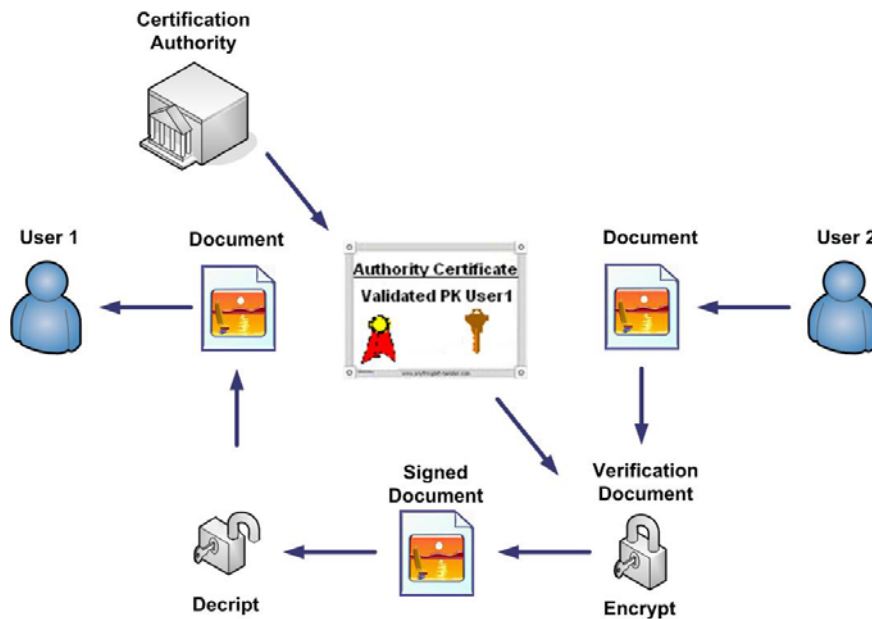


Figure 2-2: Digital Certificate Schema

### 2.2.4 Security Tokens

A security token is a block of data that communicates information about a digital identity. This then contains one or more claims that represent identity information. These claims provide information about the issuer and are protected by security such as username/password, X.509 certificates, Kerberos tickets (see Section 2.3), and so on, in or-

der that the recipient trusts the received message. Some of the current security token systems are presented below:

- **SAML Tokens.** Security Assertion Markup Language (SAML) is a XML representation for exchanging of security information between two different parties. SAML allows the making of assertions about a number of claims that represent the identity information within the security token. SAML also provides a solution for the Single Sign-On (SSO) problem in an FIM system (See Section 2.3), which is an infrastructure containing different domains where users identify themselves to one domain, and then are trusted onto other domains. SAML has been approved by the Organization for the Advancement of Structured Information Standards (OASIS) and backed by the Liberty Alliance's interoperability testing (Smith, 2008).
- **Microsoft CardSpace.** CardSpace represents a virtual environment where the user can store their identity information within an Information Card (or Info Card). It provides an Identity Selector interface where the user can select their information card in order to provide their credentials. The Information Card is then used to obtain the security token that contains the requested user's claims. This token is built using the SAML. As with SAML, CardSpace also provides a solution for Single Sign On (SSO) problem in a FIM system.

## 2.3 Federated Identity Management (FIM)

---

There are three fundamental aspects to FIM systems: Identity Providers (IPs), circles of trust, and Web Services Federated Identity (see Section 2.10.4). Some of the solutions which provide the means of FIM are (Dean R., 2006):

- Identity system based on SSO and user information exchange.
- IPs that manage different user information and user identities.
- Secure application interaction using web services technology.

FIM systems give users the control over their identity management and eliminate the use of different usernames and passwords in order to manage user identification and help users avoid phishing attacks. The function of the identity management system can be divided into two parts. The first part consists of issuing users with credentials and unique identifiers during the initial registration phase, and the second consists of authenticating users and controlling their access to services and resources based on their identifiers and credentials during the service operation phase (Josang, 2005). The different stages for an identity can contain enrolment, storage, retrieval, provisioning and revocation of identity attributes. A simple definition of a FIM is:

“A system consists of software components and protocols that handle the identity of individuals throughout their identity life cycle. A FIM system involves three main entities, namely user, IP and SP. The IP manages and potentially issues user credentials and the SP (also known as relying parties) are entities that provide services to users based on their attributes.”

(Bhargav, 2007)

A SSO is an extension of the User-Centric FIM. In this, the user can be authenticated by one IP, and can be considered to be authenticated by other Service Providers (SPs), depending on trust levels. In this system (see Figure 2-3), the user only needs to authenticate themselves once (SSO), in order to gain all the services (Josang, 2005). In a SSO system, the user provides the same identifier by every SP (Pfitzmann, 2004). An example of SSO federated system is a university where students only use one identity and obtain sign-on to access information from this university and from other academic organizations. The user's university maintains their identity and credentials. Other organizations rely on the information provided by the university to authenticate the user (Smith, 2008). This example has been implemented in a higher education environment with Eduserv Athens System, which has been created for UK higher education institutions. Eduserv is used as IP authentication for users on behalf of many SPs such as resource libraries (Smith, 2008). A Kerberos system implements this scenario, where the Kerberos authentication server is the centralized identifier and credential provider (Josang, 2005).

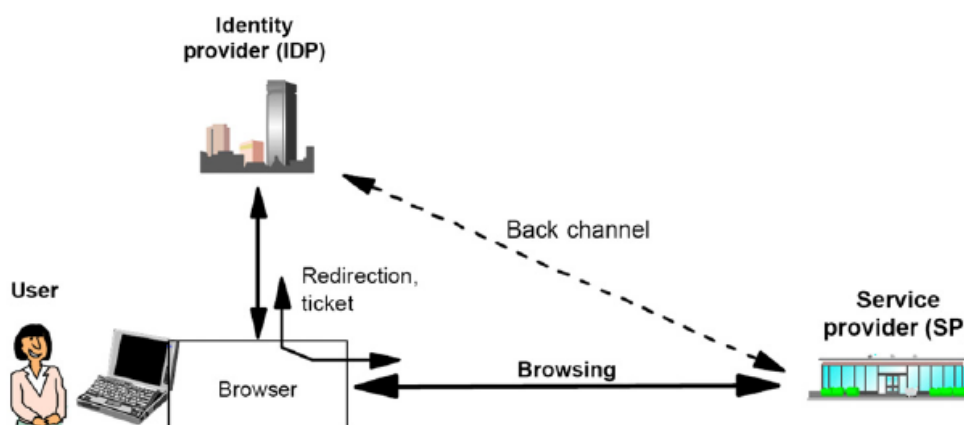


Figure 2-3: Single sign-on model (Olsen, 2007)

## 2.4 Circles of Trust in Federated Systems

Figure 2-4 shows a Circle of Trust that represents a relationship between different parties in order to share identity information and it is based on the guidelines established by the end user (Sullivan, 2005). Within this, the user can move from one trusted party to another without having to identify themselves over and over again (Olsen, 2007):

“Once a user has been authenticated by a Circle of Trust IP, that individual can be easily recognized and take part in relationships with other Service Providers within the Circle of Trust A trusts B, B trusts C so A trusts C, and so on.”

(Sullivan, 2005)

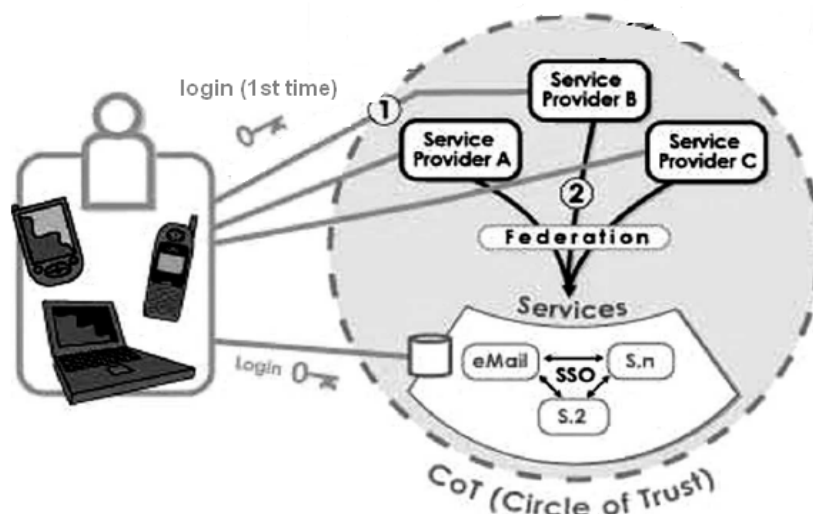


Figure 2-4: Circle of Trust between different services (Sullivan, 2005)

## 2.5 OECD Data Protection Principles

The user-centric Federated Identity Management (FIM) model should satisfy the following OECD principles of security and data protection (Anon, 1980). For this, the first principle states that there must be a limit on the personal data that is sent. In addition, information collected should be sent by legitimate and reliable means. Finally, all information sent should be sent with the knowledge of the owner (Hansen, 2008).

The second principle states that the data that is to be sent has to be used for the purposes that it was requested, where the information must thus be accurate, complete and current at the moment that it is sent. The third principle states that the goals of how the data is to be used are expressed. These goals must be expressed before the data is sent and it must be demonstrated that the data is used only for the stated purposes or for other compatible with these. The next principle (fourth) states that data should never be disclosed, delivered or used for any other purpose than that that has been stipulated in the purpose specification principle. This principle can be omitted in the case of:

- The owner of data authorized to do it.
- In the case where the law requires it.

The fifth principle states that the data that is going to be sent has to be protected against risks such as manipulation, modification, loss, unauthorized access, improper use or disclosure to other parties. The sixth principle states that there must be an open policy on the development, practices and policies that manipulate the data. The means that the way data is used must be transparent. In addition, the data controller must be easily identified as well. This seventh principle states that the owner of the data should have the following rights:

1. The owner should be able to get their data from the data controller, or get confirmation if the data will or will not be kept by the data controller.
2. The owner should have timely confirmation of the use of data belonging to him.
3. If a request is refused within the two first points, an explanation of why it was denied and the opportunity to dispute such denial
4. If the owner contests such denial and this is accepted, the system must delete, modify, the data.

The eighth principle states that the data controller must be concerned with fulfilling all the principles listed above.

## **2.6 Laws of Identity**

---

The Laws of Identity are seven essential laws that explain the successes and failures of digital identity systems. The following provides a summary of these laws, drawn from (Cameron, 2005).

### **2.6.1 User Control and Consent**

Information identifying a user must only be revealed with the consent of the user. The success of such a system is dependent on the user, where the system has to be convenient and appealing, but above all must be trusted by the user. The system must also allow the user to control the digital identities, and what information is transmitted. The user must also be protected from deception, by validating all requests for information. It is vital that the user is confident that any information given will go to the correct place, and be used for the stated purpose. The user must also be informed when they have chosen an Identity Provider that can track internet behaviour. Consistency is also important, the user should feel in control regardless of the environment. The user should have the same level of control whether in a consumer or enterprise situation. This concept of the importance of the user's permission is crucial even if a refusal would mean to break a company's conditions and employment. This serves both to inform to the employee and as a cover for the employer. The Law of User Control and Consent permits the user to employ mechanisms whereby the Metasystem memorises the decisions of the user, and the users may decide to have them applied automatically on future occasions.

### **2.6.2 Minimal Disclosure for a Constrained Use**

The best and most stable long-term solution according to Cameron (2005) is the one that reveals the least amount of identifying information and best limits its use. He believes technical identity systems should be build to utilise identifying information on the basis that violation of the system is always possible, and such a violation represents a risk. To lessen such a risk it is best to obtain information merely on a *need to know* basis and to maintain information merely on a *need to maintain* basis. By following these practices, the least possible damage can be caused in the case of a violation to the system. A system constructed with the principles of information minimalism is consequently a less appealing target for identity theft, which reduces the risk even further.

By limiting use to an explicit scenario (in conjunction with the use policy described in the Law of Control), the effectiveness of the *need to know* principle in reducing risk is further magnified. There is no longer the possibility of collecting and keeping information *just in case* it might, one day, be required. The concept of *least identifying information* must be interpreted as indicating not only the least number of requests, but the information least probable of identifying a given individual across several contexts. Multiple identity violations have taken place where this law has been breached. Cameron (2005) suggests that the law of Minimal Disclosure can be explained in this way: *aggregation of identifying information also aggregates risk. To minimize risk, minimize aggregation.*

### 2.6.3 Justifiable Parties

Digital identity systems must be created so that the disclosure of identifying information is restricted to parties having a required and justifiable place in a given identity relationship. The system must also make the user aware of the party or the parties with whom they are communicating with, whilst sharing information. The requirements of the law of justification apply to both the subject who is revealing information and the other party who depends on it. Cameron (2005) describes their experience with Microsoft Passport in this regard. Microsoft Passport was seen by Internet users to be a useful way to gain access to MSN sites, with about a billion interactions every day. Nevertheless, it did not make sense to the majority of the MSN sites for Microsoft to be engaged in their customer relationships, nor were users interested in a Microsoft Identity service to be aware of all of their internet activities. Consequently, Microsoft Passport failed in its job of providing an identity system for the internet.

Cameron argues that many more examples of the law of Justifiable Parties will be brought forward in the future. Some governments are currently considering operating digital identity services, and this would make sense for people to use when doing business with the government. However, is it necessary for government identities to be used in managing access to a user's personal usage of the computer? From the law of Control of Consent it has been stated that an identity system must be predictable and translucent in order to earn trust. Figure 2.5 shows an Identity Management system where the user obtains service from different organizations by means an IP. Any use policy would allow all parties to collaborate with authorities in e.g criminal investigations. However, this does not mean the state is party to identity relationship. This should consequently be made clear in the policy where the information is shared.

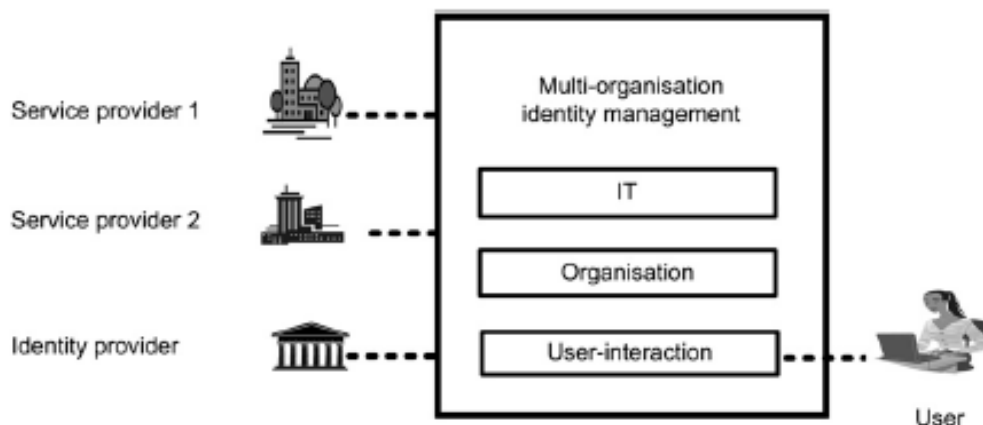


Figure 2-5: Identity Management Parties (Olsen, 2007)

#### 2.6.4 Directed Identity

A universal identity system has to support both *omni-directional* identifiers for use by public entities and *unidirectional* identifiers for use by private entities, thereby assisting discovery while avoiding any unwarranted release of correlation handles. The public entities can have identifiers that are invariant and familiar. These identifiers can be thought of as beacons due to them giving out the identity to anyone who shows up, and these beacons are always omni-directional (meaning that they are willing to expose their existence to all identifiers). An example of a well known public identity is a corporate website with a well-known URL and public key certificate. Cameron (2005) states that there is no advantage to change a public URL, in fact there are only disadvantages. It is possible for every visitor of the website to examine the public key certificate, and it is just as acceptable for everyone to know the website's public existence. A publicly visible device such as a video projector is a second example of such public entity. The video projector might be located in a conference room in a company and offers digital services by advertising itself to anyone seeing it. This is thereby an example of an omni-directional identity.

Alternatively, a consumer who is visiting a company's website can use the identity beacon of that site to make a decision as to whether or not he/she wants to begin a relationship with it. A *unidirectional* identity relation can then be established with the site by choosing an identifier to use for this site only, and if this unidirectional identity relation would be established with a different site, it would involve setting up an entirely unrelated identifier. Due to this fact, there is no relationship handle produced that can be shared between websites to gather a comprehensive list of profile activities and preferences. The omni-directional identity beacon that is described above in the example of the video projector, could be used by the computer user in the conference room where this projector is located, to decide whether or not to interact with it (in accordance to the Law of Control). If the user does interact with the projector, a momentary unidirectional identity relation would be established between the computer and the projector, providing a safe connection while revealing the least identifying information possible, as per the Law of minimal disclosure. Wireless technologies such as Bluetooth have not yet agreed with the Law of Directed Identity. This technology uses public beacons for pri-

vate entities, which explains the consumer backlash innovators in these areas are dealing with at the moment.

Another example of identification of users where privacy is an issue includes the proposed usage of RFID (Radio Frequency Identification) technology in passports and student tracking applications. The devices using the RFID technology currently release an omni-directional public beacon, which is not suitable for usage by private people. The passport readers are public devices, therefore they should use an omni-directional beacon. The passports however should only respond to an authorised reader, and should not be releasing signals to any eavesdropper that could recognise their owners and mark them as nationals of a given country.

### **2.6.5 Pluralism of Operators and Technologies**

A universal identity system must direct and enable the relationship between multiple identity technologies run by multiple Identity Providers. Ideally, there would be only one-way to express identity, but the many situations where identity is required does not allow for it. Figure 2.6 shows the relation of one user with their multiple identities. It might seem sensible to use government issued digital identity when dealing with government services, but in many cultures, neither employers nor employees would feel comfortable employing government identifiers to log in at work. An identifier of this type might be utilised to transmit taxation information, or it might be used to track employment history. In the case of employment, this on its own is sufficiently autonomous that it requires its own identity and does not have to be observed by a government-run technology.

Consumers and other individuals on the other hand are likely to desire a higher level of privacy than is likely to be provided by any employer. Therefore, with digital identity, it is not only a case of having Identity Providers lead by different parties (including the individuals themselves), but also of providing identity systems offering different (and in some cases conflicting) features.

Cameron (2005) emphasises that a universal system has to include differentiation; at the same time as identifying that, each one of us is (in different contexts and simultaneously) a citizen, an employee, a customer and a virtual persona. This shows that different identity systems must exist in a Metasystem. There is a need for an encapsulating protocol (meaning a way of agreeing on and transporting things). In addition to this, a method to get information out through a unified user experience allowing people and companies to choose appropriate Identity Providers and features as they carry out their daily activities. This fifth law defines a universal identity system that works with different identity technologies and permits the use of multiple identities providers. A universal identity system cannot be centralized because the characteristics that would make a system ideal in one context can disqualify it in a different context.



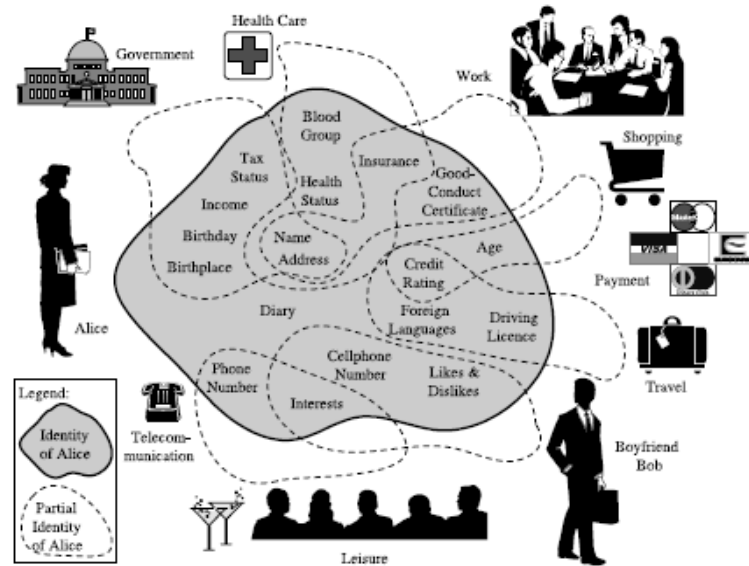


Figure 2-6: Multiple Identities for a user (Claus, 2001)

### 2.6.6 Human Integration

The Universal Identity Metasystem has to describe the individual user to be a part of the distributed system, included in unmistakable human-machine communication mechanisms giving protection against any identity attacks. Cameron (2005) argues that the securing of the channel between web servers and browsers is good quality, through the use of cryptography. But an area where less quality is provided in terms of safety is protection of the two to three feet channel which goes from the browser's display to the brain of the individual using it. The main issue here is that the user is, in the majority of cases, not aware of what identities he/she is dealing with while navigating the web. Cameron (2005) therefore claims that something has to be done to improve this service, to the extent where identity systems integrate the individual user. Figure 2.7 shows the relation of the user with a group of different identities.

As the identity system has to work and be able to function in all these different areas, it also has to be safe in all areas. Cameron describes one example of this with United Airlines' Channel 9. This specific channel transmits live conversation between the cockpit and the air traffic control, and this conversation is very important, technical and focused. The participants of the conversation do not *chat* in fact they all know precisely what to expect from the tower and the airplane. Consequently, despite there being a lot of noise or static present, it should still be easy for the pilot and controller to understand the exact content of the communication. If something goes wrong, the broken predictability of the channel indicates the urgency of the situation, making everyone aware of it. Cameron emphasises that the remaining issue is how to achieve these high levels of reliability in the communication between a system and the individual user. User testing can be used in order to assess and measure this.

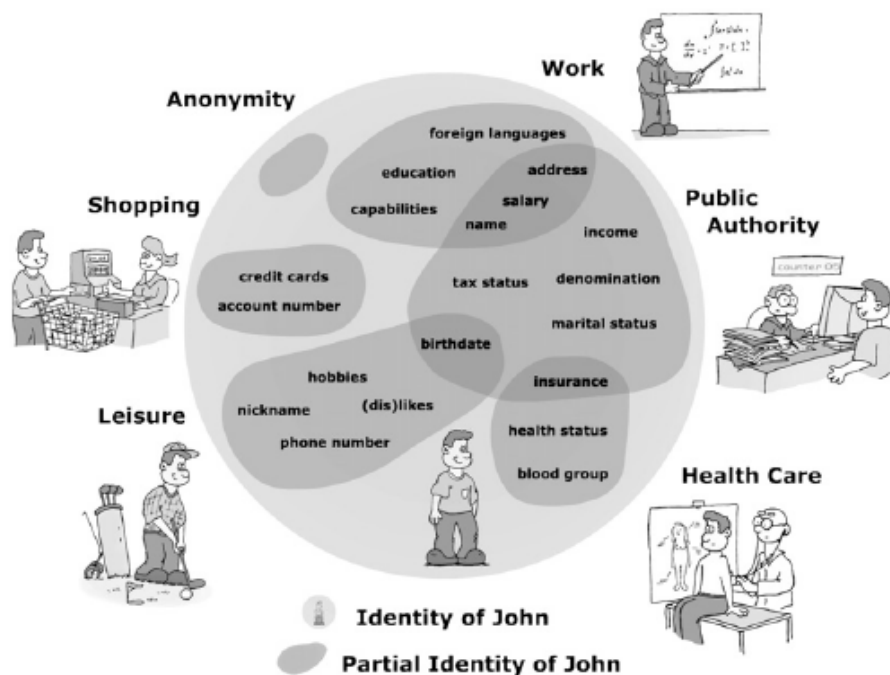


Figure 2-7: Group of different identities areas (Hansen, 2008)

### 2.6.7 Consistent Experience across Contexts

A unifying identity system has to guarantee its users a straightforward, consistent experience at the same time as allowing separation of contexts by means of multiple operators and technologies. Cameron (2005) identifies the following number of contextual identity choices:

- **Browsing:** a self-asserted identity for exploring the Web (giving away no real data).
- **Personal:** a self-asserted identity for sites with which the user wants an ongoing but private relationship (including my name and a long-term e-mail address).
- **Community:** a public identity for collaborating with others.
- **Professional:** a public identity for collaborating issued by their employer.
- **Credit card:** an identity issued by their financial institution.

This seventh law defines that a universal identity system must guarantee to the user a simple and consistent identity process. As well, this has to provide different digital identities for the user depending on the context in which the user has to be identified. The user must have the opportunity of choosing between their different digital identities, which one is most appropriate for the current context.

## 2.7 User-control in Federated Identity Management

One of the best advantages about user-centric FIM is that the user has control over their identifications. The user control is obtained by means of making multiple system properties. Some of these properties are basic, they do not depend of other properties and

others are composed from several basic properties. Figure 2.8 and the following discussion are based upon the work by Bhargav et al (Bhargav, 2007).

### 2.7.1 Basic Properties

There are three different types of basic properties:

1. FIM System properties
2. Transaction Properties
3. Identity information properties

The basic properties are represented with the nodes with indegree 0 in the directed graph in Figure 2.8.

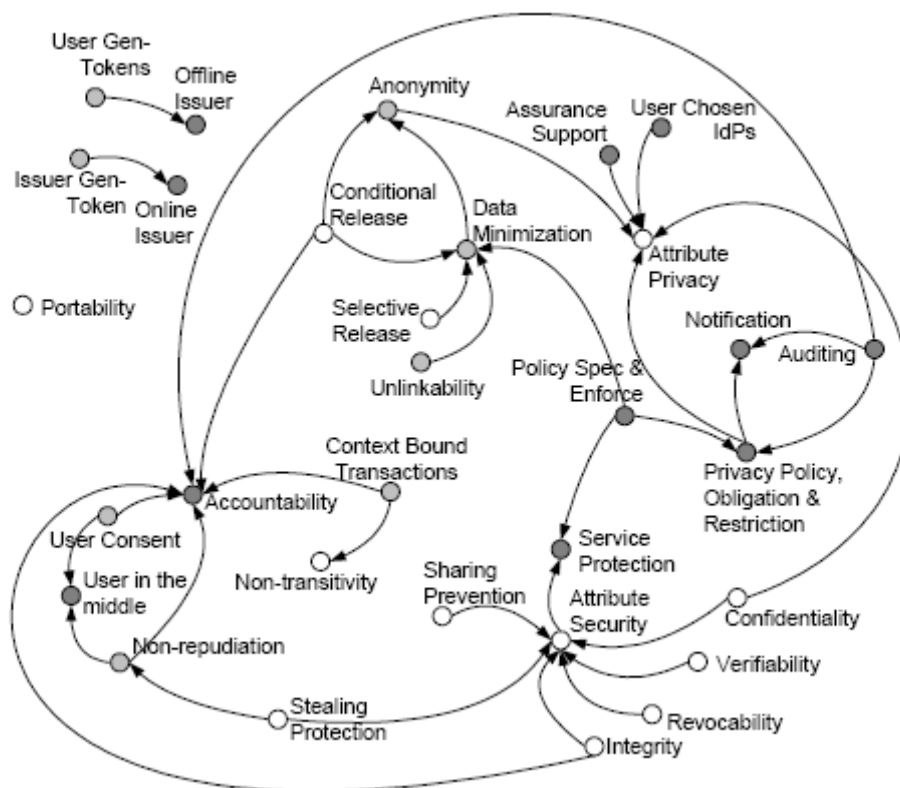


Figure 2-8: Properties of user-centric FIM systems (Bhargav , 2007)

### 2.7.2 FIM System Properties

FIM System Properties can be divided into four basic properties:

1. The first property is the user-chosen Identity Providers. The user can choose between different Identity Providers, since he can trust one IP more than another for a specific service.
2. The second property is policy specification and enforcement. This property is based on the definition, management and enforcement of different policy related issues. Other properties of the system rely on these policies.

3. The third property is the auditing. The auditing can be defined in order to obtain other desired properties of the FIM system using appropriate mechanisms.
4. The fourth property is assurance support. The assurances provide security to the user from one service, helping the user to trust their identifications to this service.

### 2.7.3 Transaction Properties

The transaction properties are all the transactions related with identity relating information. The first property is the context bound transactions. This property requires that all the messages in one transaction be bound to the context in which this transaction is achieved. The messages of one transaction do not have any value in another context. The second property is that transactions do not contain linking information, that could be used to link to the various entities involved in the communication. The third property is the user content. This property is based on the fact that a user knows the current transaction and he agrees to execute this transaction. This property is reinforced by Cameron's first law. The fourth property is user-generated tokens. This user-generated tokens property defines that the user is going to generate the tokens to provide these to the service. The fifth property is issuer-generated tokens. The issuer-generated tokens property defines that an IP is going to generate the tokens to provide these to the service.

### 2.7.4 Identity Information Properties

It is possible to classify computer security issues into three categories: confidentiality, integrity and availability. For **confidentiality**, we can define this property as the protection of identity information from unauthorized use. This property requires that this information is only revealed to an authorized recipient. For **Integrity**, it is stated that information is not modified in an illegal manner. The information cannot be modified in an unauthorized context, for the IP or for the user in the case of self-issue information. The use of a certificate is a good way of avoiding that the information could be changed.

For **verifiability**, this property permits that the user can check that the identity information provided for the Identity Provider is correct. Then the **Stealing protection** property protects the identity information, credentials and private keys against virus, worms, hackers, that try to steal user information. With **Revocation**, the property is applied in order to preserve the validity of the data. This property is more relevant when the identity information is issued by an IP, which can revoke the identity information by means of a certificate.

For **Sharing Prevention**, the property protects the user from providing their credentials to an unauthorized party. This information could be used to gain illegal access to a SP. Sharing prevention is an important issue in a user-centric FIM system, since malicious users could obtain high user privileges. With **Portability**, this property defines that the system must provide the user with the means to use their credentials in different devices such as Desktop PC's, Laptops, smart Phones.

### 2.7.5 Composite Properties

Composite properties are properties that are constructed from one or several basic properties. Composite properties are shown as the nodes with an indegree greater than zero, in Figure 2.8. The dependencies are represented in the graph with an arrow from a property A (node) to a composite property B (node), this means that a basic or composite property A is required or helpful to achieve the property B. A list of the composite properties are:

- **Attribute Security:** This is based on the protection of the user's identity information. Attribute Security must provide integrity and protection against stealing of the user's attributes. It must also prevent another person obtaining the identities of the user. Additionally, revocation of identity must be possible, since user's attributes can be provided for other parties.
- **Service protection:** This requires the use of accounts for the protection of the user's attributes. Only authorized parties can requests user's attributes.
- **Non-Repudiation:** This avoids that any of the parts involved in the transaction could deny having executed the service. Mutual non-repudiation guarantees that the user and the SP cannot later deny having executed the transaction. This is also highlighted in the individual participation principle (Anon, 1980).
- **Non-Transitivity:** This guarantees that a user cannot ask for a service, with an identity information token that has been used in a previous service.
- **Data minimization:** This is used in order to provide the minimal required information within a transaction. The identity management system must guarantee that only the requested information is released to the SP. Data minimization property is highlighted in the collection limitation principle (Anon, 1980).
- **Attribute privacy:** The identity management system must ensure that the user keeps the control over their attributes in every transaction. As well, the identity management system must permit that the user could choose between different Identity Providers. The anonymity property avoids that unnecessary identity information is given to the SP. Anonymity is obtained with the use of privacy policies, obligations and restrictions properties.

In order to ensure attribute privacy, it is necessary to support privacy policies, obligations and restrictions. This is achieved by ensuring the following properties:

- **Confidentiality:** this property guarantees that identity information is not released by mistake to one SP.
- **Accountability:** This property tries to guarantee that the different parts involved in a transaction are responsible for their actions.
- **Obligations and Restrictions:** This defines the obligations of the involved parties, and the restrictions of how they will use identity information. If the user provides

full control of their identity information to the Identity provider, then the user must satisfy all the obligations and restrictions.

- **Anonymity:** It is necessary for users to remain anonymous within a given transaction. Anonymity property is associated to the data minimization property.
- **Notification:** The identity management system must provide that the user can receive and send notifications concerning to the use of their identity information.
- **The user-in the-middle:** This property is at the heart of a user centric FIM system. This property symbolizes how the user is implicated in every transaction. The user is involved in the release of their identity information.

The user can be involved in the transaction in two different manners:

1. The user selects an IP that returns a token to the user with their identity information. Then the user redirects the token to the SP.
2. The user maintains their identity information and he is involved in the creation of the token.

## 2.8 Federated Identity Management Models

---

A Federated Identity Management System can be differentiated in to two different models, relationship focused and credential focused (Bhargav, 2007). The following sections outline there.

### 2.8.1 Relationship-focused Model

In the relationship-focused model, the user only maintains communication with the IPs (Pfitzmann, 2004), so that if the user needs to send identity information to a service, the IP provides this information. The user is involved in every transaction, so that they have control over their attributes. In this model, the FIM system controls the user's identity information by means of an Identity Provider. In each transaction, the user requests their identity information from one Identity Provider and retrieves this information dynamically during the transaction. The information is sent in a short-term identity federation token that is signed by the Identity Provider. This token is created using a protocol such as SAML, Liberty or WS-Federation.

The advantage of a relationship-focused system, is that the Identity Provider provides a short lifetime token to the user. This token contains the user's identity information and can only be used once, so this reduces the risk of the case that the token is stolen. This feature satisfies the property of sharing prevention (Bhargav, 2007). In general, if we want to develop a relationship-focused system, we need the use of an online Identity Provider in order to validate the user's account and the use of a well-known public key cryptography. The disadvantage of a relationship-focused system, the IP has to be online during the transaction between the user and the SP. This turns the Identity Provider into a single point of failure for this system. Another disadvantage from these systems is that the Identity Provider is present in every transaction, resulting in privacy concerns. This system is also open to attack that can be executed from the man in the

middle (Asokan, 2005). One person can intercept the token and can try to modify this during its lifetime.

### 2.8.2 Credential-focused Model

On the contrary, in the credential-focused model, the IP provides long-term credentials to the user. In this model, the user can contain their credentials. If the user has to communicate with a service, they can provide their attributes (Pfitzmann, 2004). In the same way with the relationship-focused model, the user is involved in every transaction, but the user maintains their long-term credentials locally. In these systems, the user can manage their credentials without involving the IP in every transaction and it is achieved by means of keeping a security token with long-term lifetime and a non-transitive credential in the user computer. Since a credential-focused system keeps long-term credentials, it is necessary to protect these with some form of cryptography. The non-transitivity property is obtained by means of cryptography.

The advantage of a credential-focused system, the Identity Provider can be offline when the user executes a transaction, breaking the need that the Identity Provider has to be present for every transaction. This also guarantees that the Identity Provider cannot trace the user's activities, satisfying the Data Minimization and Anonymity properties. The disadvantage of a credential-focused system is that it keeps long-term credentials, and this introduces a risk by means of theft or sharing of these credentials. When one token is sent in a communication, it can be intercepted in order to be modified and used in an impersonation attack. In order to protect credential-focused systems from theft or sharing credentials, sharing prevention methods have to be implemented. Revocation of credentials protects credential-focused systems against these risks. The revocation capabilities permits to a credential-focused system terminates the credential lifetime when the system notices that the user or other party have lost or misused their identity credentials. As a disadvantage, revocation of credentials generates a higher workload on the user side and this requires a more expensive user system to be executed.

### 2.8.3 Relation with the real world

If we try to compare both models with the real world, we can see two clear examples:

- A **relationship-focused model** between a user and their IP can be considered like a person and their credit card. The IP is the company that issued the credit card. If this person requires to use the credit card, the authorizing company is required to approve the transaction in the name of this person.
- A **credential-focused model** is the relationship between someone and their passport. When this person has to provide their credentials, the passport itself is sufficient to identify this person (the IP is not involved). In order to know the state of their credential (revocation from the IP), the passport state is checked.

One of the greatest benefits of the FIM system is that the user has the control of their credentials in every transaction. On the contrary, in the traditional identity management the SP has control over the user's credentials. Other advantages of this model, is that the

user can choose between different IPs to provide their credentials, so he could decide which providers would be more appropriate for required service (Biskup, 2008).

## 2.9 Standards for Federated Identity Management

---

In order to implement a Federated Identity Management system there are a number of standards that provide support for different tasks involved in the process of identification. An overview of some of these standards is now presented.

### 2.9.1 Security Assertion Markup Language

Security Assertion Markup Language (SAML) is a standard language (Coyle, 2007) of the Organization for Advancement of Structured Information Science (OASIS). SAML is an XML-based standard for exchanging authentication and authorization data between security domains, that is, between an IP (a producer of assertions) and a SP (a consumer of assertions). The IP is the involved party that produces the assertions and the SP is going to consume these assertions. SAML lets users to make assertions concerning their identities, attributes... in order to communicate with other parties like companies, applications, and so on (Madsen, 2004). One of the problems that this language tries to solve is the SSO in order to enable federated and secure web service transactions. SAML provides a standard solution for Federated Identity Management systems. SAML allows making assertions between an IP and a SP. The IP provides the identity credentials for an entity and the SP relies on the IP to identify this entity. However, SAML does not provide the implementation of any party.

### 2.9.2 SAML Components

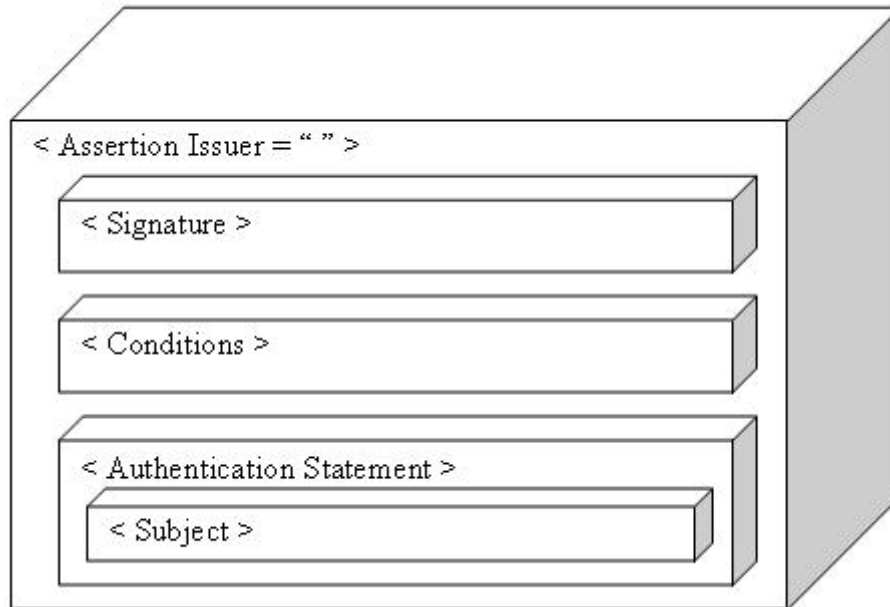
SAML is composed of the following components. The following sections outline these.

#### Assertions

An assertion is one or more statements made by a SAML authority and it is contained within a package of data (Shakir, 2007). Figure 2.9 shows a SAML statement, it can contain three different types of SAML statements (Madsen, 2004):

- **Authentication:** this statement specifies that the subject was authenticated for a particular purpose at a specific time.
- **Attribute:** this statement specifies that the particular subject is related with the supplied attributes.
- **Authorization Decision:** this statement specifies if the particular subject has been accepted or denied to access the resource.





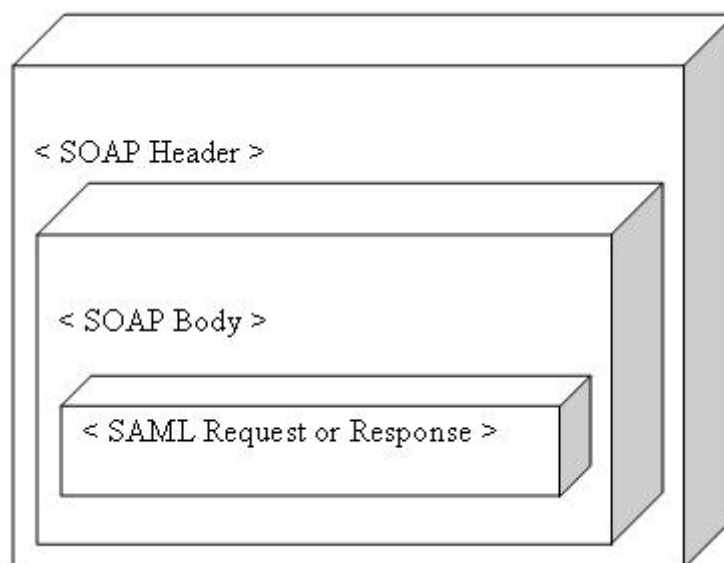
**Figure 2-9: SAML Authentication Statement**

### Protocols

SAML defines a XML-based data structure in order to represent request and response messages. The request message specifies the elements that are required in the response message (Madsen, 2004).

### Bindings

A SAML binding specifies how SAML messages are encapsulated inside a standard message or communication protocols. For example, the SAML SOAP Binding specifies how a SAML message can be sent within a SOAP message (Madsen, 2004). SOAP message is a XML document that contains a number of elements as envelope, header, body and fault. Figure 2.10 shows the SAML message within a SOAP message.



**Figure 2-10: SAML Message**

## **Profiles**

A SAML profile defines how SAML Assertions, Protocols, and Bindings are joined in order to support a particular application, with the aim to enhance interoperability (Madsen, 2004).

### **2.9.3 WS-Security**

WS-Security is a standard that can be used to build Secure Web Service applications. WS-Security defines a series of SOAP elements that contains the specifications to implement messages that require authentication, integrity, and have a confidentiality level. This standard was developed first by IBM and Microsoft. This standard is also known as WS-\* or WSS (Madsen, 2004), and describes how to attach security information such as digital signatures, encrypted data, and security tokens to the SOAP header of the message. A Security Token is defined as one or more claims. These claims are defined for the issuer and signed by an IP, in order that the recipient can trust the received message (Bhargavan, 2005).

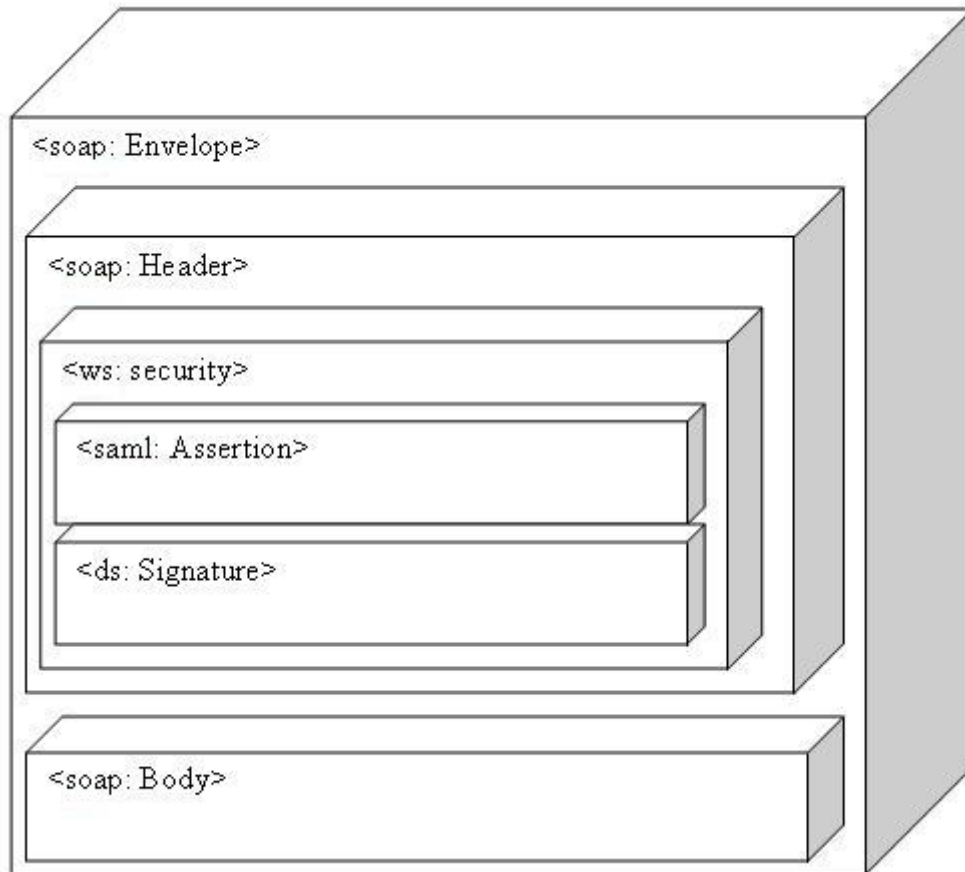
WS-Security also provides support for binary security tokens such as X.509 certificates and Kerberos tickets and XML based tokens as SAML assertions. As well, WS-Security works in the Application Layer in order to provide end-to-end security.

#### **WS-Security SAML Token Profile**

SAML Token Profile describes how to integrate SAML assertions into WS-Security header blocks. The <ws:Security> message is built within a SOAP message. This guarantees the validity of the claims from the issuer. The claims are integrated within the SAML assertion and signed with a digital signature. Figure 2.11 shows the SAML Token Profile structure (Madsen, 2004).

#### **WS-Policy**

WS-Policy specification allows Web Services to specify a number of constraints and requirements by means of their policy assertions. WS-Policy defines the relation between policy assertions, but does not define any assertion (Iacono, 2008). The WS-Policy specification defines end points in order to retrieve constraints and requirements from the Web Services and how to associate policies with services and end points. For example: supported encryption algorithms, required security tokens, privacy rules, and so on.



**Figure 2-11: WS-Security SAML Token Profile**

### **WS-SecurityPolicy**

WS-SecurityPolicy specification is an extension of WS-Policy specification and specifies a number of security policy assertions, which are utilized by the WS-Security, WS-Trust and WS-SecureConversation specifications (Iacono, 2008). Integrity and confidentiality assertions describe what part of the message we have to protect. Token assertions inform the requestor which security tokens are required to invoke a service. There are different specifications in order to implement policies, which are defined in the following sections.

### **WS-PolicyAttachment**

WS-PolicyAttachment specification is used in order to connect policies with Web Service or XML data and describe how they can be referenced from Web Service Description Language. Web Service Description Language (WSDL) is a specification used to describe a Web Service (Iacono, 2008).

### **WS-MetadataExchange**

WS-MetadataExchange is a Web Services protocol specification, which allows retrieving metadata associated with a Web Service endpoint. This specification can be used to retrieve security policies from Web Services (Iacono, 2008).

## WS-Trust

WS-Trust specification is an extension of WS-Security. This specification establishes the concept of a security token service (STS), it is a web service that can issue, renew and validate security tokens. WS-Trust can be used to establish, assess the presence of, and broker trust relationships between different parties in a secure message exchange (Iacono, 2008). Some features defined by WS-Trust are:

- Security Token Service (STS). A Web Service that issues security tokens as defined in the WS-Security specification.
- The formats of the messages used to request security tokens and the responses to those messages.
- Key Exchange Mechanisms.

## WS-SecureConversation

WS-SecureConversation specification is used on top of WS-Security, in order to provide secure mechanisms for multiple message exchanges. It introduces a security context and specifies extensions for security context establishment and sharing, beside session key derivation. A Secure Context Token (SCT) is used to obtain the session keys. This mechanism is used in order to provide a security context (Iacono, 2008).

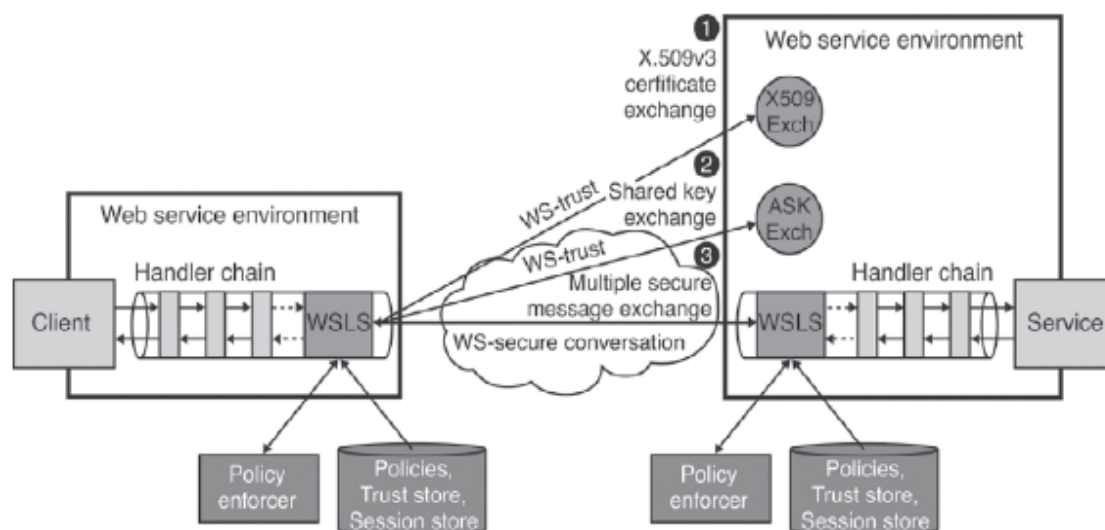


Figure 2-12: Web service layer security (Lo Iacono, 2008)

### 2.9.4 WS-Federation

Web Services Federation is an Identity Federation specification. WS-Federation is a part of WS-Security and others Web Services Security frameworks. It is used along with SOAP messages to improve the quality of protection within message integrity, message confidentiality, and message authentication. WS-Security provides a mechanism in order to associate a Security Token (like X.509 certificates, Kerberos tickets or XML tokens such as SAML) within a message (Demchenko, 2004). WS-Federation specification can be used to build a Federated Identity Management framework. WS-

Federation enables the use of authentication and authorization within a circle of trust, for example a number of Service Providers.

WS-Federation is extended from WS-Trust specification to define how Identity Providers can issue Security Tokens and how the identity information and attributes are incorporated within a Security Token. Service Providers can use tokens in order to request identity information from a user. WS-Federation can provide Identity Services by means of active requestors such as SOAP enabled applications or passive requestors such as Web browsers (Demchenko, 2004). WS-Federation enables the creation of secure federated Web Services. These Web Services can provide service to clients registered within a dominion of trust. It can also help to preserve anatomy, trust relations and defend user privacy. WS-Federation contains a number of elements used to establish the federation identity model:

### **Security Token Service**

Security Token Service (STS) is used in order to listen user requests and to provide security tokens using a common model. This security token can then be used with any service that trusts the IP (Demchenko, 2004). In order to establish a secure conversation one or several Security Tokens are sent. The Security Token contains a set of claims and a certificate to prove its validity. The set of claims contain information such as name, identity, address, and so on.

### **Identity Provider (IP)**

The IP represents an entity that provides authentication to the user in order to establish a communication between this user and a SP (Demchenko, 2004). The IP issues Security Tokens with security information about the user. It can be considered an extension of a Security Token Service.

### **Attribute Service**

Attribute Service is a Web Service that contains information or attributes about the users within the Federated Identity Management framework. The IP can act like an Attribute Service keeping information or attributes about a user or entity.

### **Pseudonym Service**

Pseudonym Service is a Web Service that contains alternative information about a user or entity within the FIM framework.

## **2.10 Sample of Federated Identity Management**

---

Figure 2.13 shows an example that is formed by four different parties; the entity or requestor, the Identity Provider for that entity, the resource or SP and the Identity Provider for that SP. The Resource requires that the requestor provides a set of claims in order to obtain a service. The resource is implemented by a Web Service that specifies a number of claims or information required to obtain the service. These set of claims are published following the WS-Policy and WS-PolicyAttachment specifications (Iacono, 2008).

The requestor can issue a Security Token with the required information within the message and apply a certificate in order to show its authenticity. This message is sent to the resource. If the requestor does not have the necessary Security Token to prove their authentication, he can contact the Identity Provider to obtain the required Security Token.

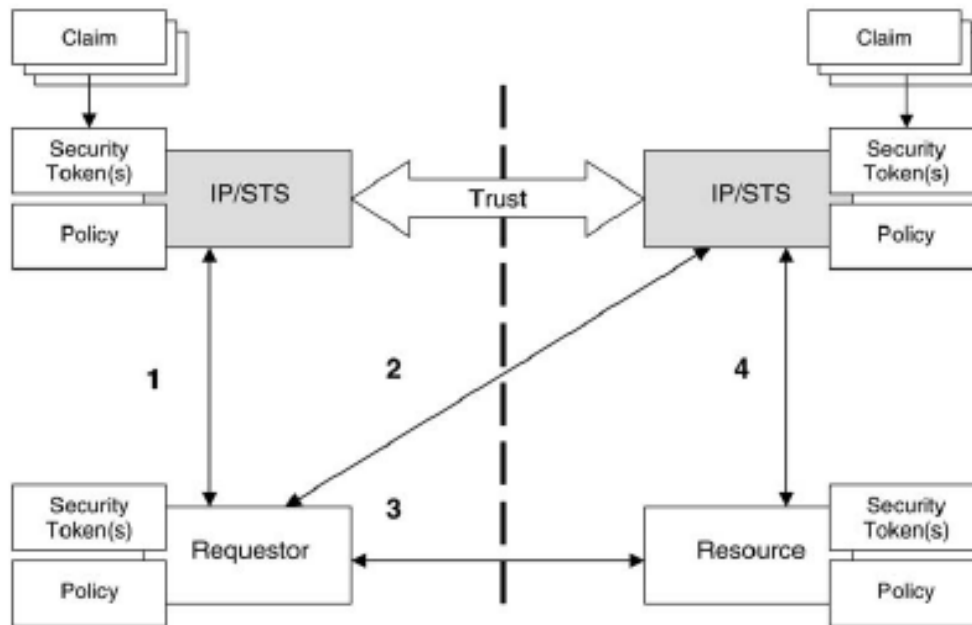


Figure 2-13: Sample of FIM (Demchenko, 2004)

In Step 1, the requestor requests a Security Token from the requestor's Identity Provider, who provides the requestor's identity information within a Security Token. In Step 2 the requestor uses the obtained token to request the necessary Security Token from the resource's Identity Provider to access the resource. Finally in Step 3 the requestor issues a new security that contains the requirements to access the resource. When the resource receives the Security Token, it carries out the following operations:

1. The resource checks that the requestor has sent all the necessary claims in order to satisfy the resource's policy.
2. The resource checks that the claims have been signed with a trusted key.
3. The resource checks that the requestor has enough privileges to issue the provided claims.

In the case that all these requirements are satisfied, the resource can process the request.

## 2.11 Conclusion

This literature review has examined the problems related to the traditional identity system management that provides a service by means of username and password. This highlighted some problems, the user information is stored in the SP, the user must memorize different identifications for different services, and the information is sent in a

non-secure way. New authentication methods that attempt to resolve the problems associated with username and password authentication were then reviewed. Biometric authentication was discussed, highlighting the issues of user acceptance. Digital certificates (based on the use of public and private key cryptography) were then presented. Finally the use of security tokens for authentication was presented. This authentication is used to build a Federated Identity Management system, where the user is in control of their identity information and a third party or Identity Provider can verify this information.

The Federated Identity Management system includes the concept of circles of trust where the user's identity information can be managed by an IP in order to obtain access to different services. When the user wants to obtain a service, they connect with the IP in order to obtain the required information, then the IP returns this information and the user provide this to the service to obtain the access. One of the advantages of the Federated Identity Management model is that the user has the control of their credentials in every transaction, since this model has to satisfy the principles of security and data protection as discussed in Section 2.7, unlike the traditional identity management where the service controls the user's credentials.

The Federated Identity Management system can be divided in two different models. Relationship focused systems where the Identity Provider manages the user's identity information. When the user wants to obtain a service, the identity information is retrieved from the Identity Provider in a short-term security token and then it is sent to the service. This model has the disadvantage that the Identity Provider has to be online during the operation. The second model is credential-focused system where the user has long-term credential. In this model, the user can handle their identity credentials without involving the Identity Provider. The user has a security token with their identity information that has been obtained by the Identity Provider. This model has the disadvantage that the security token can be stolen and modified. As well, the system has a higher workload on the user side and it is a more expensive implementation for the user.

## 3 Requirements Analysis and Design

### 3.1 Introduction

---

This chapter presents a solution to the problem outlined in the literature review, and covers the hardware and software used in the software development. The chapter also covers the development tools used to create the software, and an application has been set up in a website in order to evaluate different requirements such as performance, functionality and usability.

### 3.2 Identity Management

---

This section explains how the Identity Management system was designed for this thesis. The system provides two different types of authentication, the first one is by means of Information Cards and the second one implements the traditional Identity Management through username and password. Section 3.2.2 explains the design used in order to implement Information Cards identification.

#### 3.2.1 Identity Management Requirements

The purpose of this thesis is to create a Federated Identity Management system that can share a single authentication across different systems. The advantage of this system is that the user no longer has to remember different usernames and passwords or repeat the same authentication for different services and websites. In this thesis, CardSpace technology (Bertocci 2007) is used to build a website that integrates the Identity Management system. Microsoft provides this functionality as part of the .NET Framework 3.0 (McMurtry, 2007). CardSpace can be integrated into services and web applications (Wolfgang, 2005). In the test website, the user can provide their credentials by means of:

- Username and password authentication, where the user will store some information in the website database, or
- By creating an Information Card, which contains a number of claims about the user.

If the user decides to use an Information Card, he has to choose between two different types of cards:

- **Personal Cards:** the user self asserts the information provided in the card. This type of card can be thought of as a business card, where the person providing the card only verifies the information.
- **Managed Cards:** another party (IP) asserts a number of claims with the user information.



In order to create a communication between the user and the IP, SOAP messages have been used along with WS-Security protocols. This communication exchanges the user's identity information within the Security Token. The complexity of this thesis has increased with the need to build the IP along with Security Token Server (STS) in order to provide Managed Cards authentication. In a real application, the IP would be a well-known company, independent from the relying party so that the user does not have to trust the identity information to the relying party.

### 3.2.2 Identity Management Design

In the developed website three different roles can be distinguished: the user, the relying party and the IP.

- **The user** is the part that will request the service from the relying party. In order to provide their identity, the user has to obtain a security token with identity information about themselves.
- **The relying party** is the part providing a service to the user. The relying party will request an identity token containing a number of claims in order to identify the user.
- **The IP** is the part that provides the identity token with the information of the user. The IP is a well-known system supplying identity information to the relying party to authenticate the user. The IP has to present a certificate to the token to authenticate themselves.

Figure 3.1 demonstrates the following steps involved in the authentication process of the website:

1. The user tries to establish a connection to the website.
2. The browser sends an HTTP GET to the relying party in order to connect to the login page.
3. The relying party returns to the HTML login page. This page contains an OBJECT tag holding the policy of the relying party. This OBJECT tag is linked to the login button in this page.
4. The user presses the login button. Afterwards, the Identity Selector of CardSpace is prompted to evaluate the policy contained in the OBJECT tag. The identity selector shows in colour the information cards that meet the policy and greys out the cards that do not.
5. The cards are presented to the user.
6. The user then selects one information card to provide their credentials.
7. Then the identity selector requests to the IP to retrieve the security policy to obtain the security token
8. The IP returns its security policy.
9. The user provides their credentials to the IP and request a security token with the required claims.

10. The IP returns the security token to the user.
11. The browser sends an HTTPS POST to the relying party.
12. The relying party returns a cookie and redirects the user to the home page.
13. The browser sends an HTTP GET to the relying party to obtain the home page URL.
14. The relying party returns the home page to the browser.
15. The use is now authenticated and the home page is shown to the user.

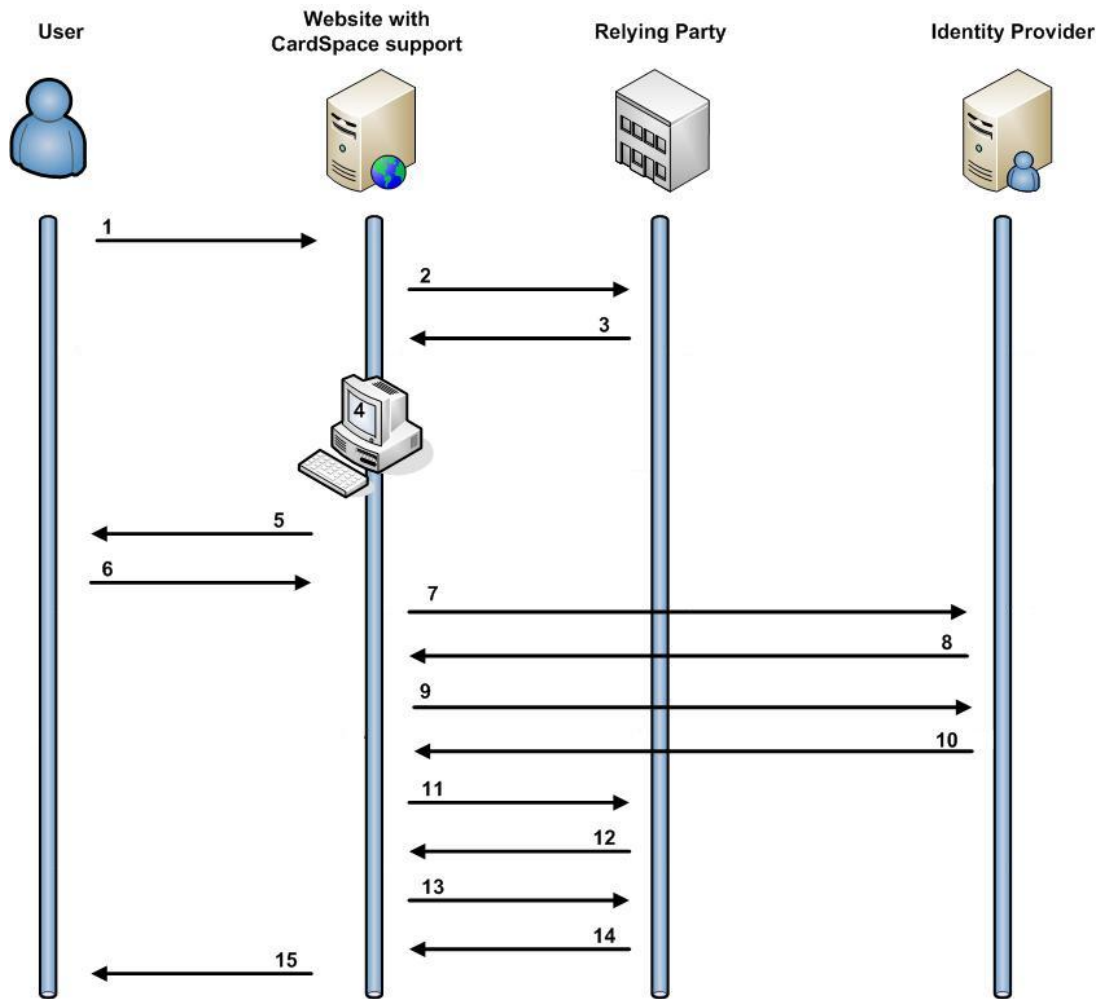


Figure 3-1: Login Page with CardSpace Support

### 3.3 Database

This section explains how the information is managed in the system. All the information of the user account is stored in a database in the SQL Server. The following sections outline the database requirements, and then follow the design of the database.

#### 3.3.1 Database Requirements

The database must be able to store the following information:

- User information, such as identification or username.

- Membership information about the users such as password, email or creation date
- Information about the information cards, so the user can link an account with their information card.
- Information about different roles so users can have different privileges, as they can be administrators, editors or basic users.

### **3.3.2 Database Design**

The database has been created in SQL Server 2005 Express. The different tables are shown in Figure 3.2, explaining the relationship between the tables. In the registration procedure on the website, the information is stored in two different tables in the database. The username is stored in the table Users, and one user ID is assigned to the new user. This ID is going to be the foreign key of the Membership table, and this will in turn save the rest of the information provided in the register form. Once the user has been added to the database, he will be assigned to one specific role. Each different role is identified with one role ID in the table of the roles, so that when one user is assigned to one role, this keeps the relation with the user ID and the Role ID in the UsersInRoles table.

In the website, the process of registering one user in the database has been divided into two different methods:

1. In the first one the user has to complete an application form with all the information required for the website. This is the traditional authentication method, where the user has to write all the information manually.
2. The second registration method is by means of Information Cards. The user can select a card containing the number of claims required by the website. In this second method the Private Personal Identifier from the Information Card is stored in the UserInformationCards table. This table keeps a relation with the Membership table by means of UserID column that is foreign key in the UserInformationCards table.

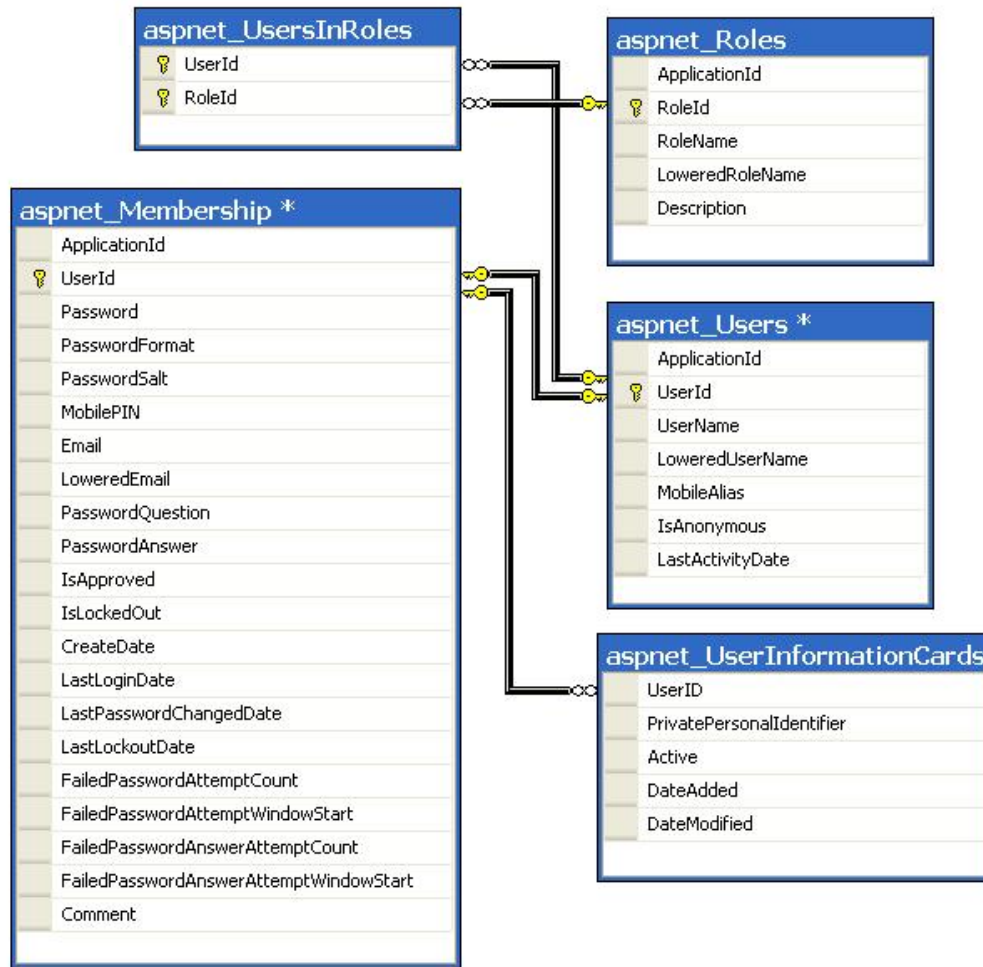


Figure 3-2: The Database Relationship

### Database Queries and Stored Procedures

A number of queries have been created to make different operations in the database. These queries are created and executed by a number of stored procedures. Some of the operations made by these stored procedures are:

- Insert new users in the database.
- Retrieve some information in order to associate one user with their information card in the database.
- Create relations between tables.
- Find information about one user.
- Add user information card.
- Add Role information and so on.

Some of these stored procedures have been obtained from the functionality provided by ASP .NET in its membership system. These stored procedures have been modified to adapt their functionality to the systems requirements. Other stored procedures have been

created specifically for this thesis. Stored procedures are being employed due to these being modular and easy to change without having to change the application code. Another advantage of using stored procedures is that it speeds up performance. Figure 3.3 shows the stored procedures that have been used in the application. Figure 3.5 demonstrates the tables used in the application. Some of these tables have been obtained from the membership system and others have been created specifically for the purpose of this thesis.

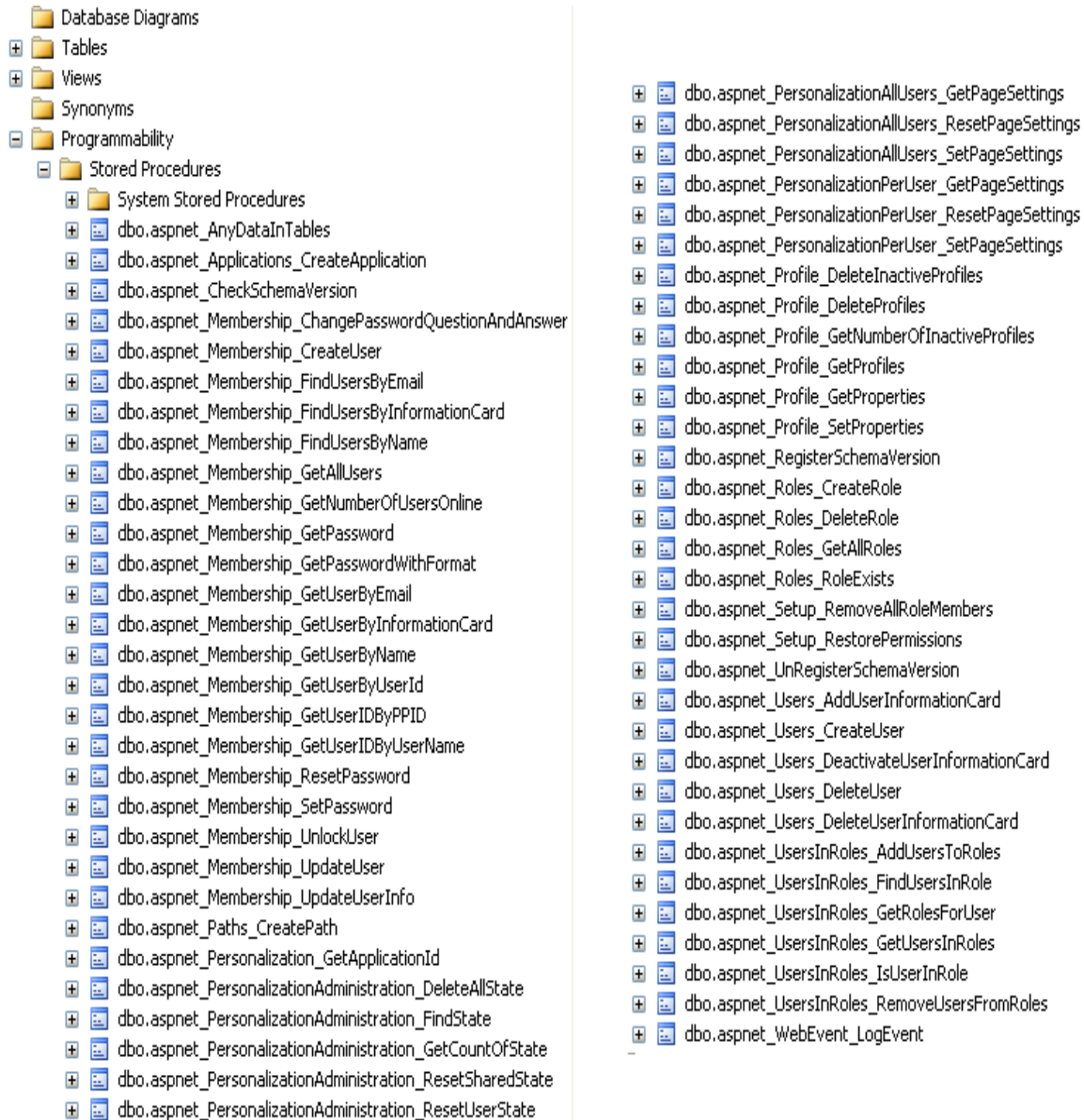


Figure 3-3: Database Stored Procedures

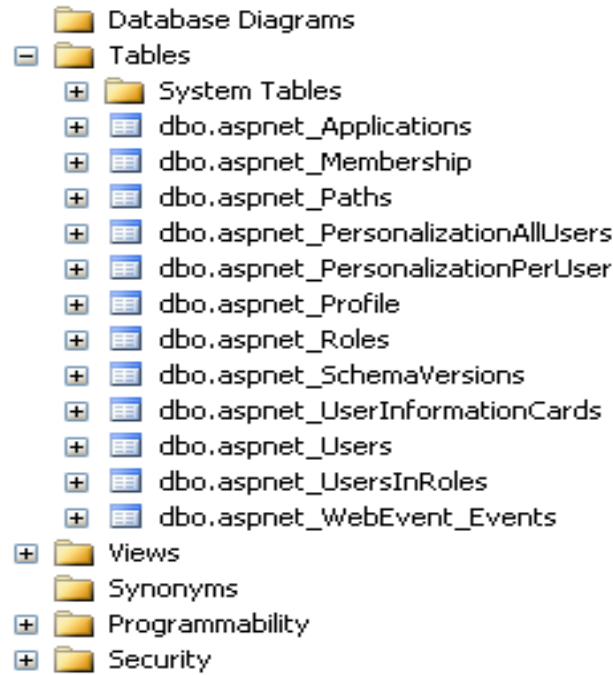


Figure 3-4: Database Tables

## 3.4 Communication

This section explains the necessary requirements to set up the system with a number of communication and security protocols. Section 3.4.2 shows how the web site has been hosted in a web server. This section also explains how the security certificate has been designed and how this certificate is used in the communication between the different parties.

### 3.4.1 Communication Requirements

This thesis aims to create a federated identity management system. The federated identity management system permits the use of Single Sign On (SSO) and results in the elimination of the use of passwords. For the construction of this system, a number of security and communication standards are used:

- A Web Server to host the application.
- A public key certificate to sign the communication between the different parties in our application.
- A standard language to exchange information between the different parties.
- A standard that can be used to build and keep the security in the web service application.

### 3.4.2 Communication Design

#### Web Server

The project has used Internet Information Service (IIS) to host the web application (Khosravi, 2008). IIS provides a set of internet services for the use of web servers. IIS permits the use of standards protocols as FTP, SMTP, HTTP/HTTPS, Java Script, and so on. Figure 3.5 shows the information stored in the Web Server.

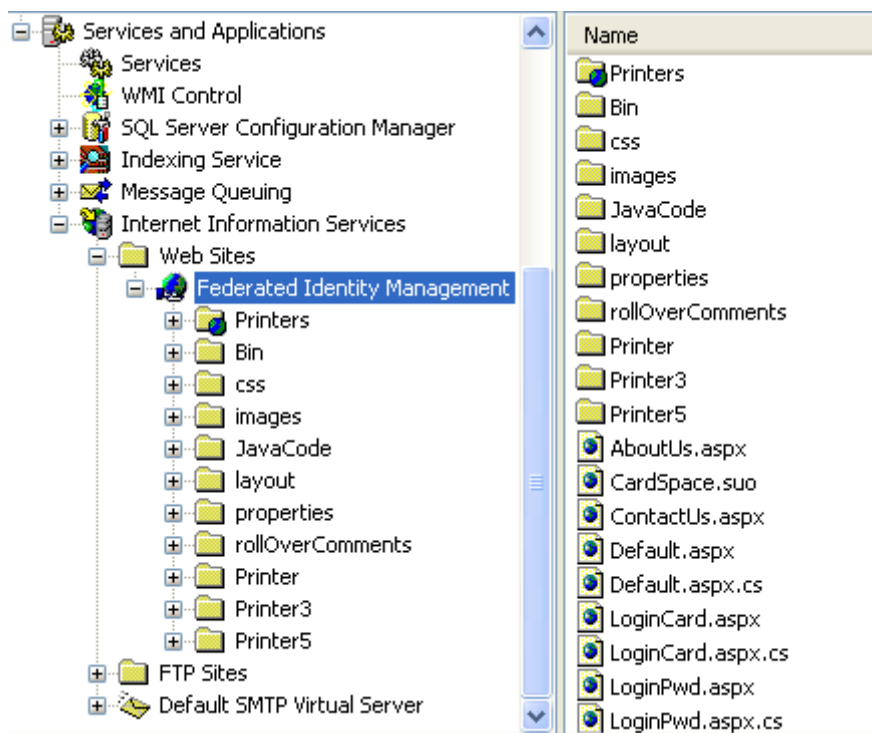


Figure 3-5: Internet Information Service

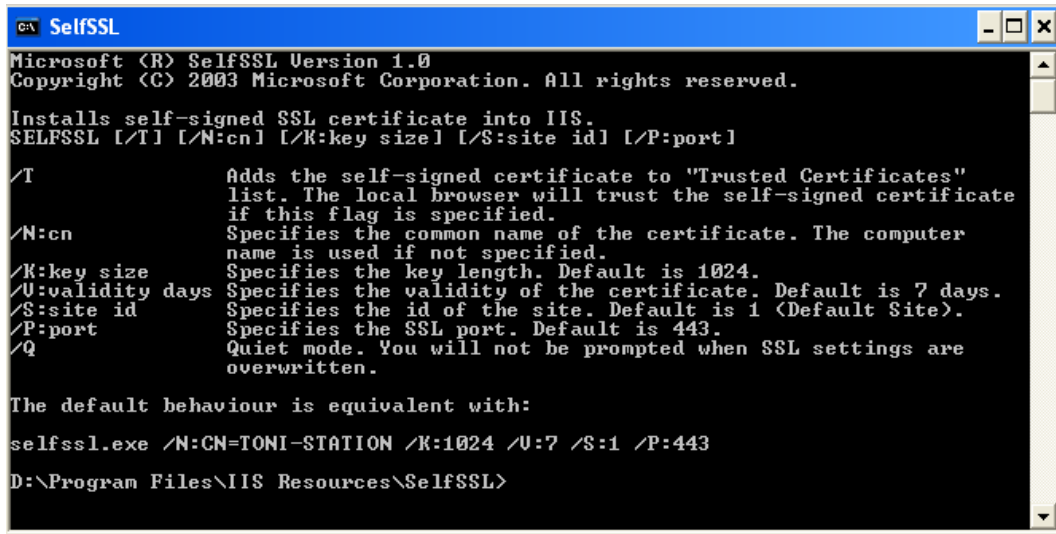
#### Hypertext Transfer Protocol over Secure Socket Layer

The website requires hypertext transfer protocol over Secure Socket Layer (HTTPS) in order to increase the security between the different parties. Unlike the HTTP protocol that uses the standard port 80, the HTTPS communication is established by means of the TCP port 443. The communication is established by an encrypted secure sockets layer, this is going to protect the communication from the man in the middle attacks (Asokan, 2005).

A security certificate has to be obtained in order to use the HTTPS protocol for the application. Figure 3.6 shows the SelfSSL application used to create the security certificate. Once the certificate has been created, this has to be installed in the Microsoft Management Console (MMC). To do this, the Snap-in has to be added for Certificates in the Microsoft Management Console and import the certificate to the Personal Certificates folder. After this step, the created certificate can be added to the IIS. The details of the certificate used for the website are shown in Figure 3-6.

Another reason why a secure certificate is being used for the communication is that CardSpace sends the security token encrypted, protecting the information of the card.

The certificate will be used to sign the XML token, afterwards the security token will be exposed to the share point.



```
c:\ SelfSSL
Microsoft (R) SelfSSL Version 1.0
Copyright (C) 2003 Microsoft Corporation. All rights reserved.

Installs self-signed SSL certificate into IIS.
SELFSSL [/T] [/N:cn] [/K:key size] [/S:site id] [/P:port]

/T          Adds the self-signed certificate to "Trusted Certificates"
           list. The local browser will trust the self-signed certificate
           if this flag is specified.
/N:cn       Specifies the common name of the certificate. The computer
           name is used if not specified.
/K:key size Specifies the key length. Default is 1024.
/U:validity days Specifies the validity of the certificate. Default is 7 days.
/S:site id  Specifies the id of the site. Default is 1 (Default Site).
/P:port     Specifies the SSL port. Default is 443.
/Q         Quiet mode. You will not be prompted when SSL settings are
           overwritten.

The default behaviour is equivalent with:
selfssl.exe /N:CN=TONI-STATION /K:1024 /U:7 /S:1 /P:443

D:\Program Files\IIS Resources\SelfSSL>
```

Figure 3-6: Creation of the Security Certificate

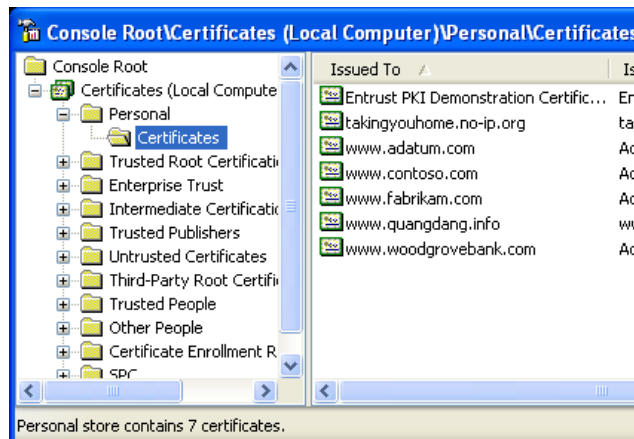


Figure 3-7: Microsoft Management Console



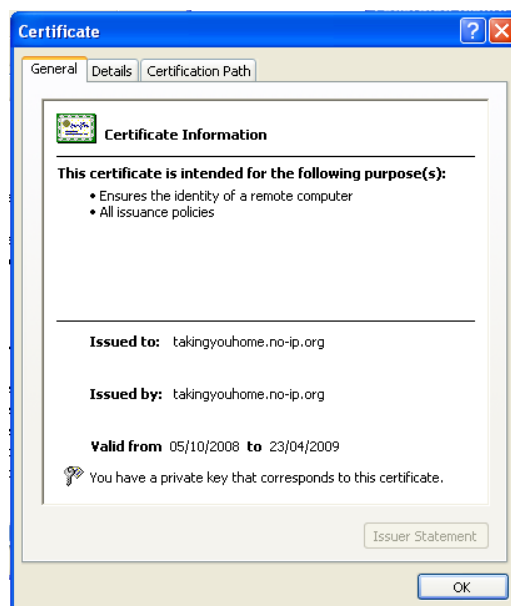


Figure 3-8: The Website Security Certificate

### Security Assertion Markup Language

Security Assertion Markup Language (SAML) is a XML-based standard language which allows making assertions between two different parties in a secure way. This language permits one part to create an assertion about authentication data and sending it to the other part, which is going to consume this assertion (Wolfgang, 2005). In this thesis, a security token has to be constructed containing a number of claims about the user who wants to be authenticated. The IP will then send this token to the SP (this is the website application) and the SP will rely on the assertion made by the IP (Shakir, 2007).

### WS-Security

WS-Security provides a standard protocol to build secure web service applications. WS-Security also provides a series of SOAP elements to build authentication, integrity and confidentiality messages (Bhargavan, 2005). WS-Security defines how to use SAML language in order to integrate signatures to SOAP messages and to attach security tokens. In this thesis, WS-Security has been used to create the Security Token Service which is going to create the security tokens requesting and retrieving the user identity information.

### WS-Policy

WS-Policy protocol permits the web service public to use its policy and requirements to provide the service. WS-Policy defines the endpoint to retrieve the policies and requirements. WS-MetadataExchange protocol allows retrieval of the metadata that is published in the endpoint.

## 3.5 Application

This section explains how the graphic interface has been developed in order to implement the system. Section 3.5.1 exposes the necessary requirements to build the system.

The system is a fictitious company built into a website, which exposes service for the users and permits the creation of user accounts with two different types of identification Information Cards and username/password.

### 3.5.1 Application Requirements

Since this thesis is centred on Internet and Network security, an application has to be provided, allowing the user to evaluate the technology that has been stated in this thesis. A fictitious company has to be created, which provides **some kind of service**. In order to access this service, the user has to provide some identity information to the company. To be able to compare the use of information cards with the current technology (username and password), this thesis provides the user with both technologies in the website. The web site has to supply services that are restricted to authenticated users.

At this point, this thesis will describe the graphic interface necessary in order to provide this functionality. The designs of the interfaces fulfilling the requirements of the system are the following:

#### Main Page

The website has to provide the following elements:

- At the top, there will be a website logo.
- Below the logo, a menu will be created with the information of the website.
- On the left hand side, a menu has to be created with the products provided by the website.
- Below this menu, some information about the customers who have used the website will be shown.
- In the middle of the website, a frame has to be created which will change depending on the information being showed.
- On the right hand side, two different menus will be displayed. One menu has to permit the user to register and login to the website with their information card and the other menu with their username and password.

The interface will have a design similar to the one showed in Figure 3.9.

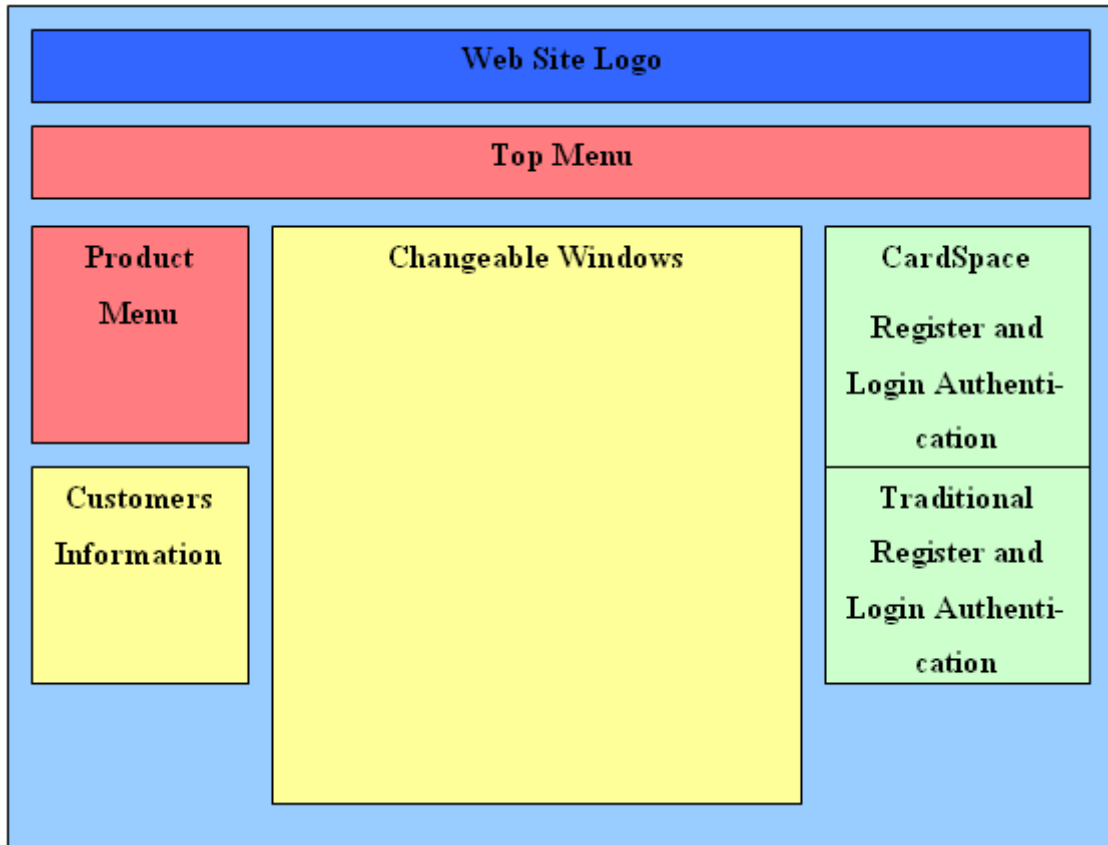


Figure 3-9: Main Page Design

### Login and Registration Menu

The login and registration menu has to provide two buttons linked with the pages providing these operations. The menu is defined in Figure 3.10. The information card and username menus work in the same way, with the difference that they are linking the user to different pages. I.e, if the user has selected to use the information card, this menu will link the user to the information card page.

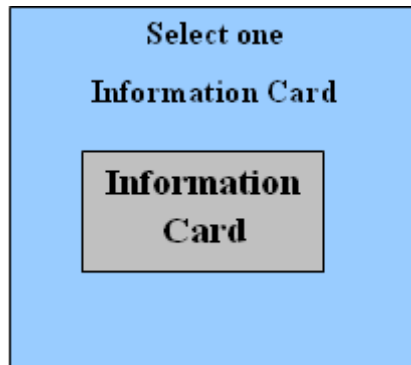


Figure 3-10: The Login and Registration Menu

### Information Card Interface

If the user has chosen to enter their credentials by using the information card, this interface has to permit the user to access the CardSpace selector. If the user presses the button, the CardSpace selector has to be triggered. Figure 3.11 shows the information card

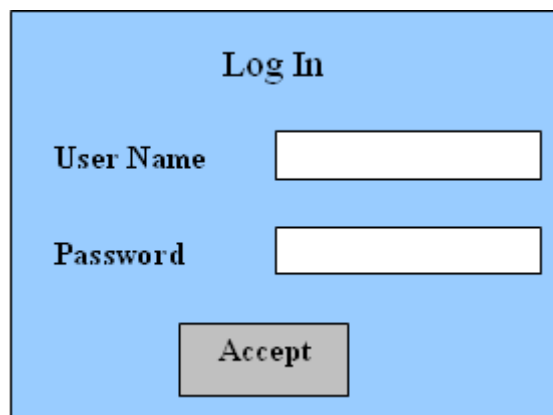
page with the interface. The login and register pages will have the same interface with the difference that the information card could be required with different information, since the user, when being registered, has to provide more information than when the user is logged on the web site.



**Figure 3-11: The Information Card Interface**

### **Username and Password Login Interface**

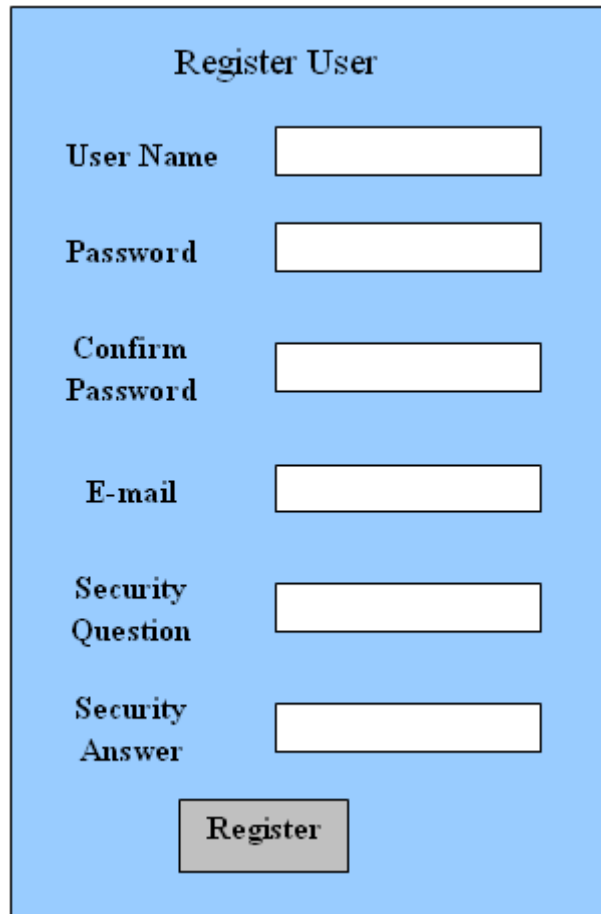
The username and password interface has to provide two text boxes to introduce the user authentication and one button to send the information to the server. One of the advantages of using HTTPS protocol is that the communication is encrypted by means of the website certificate, so the user information will be protected against phishing. Figure 3.12 shows what the interface has to provide.



**Figure 3-12: Login with Username and Password**

### **Username and Password Register Interface**

The username and password register interface has to provide different text boxes in order to provide the registration information required by the website. In the same way as on the Login page, the information is being sent to the server using HTTPS protocol, protecting the user against phishing attacks. Figure 3.13 outlines this interface.



The image shows a web form titled "Register User" on a light blue background. The form contains six input fields, each with a label to its left: "User Name", "Password", "Confirm Password", "E-mail", "Security Question", and "Security Answer". Each label and its corresponding input field are vertically aligned. At the bottom center of the form is a grey rectangular button with the text "Register" in black.

Figure 3-13: Username and Password Register Interface

### 3.5.2 Application Design

#### Main Page Website

The website has been designed to sell properties in different parts of the world. The programming language used to develop this prototype was ASP .NET (MacDonald, 2005). The website permits the registration of different users through information card or username and password. All the information shown in the website has been invented for this thesis. The website has been divided into the following parts:

- Below the logo we have created a menu with a link to information about the company, providing data such as contact details, company policy, and so on.
- On the left hand side, a menu has been created with the countries having properties for sale.
- Underneath this menu, some fictitious testimonies are shown from customers who have used the website.
- In the middle of the page, a changeable window has been created which will change depending on the information showed.

- On the right hand side, a menu has been created containing two different buttons. These two buttons link the login and registration pages together for the Information Cards.
- On the same side below this menu, a similar menu has been created in order to provide login and registration to the user through username and password credentials. The first button connects the user to the login page and the second button with the registration page.

Figure 3.14 demonstrates the home page in the created web site.



Figure 3-14: Main Page Website

### Information Card Login and Registration Page

Figure 3.15 shows the login page with information card. On the top left side of the picture a magnifying glass demonstrates that the page has to use a security certificate in order to use the information cards. On the top left side of the picture, another magnifying glass is showing a padlock, verifying that the page uses a verified certificate. When the user presses the roll over button in the middle of the page, CardSpace selector is triggered and the rest of applications are locked in the background. It is a protection that Net Framework 3.0 provides in order to avoid any modifications of the security token. The identity selector greys out the information cards that do not meet the website's policy and show with colour the information cards that success this policy.

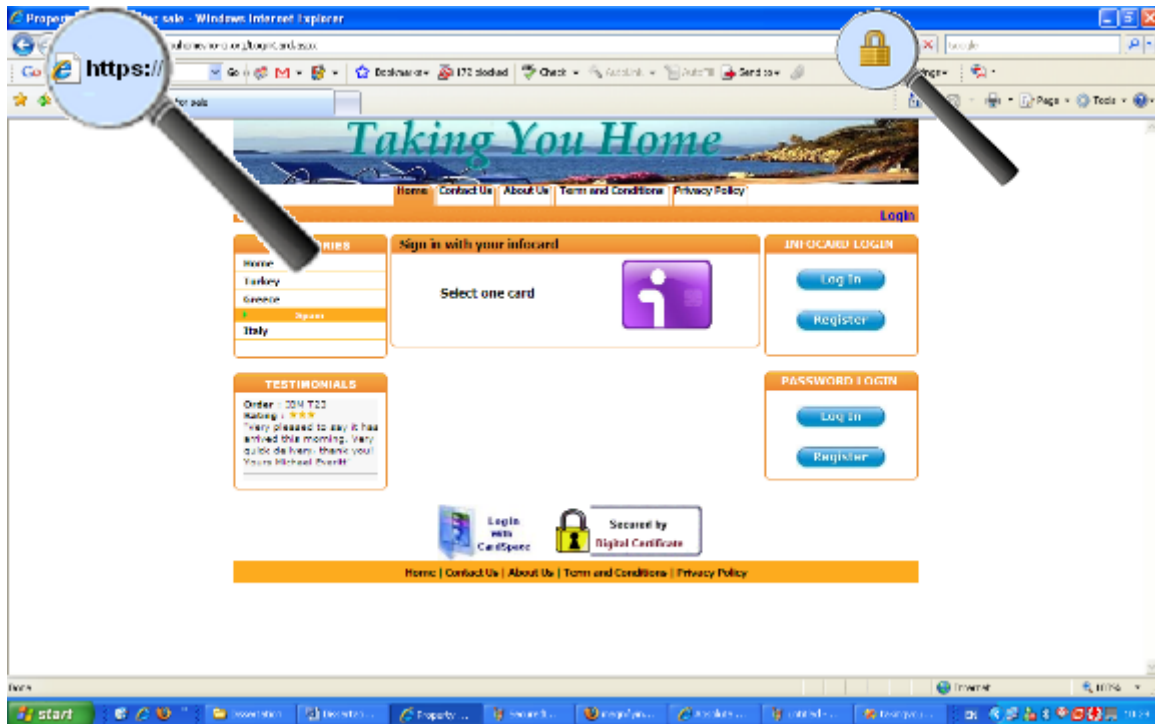


Figure 3-15: Information Card Login Page

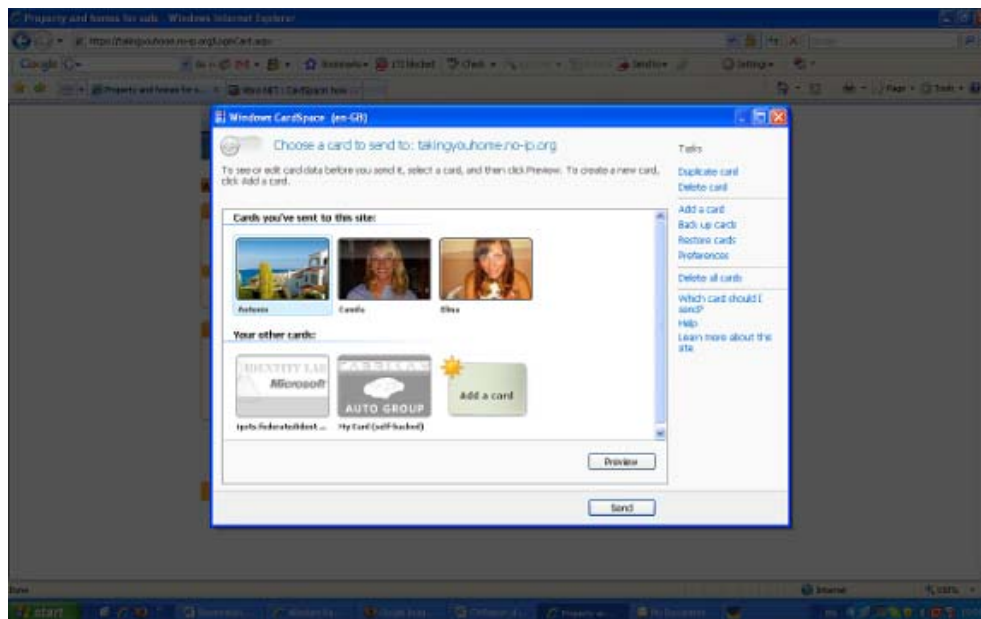


Figure 3-16: CardSpace Identity Selector

### Username and Password Login Page

The username and password page provides a form to introduce the username and password authentication and by clicking a button sends the information to the server. The information is sent to the server using the website certificate by means of Secure Socket Layer protocol (SSL). This kind of communication is not used by every website therefore, in some occasions, the user has to send the identity information unprotected and

vulnerable to attacks from the man in the middle (Asokan, 2005). Figure 3.17 shows the login page.



Figure 3-17: Username and Password Login Page

### Username and Password Register Interface

The register page provides different text boxes in order to provide the registration information required by the website. In the same way as on the login page, the information is sent to the server using the website certificate by means of https protocol. Figure 3.18 demonstrates this registration page. Since some users might find it difficult to change the way that they provide their credentials, this registration page provide an optional opportunity to associate an Information Card with the new user account. This means that the user can first register with the technology and then start to change the way that he provides their credentials.



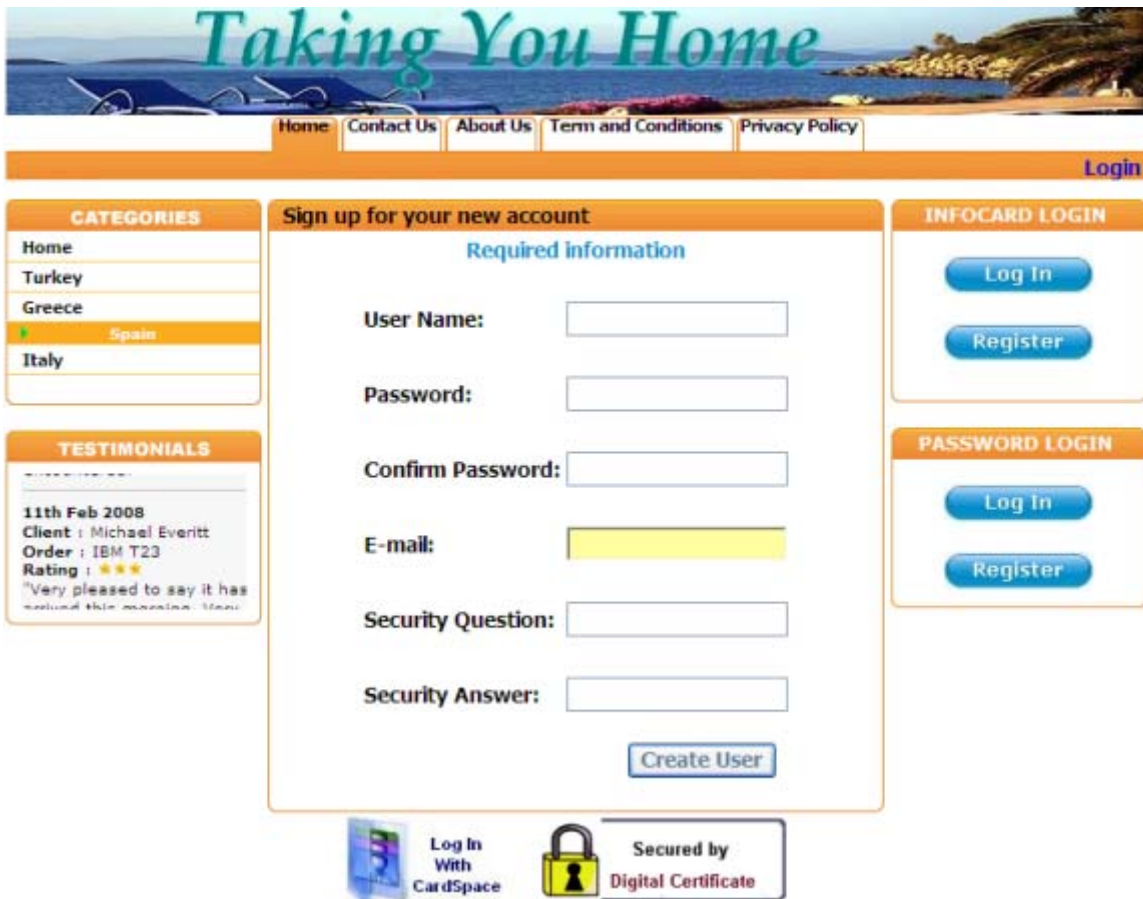


Figure 3-18: Username and Password Register Page



Figure 3-19: Associate an Information Card Page

## 3.6 Identity Provider

---

This section explains how the Identity Provider has been designed in order to provide the user authentication (McMurtry, 2007). The Identity Provider implements a Security Token Service, which receives the user's request in order to return the security token. Section 3.6.2 illustrates how the security tokens have been built and how the user can connect with this to obtain a service.

### 3.6.1 Identity Provider Requirements

This thesis provides two different solutions to handle the user authentication:

1. **Personal card:** the user controls their own identity information. The security token is being sent from the user's computer to the Relying Party.
2. **Managed Card:** the user's identity information is handled by an Identity Provider. The Identity Provider issues the information cards of the user and implements the Security Token Service (STS). The Security Token Service will receive the security token request and create the security token, which in turn will be sent to the Relying Party.

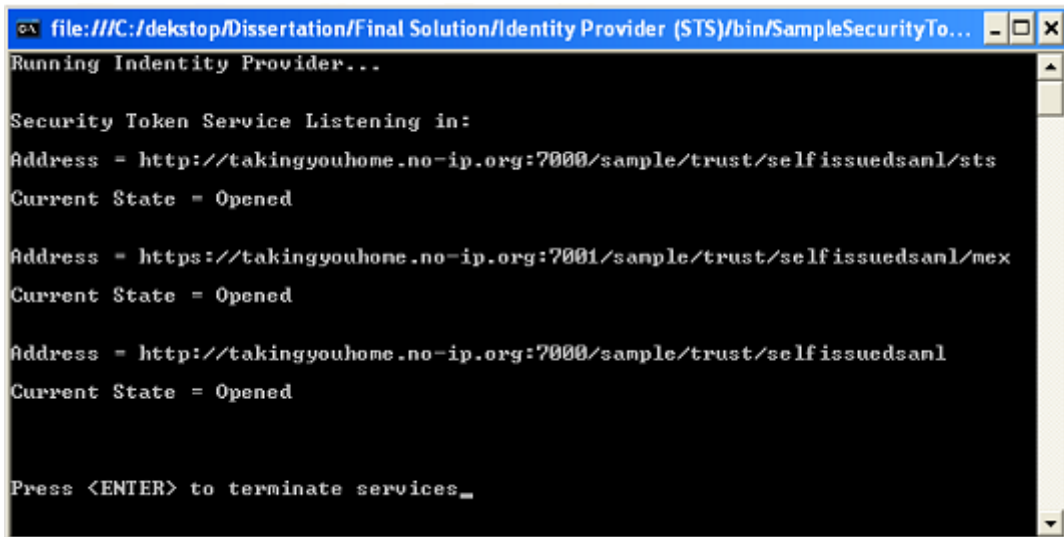
One of the disadvantages of personal cards is that the user information can only be signed by the user itself. It is a problem in the digital world, since some of the Relying Parties will only accept the identity information if this has been sent by a well-know party.

In this thesis a fictitious Identity Provider that issues security tokens is going to be developed. The Identity Provider has to implement a Security Token Service that manages identity tokens request from the user and returns identity tokens with the user's identity information.

### 3.6.2 Identity Provider Design

Information Cards provides the user with a way to manage their identity information. The information card contains a number of claims representing the user's identity information. This information can be kept in the user's computer in the case of a personal card or this can be kept in the IP server in the case of a managed card.

This thesis has been developed with an IP that is waiting to receive a request from the user. The Identity Provider and the user establish a communication using WS-Security protocols in order to exchange information. The IP runs a service called Security Token Server (STS) (Demchenko, 2004). The Security Token Serve attends the request from the user and provides the security tokens.



```
file:///C:/dektop/Dissertation/Final Solution/Identity Provider (STS)/bin/SampleSecurityTo...
Running Identity Provider...

Security Token Service Listening in:
Address - http://takingyouhome.no-ip.org:7000/sample/trust/selfissuedsan1/sts
Current State = Opened

Address - https://takingyouhome.no-ip.org:7001/sample/trust/selfissuedsan1/mex
Current State = Opened

Address - http://takingyouhome.no-ip.org:7000/sample/trust/selfissuedsan1
Current State = Opened

Press <ENTER> to terminate services_
```

**Figure 3-20: Identity Provider**

The Security Token Server establishes a number of endpoints:

- A HTTP address to host the WS-Trust endpoint. The WS-Trust endpoint provides support to issue, to renew or to validate the security tokens. The WS-Trust also validates the communication between the parties in a secure message exchange.
- A HTTP address to host the MEX endpoint. The MEX endpoint provides a HTTP address for the STS Metadata Exchange. The Metadata Exchange is used to expose the metadata that is going to describe a service.
- A HTTP address to host the Security Token Server.

The login process with a managed card is identical to the process with a personal card. When the user tries to connect to the website, the Identity Selector shows the information cards that have successfully met the website policy, and these will be displayed in colour. The user then selects the managed card to provide their credentials. The user has to provide their credentials to the IP in order to request a security token with the required claims. When the user provides their credentials, (smart card, personal card or user/password, and so on.) a request to obtain the Security Token is sent to the STS and this will send back a security token with the required claims. The user then sends the security token to the relying party in order to obtain the service.

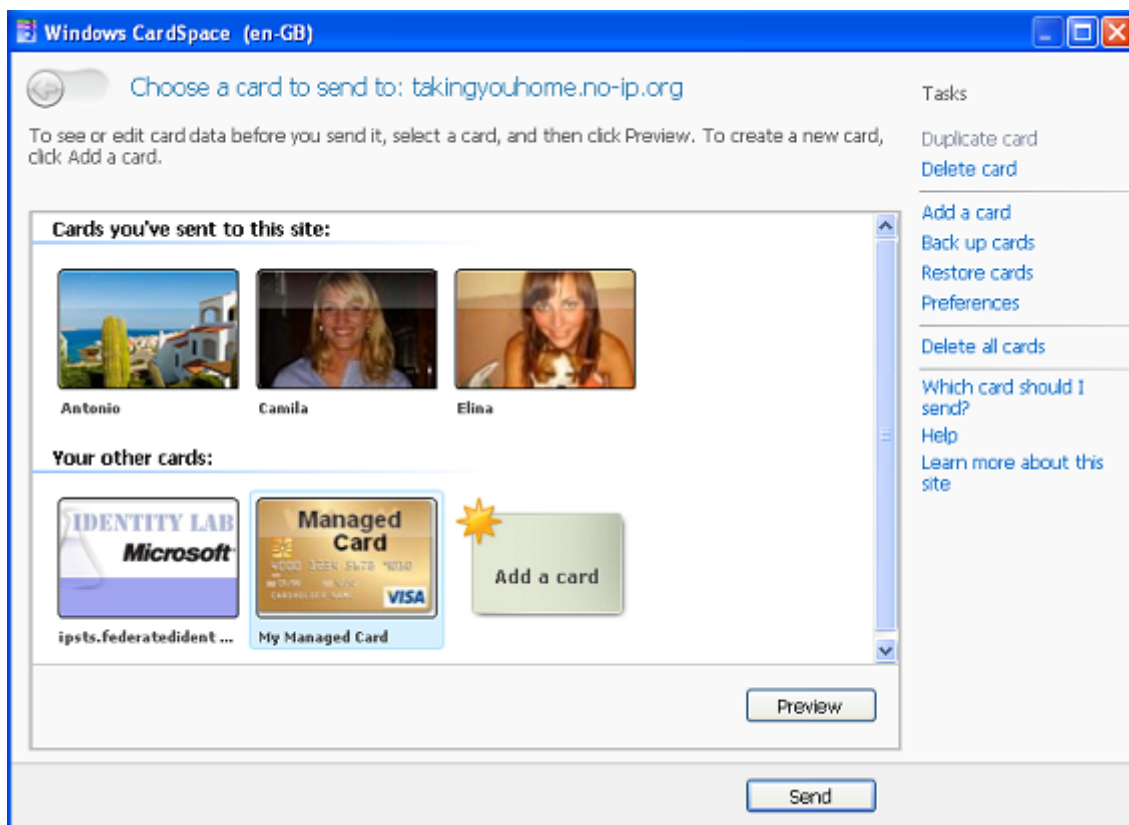


Figure 3-21: Managed Card

### 3.7 Conclusion

This chapter has demonstrated how the FIM system has been implemented in the prototype. CardSpace and .NET Framework provide the tools necessary to implement this system. The use of Information Cards permits the website to provide all the information required to obtain the service. In addition to this, CardSpace also provides a good graphic interface that provides the user with all their Information Cards satisfying the website policy, which in turn satisfies the Identity Law of Minimal Disclosure (see Section 2.6.2). Additionally, CardSpace also warns the user about the identity information released to the website, in this point the user can make the decision whether or not he/she wants to provide the information. This satisfies the Identity Law of User Control and Consent (see Section 2.6.1). CardSpace also shows the user who is going to receive the Information Card satisfying the Identity Law of Justifiable Parties (see Section 2.6.3). On top of this, the prototype integrates username and password authentication. This authentication will provide an opportunity to evaluate the Information Cards versus username and password. In this authentication, the user has to provide all their identity information in the registration process, so that the website contains all the user's identity information breaking the law of Minimal Disclosure.

Depending on the level of security required, the use of personal or managed cards is available. For transactions with a bank, the website may require the user to provide the identity information signed by an identity provider. In this thesis the IP based on the model of relationship-focused system, has been developed (see Section 2.8.1). If the user has to provide their identity information, the user selects their managed card in order to request this information signed in a Security Token by the IP. This prototype has been implemented in this model because the information is provided in a short-term identity federation token, which satisfies the property of Sharing Prevention (see Section 2.7.4). One of the disadvantages of this system is that the IP has to be online in every transaction restricting the privacy of the user, however at the same time this also reduces the workload for the user. The IP has to be executed in a different server and it will be awaiting a request from the user in order to provide the Security Token. The next chapter will now present the implementation of this design.

# 4 Implementation

## 4.1 Introduction

---

In this implementation section, the main parts of the developed applications will be presented. Not all of the developed code is presented in this chapter, due to its size. This section will demonstrate specific parts of the code that are the most pertinent parts of the Identity Management System. In the first part a high level concept of the solution is presented. The following section covers how the database is processed and stored in the system. After this part the creation of managed cards, and how to request a security token from the Identity Provider are discussed.

The last section of this chapter includes the construction of the SP. This is carried out in order to provide authentication of Information Card and username and password. The application also controls the performance of the server for both of the Identity Management systems. The system will store the time that the user needs to login and register in both authentication methods.

## 4.2 Implementation Graphic Schema

---

A graphic schema of the Identity Management System is displayed in Figure 4.1. The schema shows how the different parts of the Identity Management System are inter-linked by means of network connections.

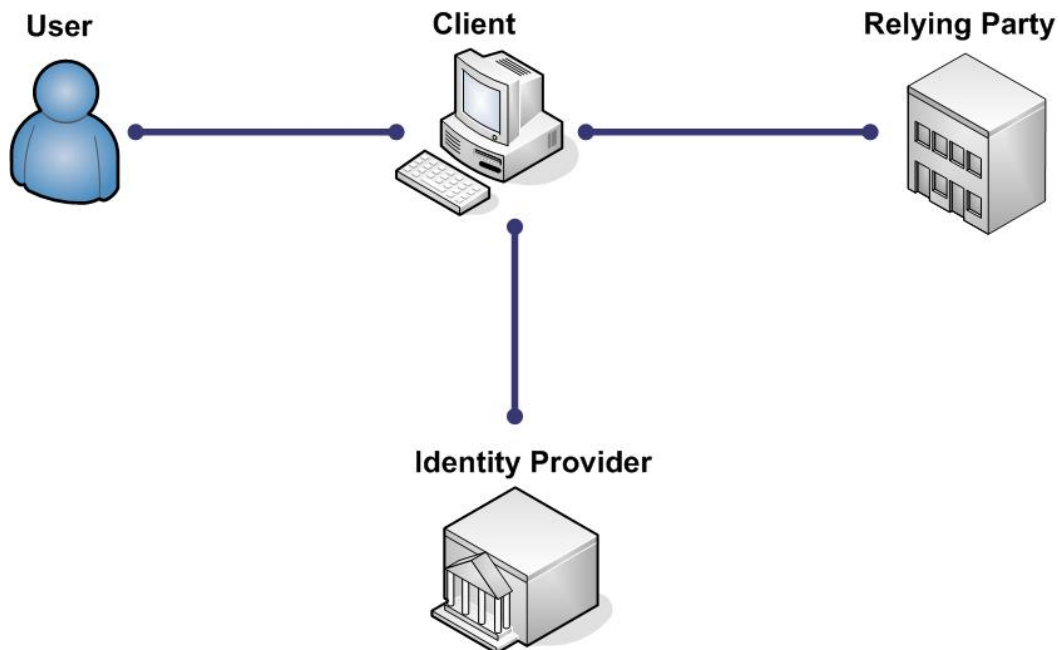


Figure 4-1: Implementation Graphic Schema

## 4.3 Database Implementation

In the design of the system a database containing 12 tables was presented. These tables are used for the storage and retrieval of data. Some of the data contains information about the creation of different roles for different users: user membership information, user account information and information cards related to user accounts.

In order to operate with the database, a number of stored procedures have been created. These stored procedures are displayed in figure 3.3. Some of the operations that these stored procedures achieve are:

- Creation of new users in the database.
- Associate the user with their information card.
- Retrieve information from the database, and so on.

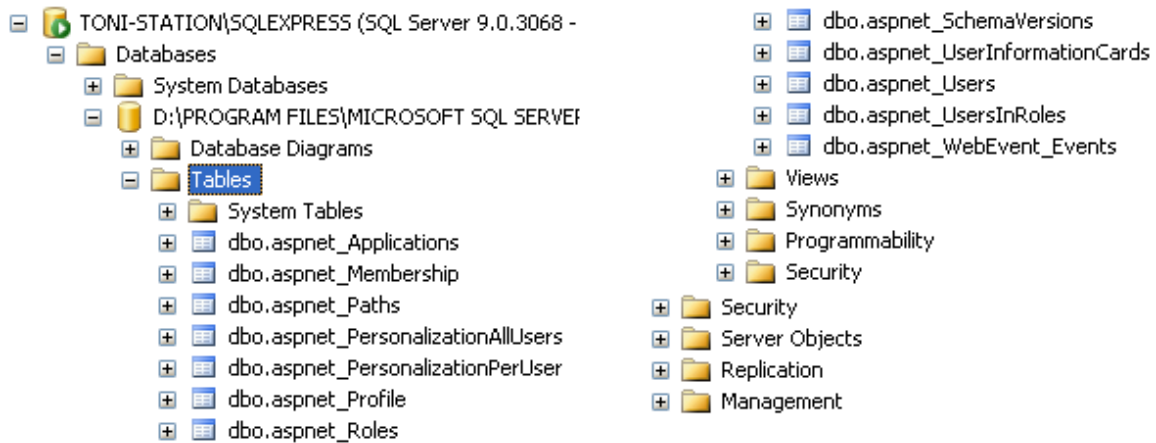


Figure 4-2: Database Implementation

### 4.3.1 SQL Query

The following script shows the creation process, which has been used to create the “UserInformationCards” table. This table contains the Private Personal Identifier, which is obtained from the user’s Information Card. The table also contains the “UserId” column, which is used in order to keep a relation with the other tables in the database.

The tables were created using Microsoft SQL Server Management Studio Express. This tool can be used free of charge for non-commercial use. One of the advantages of this tool is that the tables and the relations can be shown in a visual environment.

```
/* Author: Antonio Jose Fernandez Sepulveda */
/* Object: Table UserInformationCards */
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[aspnet_UserInformationCards](
    [UserID] [uniqueidentifier] NOT NULL,
    [PrivatePersonalIdentifier] [char](20) NOT NULL,
```

```
[Active] [bit] NOT NULL,  
[DateAdded] [datetime] NOT NULL,  
[DateModified] [datetime] NOT NULL  
) ON [PRIMARY]  
  
GO  
SET ANSI_PADDING OFF
```

**Table 4-1: Creating process of the UserInformationCards table**

### 4.3.2 SQL Stored Procedure

The following script shows one of the stored procedures created for the system. This stored procedure is going to retrieve the user's information by means of the Private Personal Identifier value. The Private Personal Identifier value can be linked to different user accounts which allows one user to link different accounts with the same information card. This stored procedure is used to retrieve the user's information when the user provides their information card from the Login page in the website.

```
/* Author: Antonio Jose Fernandez Sepulveda */  
/* Stored Procedure GetUserByInformationCard */  
  
ALTER PROCEDURE [dbo].[aspnet_Membership_GetUserByInformationCard]  
  
/* Input variable */  
@PPID char(20),  
@Email nvarchar(256),  
@CurrentTimeUtc datetime,  
@UpdateLastActivity bit = 0  
  
AS  
BEGIN  
    IF ( @UpdateLastActivity = 1 )  
        BEGIN  
            DECLARE @UserID uniqueidentifier;  
  
            // Return the UserID if the UserID is found in Users, UserInformationCards and Membership  
            // tables and the PPID is equal to PrivatePersonalIdentifier stored in the  
            // UserInformationCards table and the Email is equal to Email stored in the  
            // UserInformationCards table.  
            SET @UserID = ( Select u.UserID from dbo.aspnet_Users u,  
                            dbo.aspnet_UserInformationCards i,  
                            dbo.aspnet_Membership m  
                            where u.UserID = i.UserID  
                                AND m.UserID = u.UserID  
                                AND i.PrivatePersonalIdentifier = @PPID  
                                AND m.Email = @Email)  
  
            // Update the LastActivityDate field in the Users table  
            UPDATE dbo.aspnet_Users  
                SET LastActivityDate = @CurrentTimeUtc  
                FROM dbo.aspnet_Users  
                WHERE @UserId = UserID  
  
            // Returns the number of rows affected by the last statement  
            IF ( @@ROWCOUNT = 0 ) /* if the User ID is not found */  
                RETURN -1  
  
        END  
END
```

```
/* Return Email, PasswordQuestion, Comment... fields if the UserID is found in the Users,  
/* UserInformationCards and Membership tables and the PPID is equal to PrivatePersonalIdentifier  
/* stored in the UserInformationCards table and the Email is equal to Email stored in the  
/* UserInformationCards table.  
SELECT m.Email, m.PasswordQuestion, m.Comment, m.IsApproved,  
        m.CreateDate, m.LastLoginDate, u.LastActivityDate, m.LastPasswordChangedDate,  
        u.UserName, m.IsLockedOut, m.LastLockoutDate  
FROM dbo.aspnet_Users u, dbo.aspnet_Membership m, dbo.aspnet_UserInformationCards i  
WHERE u.UserID = i.UserID  
        AND m.UserID = u.UserID  
        AND i.PrivatePersonalIdentifier = @PPID
```



```
        AND m.Email = @Email

// Return -1 if the user is not found
IF ( @@ROWCOUNT = 0 ) -- User ID not found
    RETURN -1

RETURN 0
END
```

**Table 4-2: Stored Procedure Get User By InformationCard**

## 4.4 Managed Card Implementation

The following script shows the code used to implement the managed card. The managed card contains the following information:

- Information on the entity providing the card as a security certificate, the Security Token Service STS and the Metadata Exchange MEX endpoints.
- The picture that will be shown in the managed card when it is selected from the identity selector.
- The list of claims belonging to the user. The Identity Provider affirms that these claims are true.
- Information about how the user provides their credentials to the Identity Provider in order to obtain the list of claims from the Security Token Service.

```
[CARD]
// Define the type of authentication that is used to connect with the Security Token Service.
// We can use four different types of authentication UserName and Password, Keberos,
// SelfIssued or Personal cards and Smart Card.
TYPE=SelfIssuedAuth

[Details]
// The Name value stores the name of the card that is going to be showed in the identity
// selector.
Name=My Managed Card
// The ID value stores the identifier of the card. If we try to import two cards with the same
// ID, the identity selector will ask if we want to replace the old card with the new one.
ID=http://takingyouhome.no-ip.org/card/self/randomnumber123
// The Version value stores the version number for the card. If we import a card with the same
// ID, the version number should be incremented.
version=1
// The Image value stores the path for the picture that we are going to insert into the card.
// The picture is going to be stored in the card, so this is going to break any link with the
// user.
image=images\managedcard.jpg

[Issuer]
Name=Identity Provider
// The Address value stores the endpoint URL of the Identity Provider STS.
Address=http://takingyouhome.no-ip.org:7000/sample/trust/selfissuedsaml/sts
// The MexAddress value stores the endpoint URL for the Security Token Service. This endpoint
// requires using Secure Socket Layer protocol (SSL).
MexAddress=https://takingyouhome.org:7001/sample/trust/selfissuedsaml/mex
// The Privacy Policy stores the address where the Identity Provider's policy is found.
PrivacyPolicy=http://takingyouhome.no-ip.org/PrivacyPolicy.xml
// The Certificate stores the path where the Identity Provider's certificate is stored. The
// public key of this certificate is going to be used to sign the card. If we provide the
// cert.pfx path, we have to provide the certificate Password as well
Certificate=LOCALMACHINE/MY/takingyouhome.no-ip.org
// CertificatePassword=password

[Claims]
// Identity Provider supports. This section defines the claims that will be The claims section
// stores the Uniform Resource Identifiers URI that the Identity Provider supports. This section
// defines the claims that will be provided in the Security Token.
l=http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname
```

```
2=http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname
3=http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress
4=http://schemas.xmlsoap.org/ws/2005/05/identity/claims/streetaddress
5=http://schemas.xmlsoap.org/ws/2005/05/identity/claims/locality
6=http://schemas.xmlsoap.org/ws/2005/05/identity/claims/privatepersonalidentifier
7=http://custom-claim-uri.com/car

// The IP can provide custom claims, but we have to provide the name and description for every
// created custom claim.
[http:// custom-claim-uri.com/car]
display=Car Model
description=Provide a custom claim

[TokenTypes]
// The Token Type section specify the type of tokens that the Security Token Service support
1=urn:oasis:names:tc:SAML:1.0:assertion
;2=http://docs.oasis-open.org/wss/oasis-wss-saml-token-profile-1.1#SAMLV1.1

[Token Details]
// The RequiresAppliesTo value specifies if the Identity Provider will issue the security token
// to any Relying Party or only to the Relying Parties that provide who they are.
RequiresAppliesTo=false

[Credentials]
// The Credentials section specifies the authentication that is provided to Identity Provider.
// If we use Self-Issue card authentication, we have to provide the obtained Private Personal
// Identifier PPID using the Identity Provider's certificate.
value=il/xn0eyq4taLLFOZCkefqZPCEIE8vZZlArmfc3V930=
// The Hint stores any credential hint information
Hint=
```

Table 4-3: Managed Card Information

## 4.5 Request Security Token Implementation

The following script demonstrates how the Security Token Service (STS) obtains the contained information into an incoming request, when the user selects the managed card to obtain the security token. The STS obtains the key information that will be used to create the security token. This information is stored in properties of the class.

```
// The wst:Claims element contain a list of ClaimType elements. Each element contain the URI of
// the claim that was request for the information card. Every URI is stored in a list of
// RequestedClaims. This list is used to populate the security token with the requested claims.
private void GetRequestedClaims(XmlReader reader)
{
    while( reader.Read()
        && !( reader.NodeType == mlNodeType.EndElement
            && reader.Name == "wst:Claims"))
    {
        if (reader.IsStartElement())
        {
            if (reader.HasAttributes)
            {
                reader.MoveToFirstAttribute();
                RequestedClaims.Add(reader.Value);
            }
        }
    }
}

// The RequestDisplayToken element define if the identity selector can show
// to the user the information that is going to be sent to the Relying party
// and the RequestDisplayTokenLanguage variable stores the language in which
// this information is going to be showed.
private void GetRequestDisplayToken(XmlReader reader)
{
    RequestDisplayToken = true;
    if (reader.HasAttributes)
    {
        reader.MoveToFirstAttribute();
    }
}
```

```
        RequestDisplayTokenLanguage = reader.Value;
    }
    reader.Read();
}

// The wsid:InformationCardReference element contain two child elements
// CardId and CardVersion. These values are stored in the CardId and
// CardVersion variables in the InformationCardReference class.
private void GetInformationCardReference(XmlReader reader)
{
    while (reader.Read()
        && !(reader.NodeType == XmlNodeType.EndElement
            && reader.Name == "wsid:InformationCardReference"))
    {
        if (reader.IsStartElement())
        {
            string elementName = reader.Name;
            elementName = elementName.Substring(5);
            reader.Read();
            string elementValue = reader.Value;
            if (elementName == "CardId")
            {
                informationCardReference.CardID = elementValue;
            }
            if (elementName == "CardVersion")
            {
                informationCardReference.CardVersion = elementValue;
            }
        }
    }
}

// the ClientPseudonym element can be contained in the information card
// request. This is used to provide the user's pseudonym PPID. It is
// retrieved from a child element of PPID and stored in the ClientPseudonym
// variable.
private void GetClientPPID(XmlReader reader)
{
    reader.Read();
    reader.Read();
    ClientPseudonym = reader.Value;
    reader.Read();
}
```

Table 4-4: Security Token Service

## 4.6 Website Implementation

The following diagram displays the website that has been created to implement the Relying Party. This website provides support for using information card or username and password credentials.

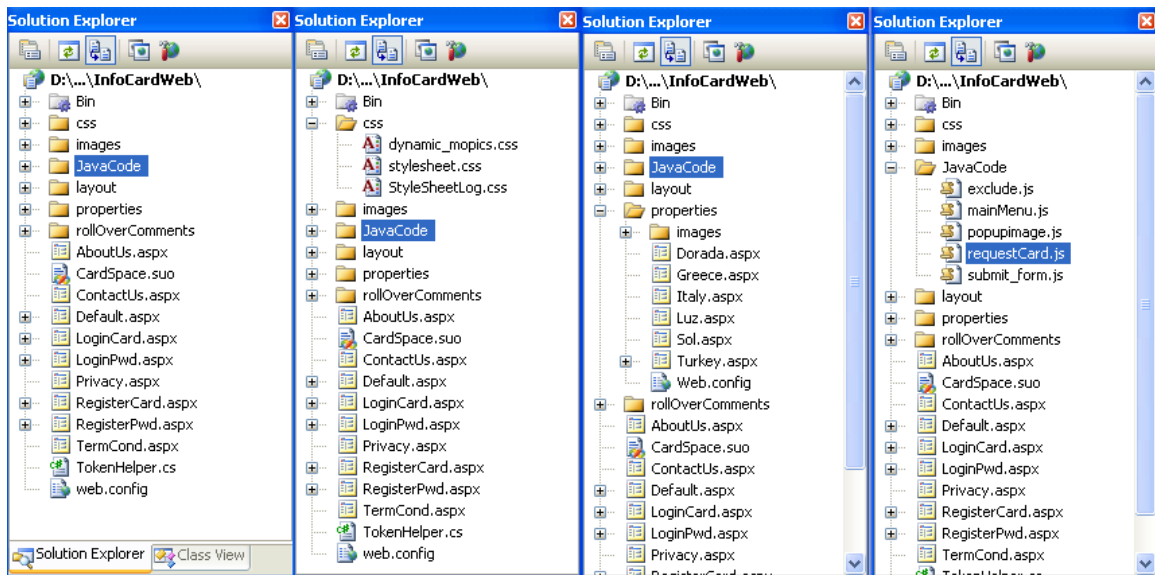


Figure 4-3: Website Solution

The solution has been distributed into the following parts:

- The root directory contains the main pages of the website, such as LoginCard, LoginPwd, Privacy, and so on.
- The CSS folder contains the Cascading Style Sheets (CSS) that have been used to describe the design of the different pages in the website.
- The images folder contains the used pictures to show the fictitious products in the website.
- The JavaCode folder contains the used Java code to trigger the identity selector and to create some functionality for the website.
- The Layout folder contains the images that have been used to create the graphic interface in the website.
- The Properties folder contains the pages that are restricted to authenticated users in the website. These pages have security restrictions which are different to the rest of the website.

## 4.7 Request Identity Selector Implementation

The following script shows how the Identity Selector is triggered when the user attempts to login to the website. The code is separated in a Java Script file and it is linked to the files required to use it.

```
// This function is executed when the user presses the button to Login into the website
function requestInformationCard()
{
    // This code is going to create the object tag dynamically and to trigger the Identity
    // Selector. This object tag contains the web site policy that is going to tell to the
    // Identity Selector which Information Cards could be used to login into the web site.
    // The required Claims parameter indicates the information that the user has to provide
    // with the information card.
    var icObject = '<object type= "application/x-informationcard" name="xmlLoginToken">';
    icObject += '<param name= "tokenType" value="urn:oasis:names:tc:SAML:1.0:assertion" />';
    icObject += '<param name= "issuer"
```

```
        value="http://schemas.xmlsoap.org/ws/identity/issuer/self"/>";
icObject += '<param name= "requiredClaims"
icObject += value= "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname ' ;
icObject += 'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname ' ;
icObject +=
    'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/privatepersonalidentifier ' ;
icObject += 'http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress" />';
icObject += '</object>';

// Assign the object XML to the "InformationCardToLoginObjectContainer" element. This
// element is a div tag that defines a division in page to insert the XML that is
// created dynamically. The innerHTML variable inserts the object into the page and
// getElementById property assigns this object to
// "InformationCardToLoginObjectContainer" div.
document.getElementById("InformationCardToLoginObjectContainer").innerHTML=icObject;

// The Submits function is going to submits the form that is going to trigger the
// identity selector
document.getElementById("form1").submit();
}
```

**Table 4-5: Request Identity Selector**

## 4.8 Login Implementation

The following script illustrates how the user is logged into the website when an Information Card is provided from the Identity Selector.

```
// This function receives the Security Token and then it is going to
// retrieve the user's information and to check
// if the information card is associated to one user account
bool AuthenticateInformationCardUser(string xmlToken)
{
    // This method is going to retrieve the user's information from the xmlToken
    RetrieveTokenClaims(xmlToken);

    // Create a connection string to our SQL Server database and the we are going to open
    // the connection
    string connString= ConfigurationManager.ConnectionStrings[
        "LocalSqlServer"].ConnectionString;
    SqlConnection MyConnection = new SqlConnection(connString);
    MyConnection.Open();

    // We are going to create a instance of GetUserByInformationCard stored procedure to
    // execute in the SQL Server database
    SqlCommand MyCommand = new SqlCommand("GetUserByInformationCard", MyConnection);
    MyCommand.CommandType = CommandType.StoredProcedure;

    // We are going to create the email property for the stored procedure.
    MyCommand.Parameters.Add(new SqlParameter("@Email", SqlDbType.NVarChar, 256));
    MyCommand.Parameters["@Email"].Value = _email;
    MyCommand.Parameters.Add(new SqlParameter("@PPID", SqlDbType.Char, 20));

    // We assign the PPID value that we have obtained from the Security Token to the PPID
    // property for the stored procedure
    MyCommand.Parameters["@PPID"].Value = _ppid;

    // We are going to create the CurrentTimeUtc property for the stored procedure
    MyCommand.Parameters.Add(new SqlParameter("@CurrentTimeUtc", SqlDbType.DateTime));

    // We assign the current time value to the CurrentTimeUtc property for the stored
    // procedure
    MyCommand.Parameters["@CurrentTimeUtc"].Value = DateTime.Now;

    // We are going to create the UpdateLastActivity property for the stored procedure
    MyCommand.Parameters.Add(new SqlParameter("@UpdateLastActivity", SqlDbType.Bit));

    // This property is used to set the last time that the database was queried
    MyCommand.Parameters["@UpdateLastActivity"].Value = 1;

    // Execute stored procedure
```

```
SqlDataReader reader = MyCommand.ExecuteReader();

// Return true if there is a user associated with this information card
return reader.HasRows;
//Close the connection.
MyConnection.Close();
}
```

**Table 4-6: Login with Information Card Process**

## 4.9 Timing Test Query

In this section, we are going to check the performance for the Login and Registration procedures. Since the process of login and registration is carried out in the user's local machine, a number of calls to the server have to be created, in order to record the total time taken. Controlling the time is not an easy task, since every time a user is connecting to the server, the time has to be reset in the class where this variable is created. In order to solve this problem, session variables are being used which will not be modified during the time the user is interacting with a procedure. The following script shows a specific part of the code that uses this variable:

```
/// <summary>
/// This method is called every time the the page is loaded
/// </summary>
protected void Page_Load(object sender, EventArgs e)
{
    // obtain the ArrayList that has been created in the session
    ArrayList ListWatchTimer = (ArrayList)Session["Timer"];

    // if the WatchTimer instance has not been created
    if (ListWatchTimer.Count == 0)
    {
        // create the WatchTimer instance and assign page name
        WatchTimer newWatchTimer = new WatchTimer();
        newWatchTimer.Identifier = "Login Information Card";
        // add to the ArrayList a new WatchTimer class
        ListWatchTimer.Add(newWatchTimer);
    }
    else
    {
        // search if there is instance from other page
        foreach (WatchTimer VariableTimer in ListWatchTimer)
        {
            // if the identifier is different to the current page
            if (VariableTimer.Identifier != "Login Information Card")
            {
                // clear the ArrayList in the session
                ListWatchTimer.Clear();
                // create the WatchTimer instance and assign page name
                WatchTimer newWatchTimer = new WatchTimer();
                newWatchTimer.Identifier = "Login Information Card";
                // add to the ArrayList a new WatchTimer class
                ListWatchTimer.Add(newWatchTimer);
                break;
            }
        }
    }
}
```

**Table 4-7: Timing Control**

Every time a user tries to login to or register with a website, the WatchTimer class is created and the timer is initialized. When the user has completed the process, the timer is stopped and the event is stored by means of a stored procedure into the database in a table that has been created to register this process. Table 4.4 shows a number of opera-

tions that have been registered in the database. The table contains information about the user calling the process, the operation type, the time that the operation took and the date in which the operation was made and the time that the server takes to respond to the service.

As seen from the table below, in most cases the time it takes to login to and registers in the website is greater when the user utilizes username and password authentication however, the time of the response from the server is faster when the user employs username and password authentication. When analysing and comparing the total time that the user spends when providing their credentials, it can be deduced that Information Card provides a better service. These results can change depending on a number of factors such as server response, connection speed, SQL Server performance, and so on.

TONI-STATION\.....aspnet_Timer					
	Username	OperationType	OperationTime(Ms)	Date	ServerResponseTime(Ms)
	newcard	Login Information Card	27490	11/11/2008 21:18:35	6388
	obama	Login Information Card	4372	11/11/2008 21:18:43	207
	Elina	Login Information Card	7548	11/11/2008 21:18:53	40
	Antonio	Register Username/Password	167581	11/11/2008 21:59:41	83790
	Antonio	Login Username/Password	20679	11/11/2008 22:00:13	20
	Elina	Login Information Card	4309	11/11/2008 22:00:40	62
	Camila	Register Information Card	9418	11/11/2008 21:19:25	3807
	newcard1	Register Username/Password	110203	11/11/2008 21:21:25	55101
	newcard1	Login Username/Password	19248	11/11/2008 21:21:46	20
	antonio	Login Information Card	4675	11/11/2008 22:00:23	44
	Elina	Register Information Card	5306	11/11/2008 22:00:32	136

**Figure 4-4: Database Timer**

## 4.10 Conclusion

In this chapter, the most important parts of the development of the system have been covered by providing key parts of the code for the implementation and providing comments about the functions of this code within the application. In the first section a general schema is shown of how the system is linked to the different parties in the Federated Identity Management system. This schema satisfies the Relationship-focused Model showed in the Section 2.8.1. In this model, the IP has to be online if the SP requires that the Security Token will be signed by it. In this model, the SP can accept both personal cards as well as managed cards. If the user uses a managed card, the IP will have to be online to obtain the user's credentials. In Section 4.3, the creation and manipulation of the database is demonstrated. The database has been manipulated by means of stored procedures. Some advantages of the use of stored procedures are that these reduce the execution time and allow executing complex set of SQL statements. In the Section 4.4 is explained how to built the managed card in order to communicate with the Identity Provider and obtain the security token. In the Section 4.5 is explained how the Security Token Service (STS) has been implemented in order to respond to the user, when the user sends their managed card to obtain the security token. In the Section 4.6 an overview of how the website has been designed and distributed in order to create

the Service Provider. In Section 4.7, it is shown how the system triggers the Identity Selector in order that the user provides their Information Card. In the Section 4.8 it has been shown how the information is retrieved from the Information Card and stored in the database, when the user logs into the website. In the last section, additional functionality that checks the server performance when the user provides their authentication by means of Information Cards or username and password, has been presented. The system logs the time both for the login process and the registration process. We can therefore conclude that the system meets the functional requirements as presented in the design chapter.



# 5 Evaluation

## 5.1 Introduction

---

The aim of this evaluation is to provide a clear conclusion of the technologies introduced in this thesis. The evaluation process will cover testing of the most important aspects of the system. Since the technology provides a possible alternative to the use of username and password authentication, the testing process will cover both technologies. In this chapter, an evaluation will be carried out of the necessary requirements in order to use Information Cards and the current support for this technology. An evaluation of the current Identity Management systems will also be performed. In the last part of this chapter, an assessment of the user experience with Information Cards will be carried out, to see if this technology could be a possible solution to the security problems of the traditional Identity Management approach.

## 5.2 Evaluation Test

---

In this section, the functionality provided in the website will be tested. This test will cover the performance of the use of Information Cars authentication and of the use of username and password authentication. This test will also cover the use of personal and managed cards and how the information can be retrieved from the Identity Provider. In the last part of this section, an assessment of the website, when the user requests a service will be carried out.

### 5.2.1 Login to the website using an Information Card

The key of this test is to login to the website using an Information Card supplied by the user. This process is described below:

Steps for the Login process		
1.	Open the browser and go to the address <a href="https://takingyouhome.no-ip.org/">https://takingyouhome.no-ip.org/</a>	
2.	Go to the “Infocard Login” section and press the “Login” button	
3.	The “LoginCard” page will then load	
4.	Go to the “Sign in with your infocard” section and press the purple button next to the “Select one card” text	
5.	The CardSpace selector is then triggered and the cards that find the policy requirements of the website are showed in colour	
6.	Select one of these cards and press the “Send” button	
7.	Then the card is sent to the server and if the user is registered in the website, the user is authenticated and returned to the “Default” page	
<b>Process Result</b>		Passed <input checked="" type="checkbox"/> Failed <input type="checkbox"/>

Table 5-1: Login process with Information Cards

### 5.2.2 Register into the website using an Information Card

The key of this test is to register into the website using an Information Card supplied by the user. This process is described below:

Steps for the Register process		
1.	Open the browser and go to the address <a href="https://takingyouhome.no-ip.org/">https://takingyouhome.no-ip.org/</a>	
2.	Go to the “Inforcard Login” section and press the “Register” button	
3.	The “RegisterCard” page will then load	
4.	Go to the “Register using your Information Card” section and press the purple button next to “Select one card” text	
5.	The CardSpace selector is then triggered and the cards that find the policy requirements of the website are showed in colour	
6.	If there is not a Information Card to select, press “Add a Card” link and then select “Create a personal card” link	
7.	Complete all the red fields to success the website policy and then press the “Save” button	
8.	Select the created card and press the “Send” button	
9.	The card is sent to the server and the user is registered in the website	
10.	The user is authenticated and returned to the “Default” page	
<b>Process Result</b>		Passed <input checked="" type="checkbox"/> Failed <input type="checkbox"/>

Table 5-2: Register process with Information Cards

### 5.2.3 Login to the website using an username and password

The key of this test is to login to the website using a username and password supplied by the user. This process is described below:

Steps for the Login process		
1.	Open the browser and go to the address <a href="https://takingyouhome.no-ip.org/">https://takingyouhome.no-ip.org/</a>	
2.	Go to the “Password Login” section and press the “Log In” button	
3.	The “LoginPwd” page will then load	
4.	Go to “Sign in with your user and password” section and complete the “User Name” and “Password” fields and then press the “Sign In” button	
5.	The information is then sent to the server and if the user is registered in the website, the user is authenticated and returned to the “Default” page	
<b>Process Result</b>		Passed <input checked="" type="checkbox"/> Failed <input type="checkbox"/>

Table 5-3: Login process with username and password

### 5.2.4 Register into the website using an username and password

The key of this test is to register into the website filling in a form with the user’s personal information. This process is described below:

Steps for the Register process		
1.	Open the browser and go to the address <a href="https://takingyouhome.no-ip.org/">https://takingyouhome.no-ip.org/</a>	

2.	Go to the “Password Login” section and press the “Register” button
3.	The “RegisterPwd” page will then load
4.	Complete the form with the required information and press the “Create User” button
5.	A second page will then load. An Information Card can be associated with the user account. This is optional to the user, so next can be pressed in the case that the user do not want to associate an Information Card
6.	Press the purple button next to “Select Card” text
7.	The following steps have been explained in the section 5.1.2 (Steps from 5 to 10)
8.	Press the “Next” button to send the information to the server and register the user into the website
9.	The user is authenticated and returned to the “Default” page
<b>Process Result</b>	
Passed <input checked="" type="checkbox"/> Failed <input type="checkbox"/>	

**Table 5-4: Register process with username and password**

### 5.2.5 Login to the website using an managed card

The key of this test is to login to the website using an Information Card supplied by the Identity Provider. The user needs to have the managed card provided by the Identity Provider, and the Security Token Service has to be online to provide the security token with the user’s information. This card is contained in the Identity Selector together with the personal cards. This process is similar to the use of personal card and it is described below:

<b>Steps for the Login process</b>	
1.	Open the browser and go to the address <a href="https://takingyouhome.no-ip.org/">https://takingyouhome.no-ip.org/</a>
2.	Go to the “Inforcard Login” section and press the “Log In” button
3.	The “LoginCard” will then load.
4.	Go to “Sign in with your infocard” section and press the purple button next to “Select one card” text
5.	Then CardSpace selector is triggered and the cards that find the policy requirements of the website are showed with colour
6.	Select the managed card
7.	Press the “Send” button which is going to request to the Identity Selector the security token
8.	The security token is then retrieved from the Identity Provider and it is sent to the server
9.	If the user is registered in the website, the user is then authenticated and returned to the “Default” page
<b>Process Result</b>	
Passed <input checked="" type="checkbox"/> Failed <input type="checkbox"/>	

**Table 5-5: Login process with managed cards**

One of the differences with the use of personal card is that when the user selects the managed card, the user’s information is not displayed, as with a personal card. It is be-

cause the information is never in the user’s computer, as the information is retrieved only in the security token and it is then sent to the website.

### 5.2.6 Register into the website using an managed card

The key of this test is to register into the website using an Information Card supplied by the Identity Provider. In the same way as the previous point, the user needs to have the managed card provided by the Identity Provider and the Security Token Service has to be online to provide the security token with the user’s information. The card has to contain all the information required by the website’s policy. This process is similar to the use of personal card and it is described below:

Steps for the Register process		
1.	Open the browser and go to the address <a href="https://takingyouhome.no-ip.org/">https://takingyouhome.no-ip.org/</a>	
2.	Go to the “Inforcard Login” section and press the “Register” button	
3.	The “RegisterCard” page will then load	
4.	Go to “Register using your Information Card” section and press the purple button next to “Select one card” text	
5.	The CardSpace selector is then triggered and the cards that find the policy requirements of the website are showed in colour	
6.	Select the managed card	
7.	Press the “Send” button, which is going to request the Identity Selector the security token with the information required by the website	
8.	The security token is then retrieved from the Identity Provider	
9.	The security token is then sent to the server and the user is registered in the website	
10.	The user is authenticated and returned to the “Default” page	
<b>Process Result</b>		Passed <input checked="" type="checkbox"/> Failed <input type="checkbox"/>

Table 5-6: Register process with managed cards

### 5.2.7 Request a service from the website

The key of this test is to check how the user can obtain a service when he/she is not logged on to the website. This process is described below:

Steps for the Login process	
1.	Open the browser and go to the address <a href="https://takingyouhome.no-ip.org/">https://takingyouhome.no-ip.org/</a>
2.	Go to the “Categories” section and select the link with the name of the place that obtains the information, for example the Turkey link
3.	The “LoginCard” page will load
4.	Go to the “Sign in with your infocard” section and press the purple button next to “Select one card” text
5.	The CardSpace selector is then triggered and the cards that find the policy requirements of the website are showed in colour
6.	Select one of these cards and press the “Send” button

7.	The card is sent to the server and if the user is registered in the website, the user is authenticated		
8.	Then the requested page is returned		
<b>Process Result</b>		Passed <input checked="" type="checkbox"/>	Failed <input type="checkbox"/>

**Table 5-7: Request service from website**

## 5.3 Quantitative Evaluation

This section will provide a quantitative evaluation of Information Cards over the more traditional username/password approach to authentication.

### 5.3.1 Information Card Requirements

One disadvantage of information cards with respect to username and password authentication is that this technology needs some installations on the client side. In order to use windows Cardspace Selector, the client has to install .NET Framework 3.0.

When the user has installed .NET Framework 3.0, the user is able to see the Cardspace icon in the Control Panel. This will allow the user to create and to modify information cards.

### 5.3.2 Comparative Test

Since Information Cards is a recent technology, not all of the current browsers provide support for this technology. An evaluation of the most important current browsers providing support for Information Cards has been made. The following table shows the result of this evaluation:

Web Browser	Operating System	Built In	Need Ext Plug-in
Internet Explorer	Windows XP / Vista	Yes	No
Firefox	Windows XP / Vista	No	Yes
Opera	Windows and Mobiles	No	Yes
Chrome	Windows	No	No
Safari	Mac and Mobile	No	Yes

**Table 5-8: Comparative table of the most important Browsers**

In the table we can see that the majority of the most important browsers include support for Information Cards. Internet Explorer is the only one that includes CardSpace support built in. Firefox, Opera and Safari browsers provide plugins which can be installed to support Information Cards. A disadvantage is that, since the technology is quite new, the provided plugins are not easy to install for a normal user. Google's new Chrome browser does not provide support for Information Cards at the moment.

## 5.4 Qualitative Evaluation

In this section, a number of different questionnaires will be created in order to evaluate the user experience of using an Identity Management System. This evaluation has been carried out by thirteen different users from different countries such as UK, Sweden and Spain. The results from this evaluation are presented as an overall percentage at the end of each evaluation. The users that carried out the evaluation came from a range of disciplines, approximately half of the subjects came from a technical background. The evaluation is thus using a biased sample, however due to the technical nature of the subject it was necessary to do this. In the cases that the subjects were non-technical, each of the different Identity Management systems were explained to them in advance.

### 5.4.1 Identity Management Systems Evaluation

The following table displays an evaluation of the current systems of authentication. The scale of bad to excellent allows the user to rate the various systems in order to evaluate which Identity Management System is the best in terms of security and usability. These authentication systems have previously been explained in the authentication section of the literature review.

	<b>Bad</b>	<b>Average</b>	<b>Good</b>	<b>Excellent</b>
<b>Username and password</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>CardSpace</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<b>Digital Certificate</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Face Recognition</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Fingerprint</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Hand Reader</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Iris Scanning</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>DNA Identification</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Keystroke Typing</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Signature Detection</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Voice Verification</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**Table 5-9: Comparative table of Identity Management Systems**

Username and password is a simple authentication method that is used to gain access to a service for example in a website. The user has to remember their username and password in order to access the service. The results of this authentication method reveals the majority of people's opinions on this method:

- Easy to use, people know how to use it.

- Difficult to remember the username and password for all the different services, which makes most of the people use the same authentication for different services.

CardSpace is an authentication system based on the use of information cards to provide the user's credentials. CardSpace provides a client interface called Identity Selector, where the user can make a number of operations such as creation, storage, sending, and so on. When a user wants to obtain a service, he/she can select the Information Card that contains the claims requested from the service. The results of this authentication method reveal the majority of people's opinions on this method:

- Problematic to install CardSpace, this does not work in all the different Browsers.
- The technology is very convenient, the user only needs to remember which Information Card to use. This Information Card is linked to one picture, so the user can remember this very easily.
- Very fast in providing the user's credential.

Digital Certificates is an authentication method based on the use of public and private key cryptography and SSL. The digital certificate is issued by independent, recognized and trusted third party CA, guaranteeing the service to be what it claims to be. The digital certificate binds this service with its public key. The digital certificate contains the service's name, the digital signature, the service's public key, expiration date, and so on. When digital certificates are in order, the browser establishes the secure connections. The results of this authentication method reveal the majority of people's opinions on this method:

- Most of the people are not familiar with the technology.
- There is not enough support for the service.
- Obtaining a certificate is considered expensive.

Face recognition is an authentication method based on the recognition of the user's face by means of a digital image. The system compares the image obtained as input with an image of the user recorded in the database. The results of this authentication method reveal the majority of people's opinions on this method:

- The technology is very convenient, since the user does not need to remember anything specific such as a password, but it involves the introduction of specific hardware.
- They do not like the technology, because it means a possible invasion of their privacy
- Difficult to introduce for every public service and user.

Fingerprint is an authentication method based on the impression of the bottom of the user's finger. The authentication system scans the fingerprint of the user and compares

this with the user's fingerprint stored in the database. The results of this authentication method reveal the majority of people's opinions on this method:

- The technology is very convenient, since the user does not need to remember anything such as a password, but it involves the introduction of a specific hardware.
- They do not like the technology, because it reveals important information about the person.
- Difficult to introduce for every public service and user.

Hand Reader is an authentication method based on the geometry of the user's hand. The authentication system scans the unique geometry of the user's hand and compares this with the template stored in the database. The obtained results for this authentication method are the same as those obtained for Fingerprint.

Iris Scanning is an authentication method based on a high-resolution image of the user's iris. The authentication system uses a camera with infrared illumination to obtain an image of the user's iris, this image is converted to a digital template and is compared with the previous template of the user stored in the database. The results of this authentication method reveal the majority of people's opinions on iris scanning and hand reader technology:

- Involves the introduction of a specific hardware and some users do not like to expose themselves to this type of authentication.
- They do not like the technology, because this reveals important and private information about the person.
- Difficult to introduce for every public service and user.

DNA Identification is an authentication method based on the structure present in every human cell. The authentication system obtains a sample of the user's DNA (the sample can be obtained from blood, saliva, hair, semen or tissue) and compares this to the previous template of the user stored in the database. The results of this authentication method reveal the majority of people's opinions on this method:

- They do not like at all to provide blood, semen, and so on because this reveals really important information about the person.

Signature Detection is an authentication method based on the user's handwritten signatures. The authentication system scans the user's signature and compares this to the image signature stored in the database. Advanced Signature Detection systems can check rhythm, acceleration and pressure of the user's writing. The results of this authentication method reveal the majority of people's opinions on this method:

- The technology involves the introduction of a specific hardware.



- Most of the people find it difficult to reproduce their signature in the same way.

Voice Recognition is an authentication method based on the user's voice tone. The authentication system compares the user's voice to the pattern stored in the database. The results of this authentication method reveal the majority of people's opinions on this method:

- The technology is very convenient, since the user does not need to remember any specific information, such as a password.
- Some of the people do not like to record their voices, because this can reveal important and personal information about the person.

Keystroke Typing is an authentication method based on measuring the time a key of the keyboard is held down and the duration between taps when the user writes their authentication. When the user tries to write their authentication, the system measures their results with the pattern stored in the database. The results of this authentication method reveal the majority of people's opinions on this method:

- The technology is very convenient, since the user only needs to write some text to obtain access to the service, but it can involve the introduction of a specific hardware.
- Some people do not like the technology since they might not always write in the same way, this can therefore create a problem when introducing their credentials.

From the results obtained from the identity management systems evaluation, a chart has been created as shown in Figure 5.1 This chart is demonstrating the subjects' opinions with regards to the different authentication technologies. The CardSpace and Username/Password authentications were found to be the most popular. The reasons behind these answers are explained in more detail following the explanation of the technologies, as seen above. The comments from the subjects are a summary of their opinions of the different Identity Management systems.

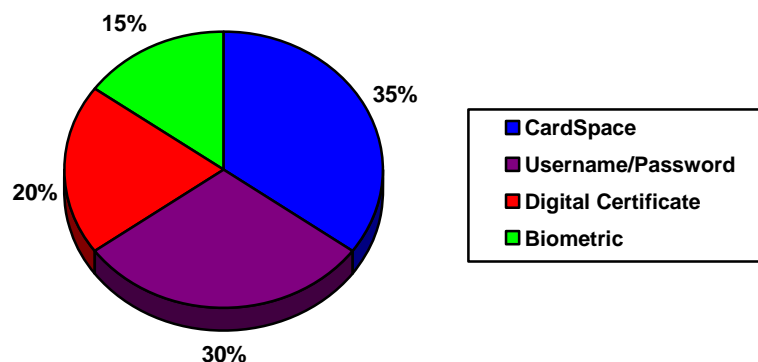


Figure 5-1: Results from the Identity Management Systems Evaluation

### 5.4.2 Information Card Technology Evaluation

In the web site created for this application, two different methods have been created for authentication, Information Card identification and username and password identification. The following tables show an evaluation of Information Card authentication versus username and password authentication.

The following table of questions represents a degree of evaluation that has a range from bad to an excellent level of acceptance by users. Some of the questions are of a general nature and others require a minimum level of knowledge in some of the issues evaluated in this thesis.

	<b>Bad</b>	<b>Average</b>	<b>Good</b>	<b>Excellent</b>	<b>(%)</b>
<b>Would you consider it easy to find the pages for info card and password login?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>90%</b>
<b>Do you find it easy to register on the website with Information Cards?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>75%</b>
<b>Do you find it easy to login and logout with Information Cards?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>85%</b>
<b>Do you consider login with Information card to be a fast authentication?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>75%</b>
<b>Do you consider it good to encrypt your identity information with a Security Certificate?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>55%</b>
<b>Do you find it easy to use Windows CardSpace?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>75%</b>
<b>Do you find it secure to use an Identity Provider to handle your identity information?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>65%</b>
<b>Do you find it easy to create an information card?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>70%</b>

**Table 5-10: Acceptance Level Information Cards Authentication**

The following table of questions represents an evaluation of Information Card authentication versus username and password authentication. Just like the previous table, some

of the questions are of general nature and others require a minimum level of knowledge in some of the issues evaluated in this thesis.

	<b>Yes</b>	<b>No</b>	<b>(%)</b>
<b>Do you consider Information Card easier to handle than Username and Password?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>75%</b>
<b>Do you consider Information Card to be a faster method than Username and Password?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>80%</b>
<b>Do you find it easy to login and logout with Information Cards?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>75%</b>
<b>Do you consider it good to encrypt your identity information with a Security Certificate?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>85%</b>
<b>Do you find CardSpace easy to use in different browsers?</b>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<b>90%</b>
<b>Do you find it easy to use Windows CardSpace?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>65%</b>
<b>Do you consider Information Card technology to be a more secure method than username and password?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>90%</b>
<b>Do you consider encrypting the identity information with a Security Certificate to be a secure communication?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>80%</b>
<b>Do you find it useful to have different cards for different services?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>90%</b>
<b>Do you think it is better to use one card for different services than to remember different usernames and passwords?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>95%</b>
<b>Do you find it easy to set up Information Cards in your computer?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>65%</b>
<b>Do you think it is easier to use an information card than to remember a username and password to login to a service?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>85%</b>
<b>Do you consider Security Token to be a secure method to exchange information?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>75%</b>
<b>Do you think that Information Card would be a good replacement for username and password authentication?</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<b>80%</b>

**Table 5-11: Information Cards versus user names and passwords**

From the results obtained from this evaluation, the following comments were highlighted by a large number of users:

- It is very convenient to Login to website with Information Cards
- Finding the Login section on the website is straightforward
- Login with Information Cards is faster than with usernames and passwords
- To encrypt the users' information with a security certificate is a good practise
- CardSpace interface makes handling Information Cards very easy
- Most of the users think that the Identity Provider can be a good technique to handle the users' credentials, but only when the user can choose their own Identity Providers
- To create an Information Card is a simple process, but most of the users need to spend some time to know the CardSpace graphic interface
- To use Information Cards eliminates the problems associated with username and password authentication such as if the user forgets their credentials
- The register process with Information Cards is very fast, when the user has a defined Information Card that satisfies the website policy
- CardSpace presents some problems, if it is executed on different browsers outside Internet Explorer
- Using one Information Card for different service providers makes the authentication process easier than using different usernames and passwords for every different service provider
- Since Security Tokens contain encrypted credentials of the user, the user feels safer when he/she provides the identity information

The figure 5.2 shows the obtained percentage for the evaluation of Information Cards versus usernames and passwords authentication. The 70% of the users think that Information Cards is a better authentication method than usernames and passwords authentication, but some of the users within this percentage also believe that CardSpace needs to provide a better support outside Internet Explorer. These users value the security of the use of Information Card authentication and the facility to manage their credentials. On the other hand 30% of the users are more reluctant to use this new technology and they prefer to continue using username and password, because they are familiar with the technology and they can use it in every different browser or Operating System.

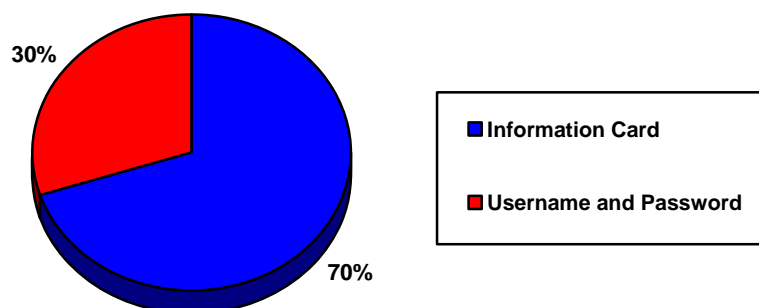


Figure 5-2: Information Cards versus Usernames/Passwords evaluation

## 5.5 Conclusions

---

In this chapter, the most important parts of the Federated Identity Management and the traditional Identity Management systems have been evaluated. A comparison between both models has been made in order to try to find the strengths and weaknesses of each model. In the first section, a process has been presented that shows how both Information Card authentication and username/password authentication has been tested. This section tries to evaluate the functionality of the system and shows the user how to make the different operations in the prototype. In addition to this, the use of managed cards has been tested. If the user wants to use a managed card, the IP has to be online and the user has to provide some type of authentication as a personal card, a username/password, a Security Certificate, and so on. In Section 5.2 the necessary requirements are analysed, in order to use Information Card technology in the user's computer.

This section also compares the different current browsers providing support for Information Cards, since CardSpace does not provide support for all the different types of browsers. Some of these browsers provide additional plug-ins that can be installed in order to provide the Identity Selector. In Section 5.3.1, a series of questions have been put together in order to evaluate the current Identity Management systems, such as Digital Certificates, Biometric, username and password, and so on.

In sections 5.3.2 and 5.3.3, an evaluation has been carried out by thirteen different users. This evaluation analyses the use of Information Cards and the security that this technology provides versus the username and password authentication. Additionally, the user experience with this prototype has been evaluated in order to know if the technology could be a standard for authentication. After analysing the obtained results, this evaluation can conclude that the use of Information Cards can be a satisfactory solution for the problems of authentication encountered in the current systems. As previously mentioned, a disadvantage of this technology is that the present prototype is not supported by all the current browsers as in the case of username and password, but this situation could be changed by standardising the use of Information Cards.

# 6 Conclusion

## 6.1 Achievement of aim and objectives

---

This thesis has presented an analysis of the current Identity Management systems, demonstrating the problems related with the security and functionality. This thesis has also introduced a FIM system in order to solve the problems related with the previous Identity Management models. In this section the thesis's aims, objectives, general conclusion and possible future work in this area, are discussed.

The first part of this thesis has covered a critical review of the current Identity Management systems. Different methods of authentication (Section 2.2) have been reviewed, such as the use of passwords, Biometrics, Digital Certificates, and so on. Current Identity Management systems take control over the user's identity information.

In order to solve this problem this thesis has reviewed the FIM system and Circles of Trust, where the user has the control of their authentication (Section 2.3 and Section 2.4). In a Federated Identity Management system a number of principles of data protection have been established, (Section 2.6) and the Laws of Identity (Section 2.7) moving the identity control on the user side.

This control on the user side is obtained by means of a number of properties (Section 2.8) which are based on the FIM model. Existing FIM models have been reviewed (Section 2.9) providing advantages and disadvantages of these models. In order to create a real prototype of this system, this thesis has reviewed a series of different standards (Section 2.10) that permit the implementation of the properties defined on the FIM model.

For this thesis, a prototype of FIM system has been designed (Section 3.3), where the user always has control over their credentials. The Identity Management system is based on the use of Information Cards containing the user's identity information. When the user wants to obtain a service from a SP, the user needs to provide their credentials, the user can select an Information Card and use this in order to send their identity information. In the prototype, an Identity Provider has also been introduced to prove that the user's identity information really belongs to that user and no other, and that the SP is a reliable party. The Identity Provider satisfies the law of justifiable party (Section 2.7.3) where a well-known party verifies the validity between the user and the SP.

The prototype has been implemented in a website allowing the creation of different user accounts in a database (Section 3.4). This prototype provides an Identity Management system based on the use of Information Cards. The user provides their identity information on a Security Token (Section 2.10.3) and the communication between the user and the website is established by an encrypted secure sockets layer SSL (Section 3.5).

A prototype of Identity Management system has been developed. This Identity Management system has been used to make a comparison (Section 5.1) between Information Cards and username/password models. The time it takes to provide the user authentication for every model has also been calculated. In the last part of this thesis, an evaluation of the user experience with the use of Information Cards versus the use of username and password authentication has been carried out.

## 6.2 General conclusion

---

The Internet is an inevitable part of everyday life, and the number of services increase every day, but at the same time the number of crimes committed online increases (Network Security, 2007). Users have to provide personal information in order to obtain a service and in some occasions, this information is not relevant to the service that they are going to obtain infringing the minimal disclosure law (Cameron, 2005). The use of Information Cards forces the Service Providers to specify the data that is required in order to provide the service. In addition to this, the user has the control over providing their identity information if he/she agrees with the service policy.

Another problem is the usage of username and password as an authentication method. This authentication method forces the user to remember one authentication for every different service, or in other occasions to use the same authentication for different services. With Information Cards authentication, the user does not have to remember any type of authentication information, he/she has a group of Information Cards representing the different information about the user. The user can then decide to use one card to obtain some services and other cards for other more relevant services.

Another problem related to the usage of username and password authentication is that some Service Providers can send the user's identity information through Internet without using any type of encryption. On the contrary, the user information is contained within a security token and protected with a security certificate when using Information Cards. CardSpace also limits the use of Information Cards within an encrypted secure sockets layer SSL.

It is also worth mentioning that in the current Identity Management system, the user has to spend a lot of time writing their personal information in order to register with a SP and most of the time this information is the same. The use of Information Cards facilitates this task for the user, improving the user experience and reducing the time that the user has to spend with this process (see Section 4.9).

As my personal conclusion, Information Cards can provide a high-quality solution in Identity Management systems. This model protects the privacy rights of the user, at the same time as providing an environmental-friendly and easy to use security system.

## 6.3 Future Work

---

For this thesis, a prototype of a Federated Identity Management system has been created. In order to create this prototype, a differentiation between two different parts has been made; the Identity Provider and the Service Provider. These two parties should be

implemented separately, since the Identity Provider has to be implemented by a well-know organisation such as the government, a bank, a public entity, and so on. The Service Provider on the other hand can be implemented for any company that provide a service to the user.

Some improvement to this prototype might be:

1. Improving the Identity Provider in order to supply managed cards in an automatic way for different users.
2. Improving the website to provide customised pages for different users depending on the Information Card supplied by the user.
3. Implementing Information Card support in a real website, since the one made in this thesis is only a prototype. In a real website, the server could give different services to the user such as account recovery, if the user has lost the Information Card, associate different Information Cards to one account, and so on.
4. Obtaining a certification authority (CA). For the prototype used in this document, a personal certificate is utilised and this raises a number of problems when opening the page in the user's browser, as the certificate is not reliable.

To my personal opinion, this work could be used as a basic framework to build a real SP that supports Information Cards authentication.



## 7 References

- Anonymous, (1980). The OECD Principles. Guidelines on the Protection of Privacy and Trans-border Flows of Personal Data', Paris.
- Bertocci V., Serack G. and Baker C. 2007. *Understanding Windows CardSpace, An Introduction to the Concepts and Challenges of Digital Identities*. Crawfordsville, Indiana: Addison Wesley US.
- Biskup J., Hielscher J. and Wortmann S. (2008). A Trust- and Property-based Access Control Model. *Electronic Notes in Theoretical Computer Science*, Vol. 197, No. 2, pp. 169-177.
- Bhargav A., Camenisch J., Gross T., Sommer D. (2007). User Centricity: A Taxonomy and Open Issues. The IST Project PRIME.
- Bhargavan K., Fournet C. and Gordon A. (2005). A semantics for web services authentication. *Theoretical Computer Science*, Vol. 340, pp. 102-153.
- Cameron K., (2005). The Laws of Identity. Web Services Technical Articles
- Claus S. and Kohntopp M. (2001). Identity management and its support of multilateral security. *Computer Networks*, Vol. 37, pp. 205-219.
- Coyle K. (2007). Identity crisis. *The Journal of Academic Librarianship*, Vol. 33, No. 4, pp. 512-514.
- Dean R., (2006). Identity management back to the user. *Network Security*. Vol. 2006, No. 12, pp 4-7.
- Demchenko Y., (2004). Virtual organisations in computer grids and identity management. Information Security Technical Report. Vol. 9, No. 1, pp 59-76.
- Galindo D. and Herranz J. (2008). On the security of public key cryptosystems with a double decryption Mechanism. *Information Processing Letters*, Vol. 108, No. 5, pp. 279-283.
- Hansen M, Pfitzmann A. and Steinbrecher S. (2008). Identity management throughout one's whole life. *Information security technical report*, Vol. 13, pp. 83-94.
- Iacono L., Wang J., (2008). Web Service Layer Security (WSLS). *Network Security*. Vol. 2008, No. 2, pp 10-13.
- I-En L., Cheng-Chi L. and Min-Shiang H. (2005). A password authentication scheme over insecure networks. *Journal of Computer and System Sciences*, Vol. 72 (2006), pp. 727-740.
- Josang A., Pope S., (2005). User Centric Identity Management. CRC for Enterprise Distributed Systems Technology.
- Khosravi S. 2008. *Professional IIS 7 and ASP.NET Integrated Programming*. Indianapolis: Wiley Publishing.
- Lo Iacono L. and Wang J. (2008) Web service layer security (WSLS). *Network Security*, Vol. 2008, pp. 10-13.
- MacDonald M. and Szpuszta M. 2005. *Pro ASP.NET 2.0 in C# 2005*. 1 ed. New York: Apress.

- Madsen P., (2004). Federated Identity and Web Services. Information Security Technical Report. Vol. 9, No. 3, pp 56-65.
- McMurtry C., Mercuri M., Watling N. and Winkler M. 2007. *Windows Communication Foundation*. Indianapolis, Indiana: SAMS.
- Mercuri M. 2007. *Beginning Information Cards and Cardspace*. New York: Apress.
- Network Security. (2007). In brief. *Network Security*, Vol. (2007), page 3
- Oasis. (2004). *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) Version 2* [online] Available from: <http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf>
- Olsen T. and Mahler T. (2007). Risk, responsibility and compliance in 'Circles of Trust' - Part I. *Computer Law & Security Report*, Vol. 23, pp. 342-351.
- Pfitzmann B. (2004). Privacy in enterprise identity federation - policies for Liberty 2 single sign on. *Information Security Technical Report*, Vol. 9, No. 1, pp. 45-58.
- Schneider R. 2006. *Microsoft SQL Server 2005 Express Edition For Dummies*. Canada: Addison Wesley US.
- Seoksoo K., Soongohn K. and Geuk L. (2006). Structure design and test of enterprise security management system with advanced internal security. *Future Generation Computer Systems*, Vol. 25, pp. 358-363.
- Shakir J. (2007). *Web Single Sign-On Systems*. [online] Available from: <http://www.cse.wustl.edu/~jain/cse571-07/ftp/webssso/> [Accessed June 5 2008].
- Smith, D., (2008). The challenge of federated identity management. *Network Security*, Vol. 2008, No. 4, pp. 7-9.
- Sullivan R., (2005). The case for federated identity. *Network Security*. Vol. 2005, No. 9, pp. 15-19.
- Vildjiounaite E., Kyllonen V. and Heikki A. (2006). Empirical evaluation of combining unobtrusiveness and security requirements in multimodal biometric systems. *Image and Vision Computing*, Vol. 27, pp. 279-292.
- Wolfgang H. and Helmut R. (2005). Federated Identity Management in Business-to-Business Outsourcing. *iPortalMais*, (2005) pp. 81-93.



# 8 Appendix A

## 8.1 Hardware and software requirements

---

This section outlines the hardware and software resources used in order to implement this thesis. The specifications of the hardware were:

- Intel Core Duo 1.73GHz.
- 1024 MB of RAM.
- Mobile Intel 945GM 224 MB.

In order to implement the Federated Identity Management system, the following software tools were used:

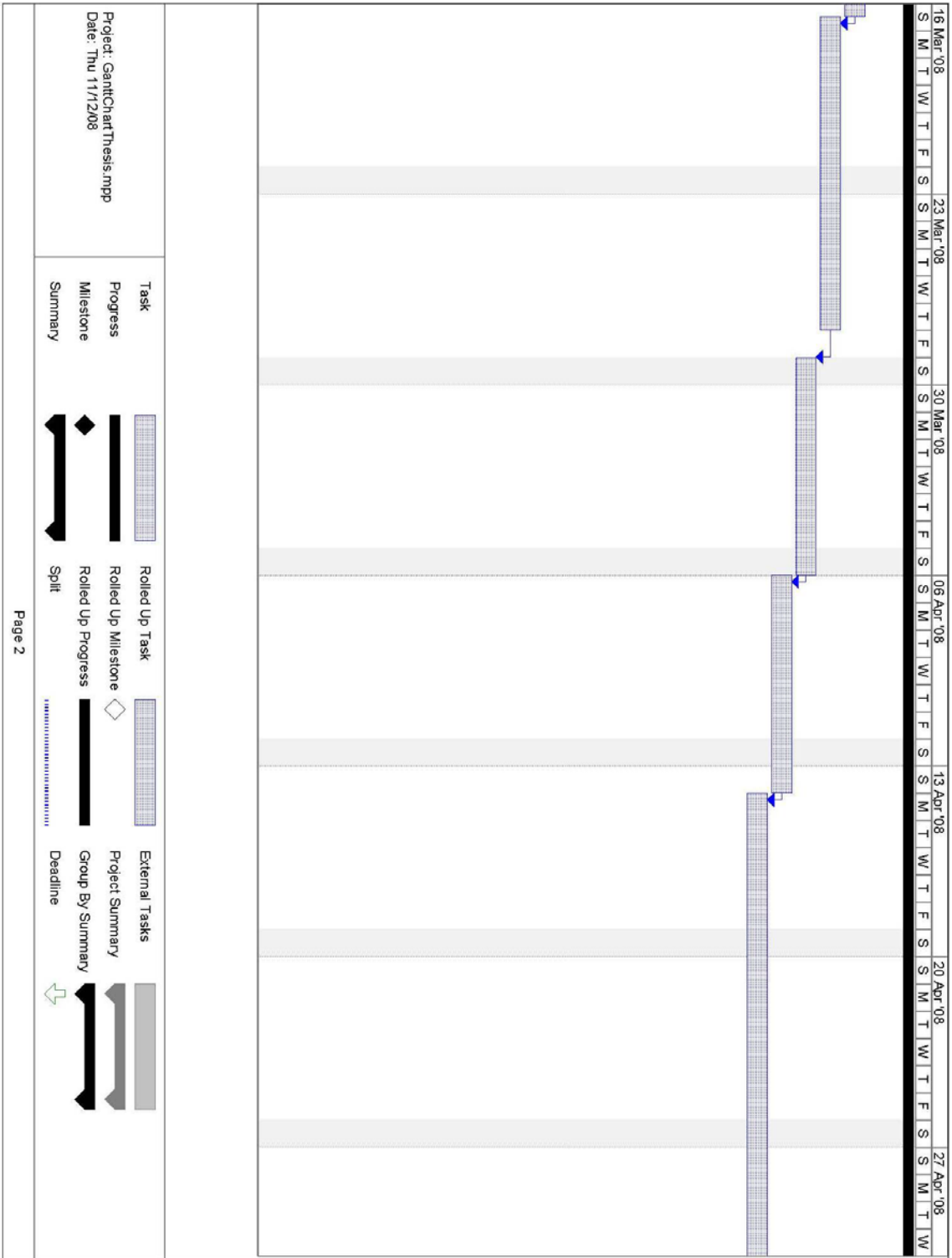
- Windows CardSpace and Windows Communication Foundation. The development environment Visual Studio 2008 express edition.
- The database has been implemented in SQL Server 2005 Express Edition.
- The website has been loaded in Internet Information Services (IIS) Server 5.1. The Windows XP operating system.

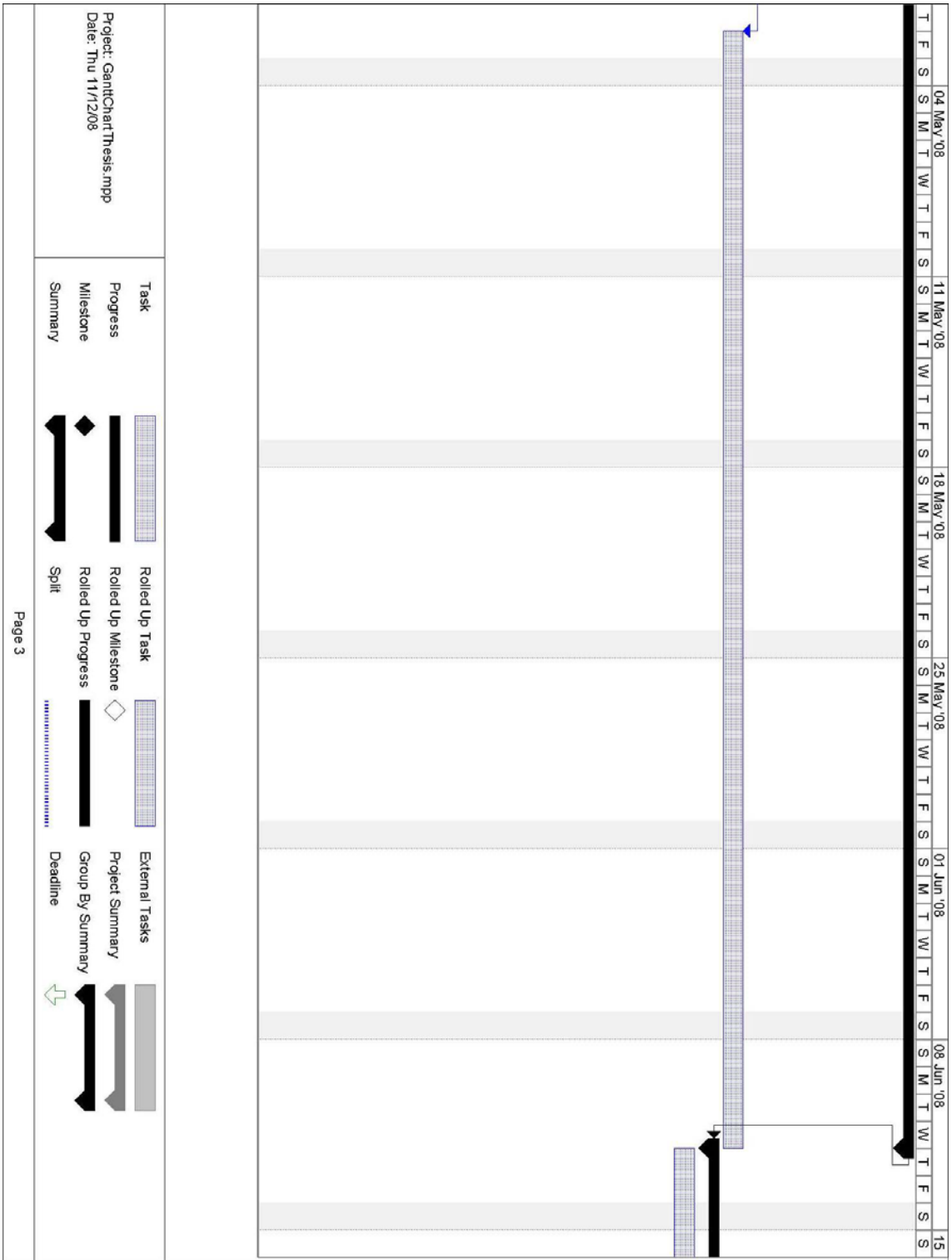
The programming languages that have been used are: C#, SQL and ASP .NET.

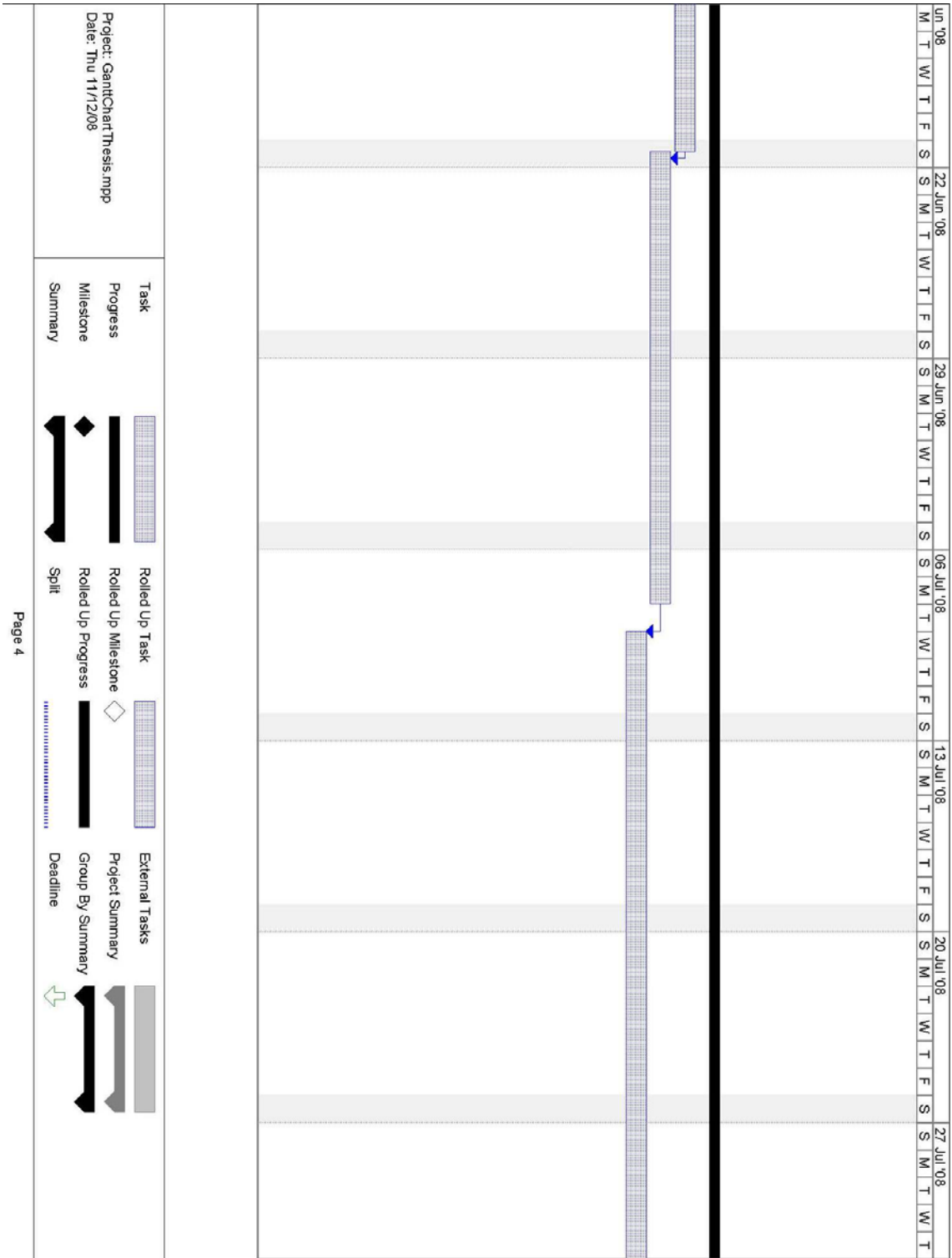
# 9 Appendix B

## 9.1 Project Management (Gantt chart)

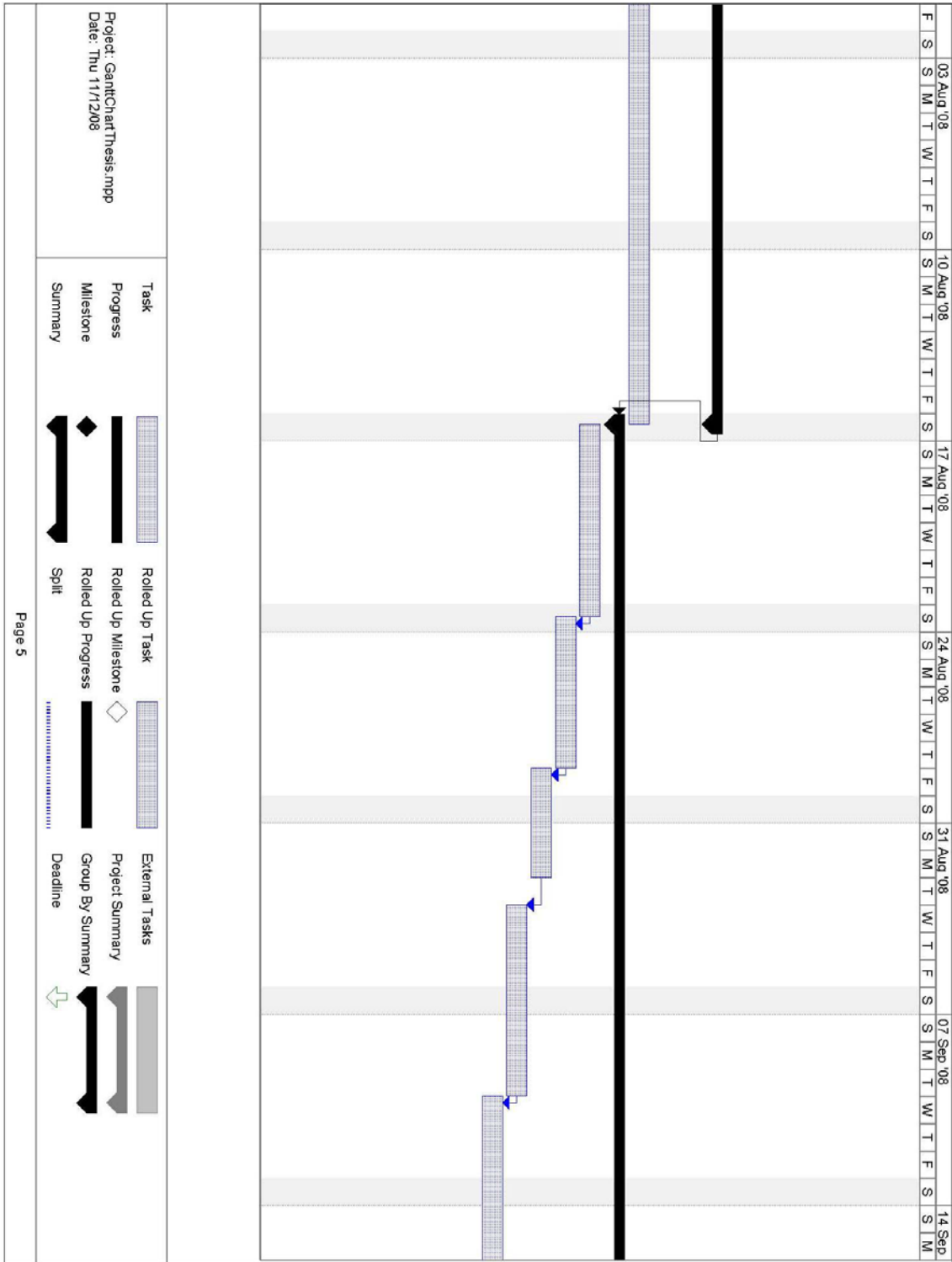


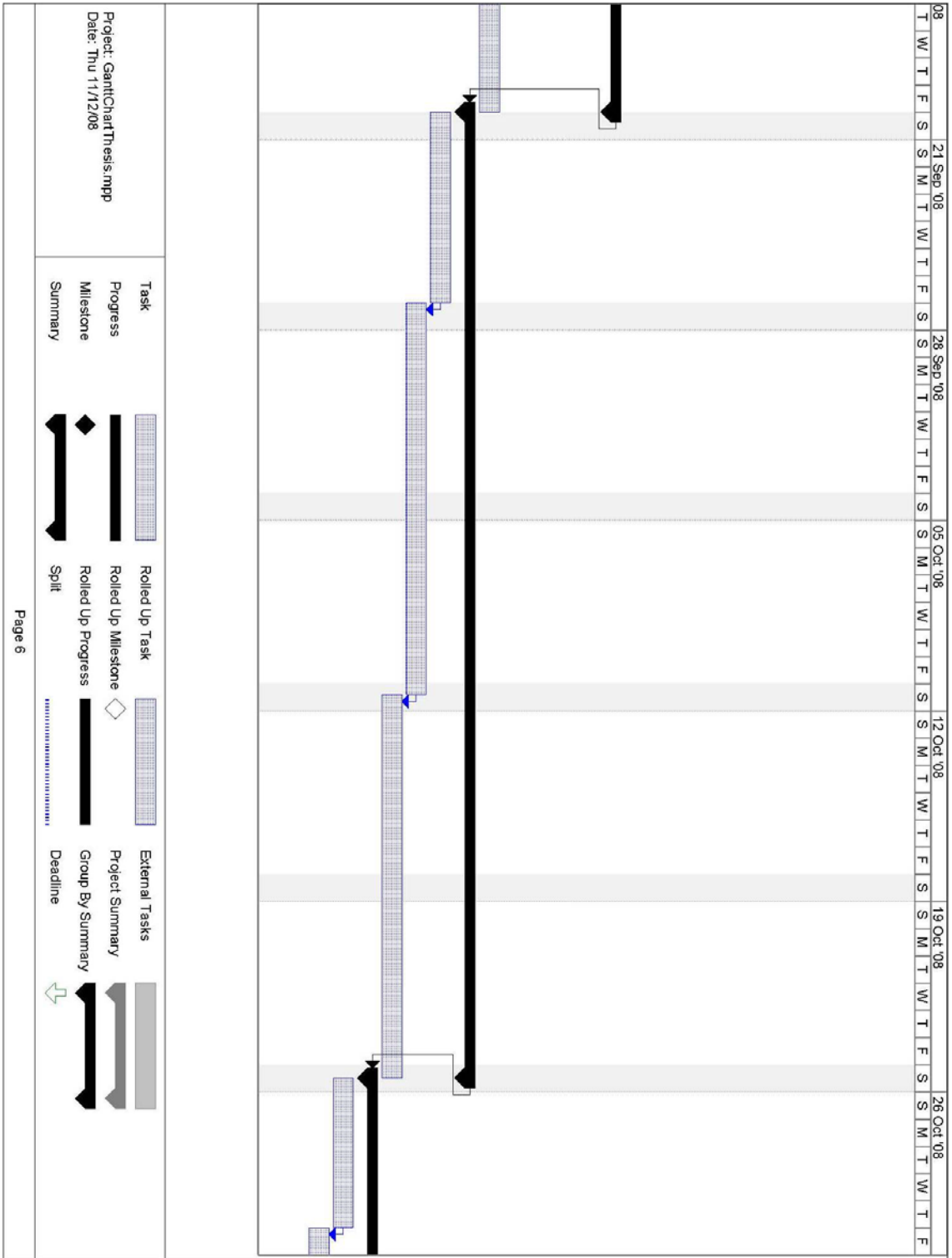


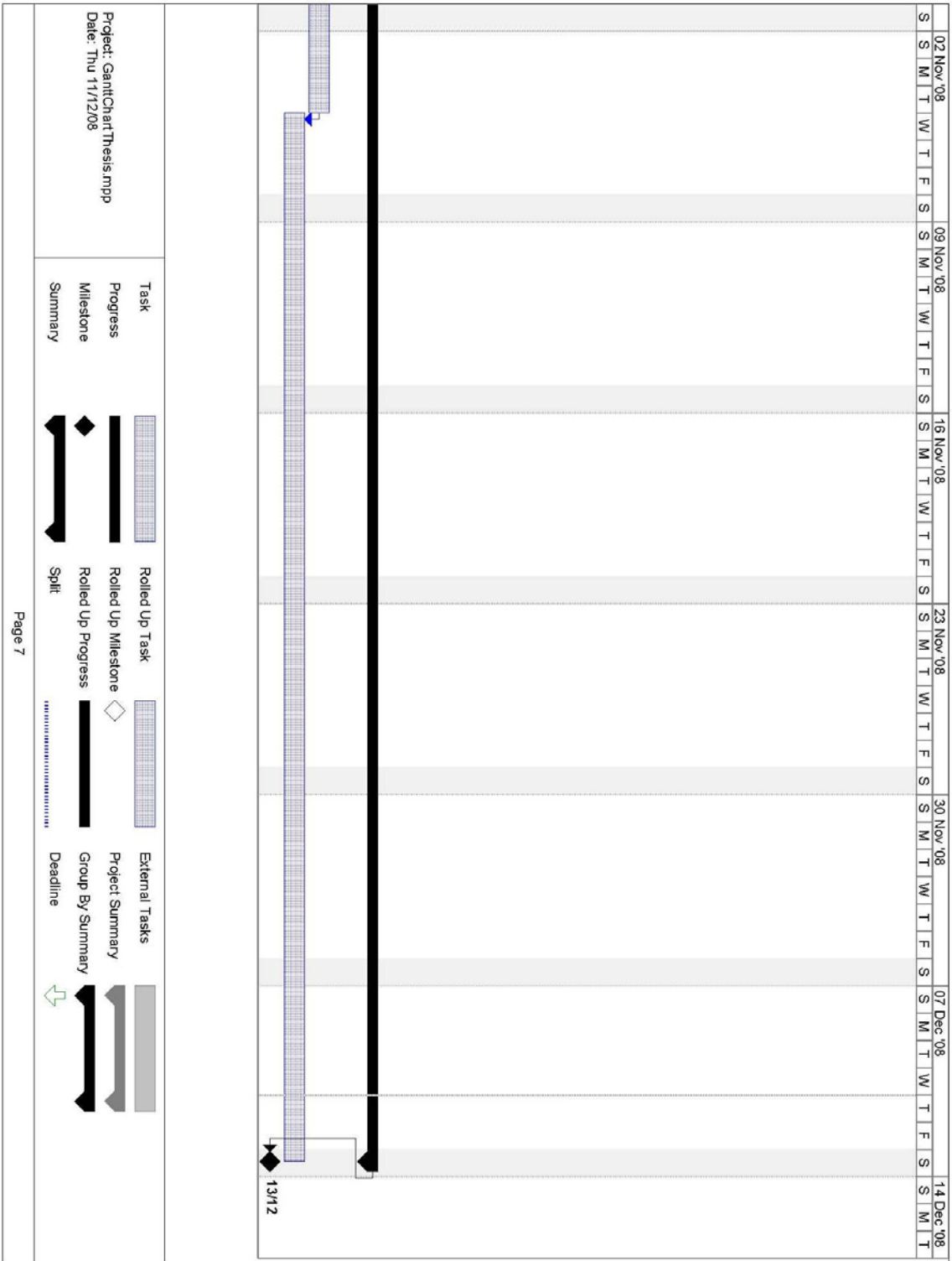












## 9.2 Project Management (Project Diaries)

---

NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

**Student:** Antonio J. Fernandez Sepulveda

**Supervisor:** Bill Buchanan

**Date:** 22 February 2008

**Last diary date:** <none>

**Objectives:**

Notes:

- First meeting with supervisor

Objectives:

- Start research about CardSpace Technology
- Start review about Identity Management systems and Federated Identity Management systems
- Start think about create a model that integrates Information Cards authentication

**Progress:**

--

**Supervisor's Comments:**

Create a website in order to integrate a prototype of this technology
---

## **NAPIER UNIVERSITY**

### **SCHOOL OF COMPUTING**

#### **PROJECT DIARY**

**Student: Antonio J. Fernandez Sepulveda**

**Supervisor: Bill Buchanan**

**Date: 9 May 2008**

**Last diary date: 22 February 2008**

#### **Objectives:**

- Continue reviewing literature and following thematic approach with all taken notes
- Gain competency in ASP .NET with some tutorials or books
- Create a prototype system that support Information Cards and user name and password authentication

#### **Progress:**

- Wrote Literature Review first part
- Experimented with Windows Communication Foundation
- Experimented with .NET 3.0 framework and CardSpace
- Installed Microsoft Visual Studio and SQL Server 2005
- Experimentation using agent timer method (for project evaluation)

#### **Supervisor's Comments:**

- Gather more information from research papers and publications
- He sent me some examples about ASP .NET in order to know this technology
- Develop the website in ASP .NET technology
- Register and Login the users by means of CardSpace and traditional user-name and password authentication
- The website has to remain the user's data, when the user signs to the website

## **NAPIER UNIVERSITY**

### **SCHOOL OF COMPUTING**

#### **PROJECT DIARY**

**Student: Antonio J. Fernandez Sepulveda**

**Supervisor: Bill Buchanan**

**Date: 5 September 2008**

**Last diary date: 9 May 2008**

#### **Objectives:**

- Design a prototype of Identity Provider
- Obtain a Security Certificate in order to provide Secure Socket Layer for the communication
- Create a implementation of the website
- Create a implementation of the FIM system and the traditional authentication
- Create a implementation of the Identity Provider

#### **Progress:**

- Finished Literature Review
- Finished design of prototype with support for CardSpace
- Experimentation using Information Cards authentication

#### **Supervisor's Comments:**

- Create website prototype with support for CardSpace in ASP .NET
- Store the user's information in SQL server database
- Implement managed cards for next appointment.
- Think about an evaluation for this technology against username and password.
- Improve the website design with different privilege for registered users.

## NAPIER UNIVERSITY

### SCHOOL OF COMPUTING

#### PROJECT DIARY

**Student: Antonio J. Fernandez Sepulveda**

**Supervisor: Bill Buchanan**

**Date: 10 October 2008**

**Last diary date: 5 September 2008**

#### **Objectives:**

- Design a timing test for registration and login with Information Cards
- Design a timing test for registration and login with username and password authentication
- Create a evaluation by every prototype
- Design a quantitative evaluation of Information Cards and CardSpace
- Design a qualitative evaluation of Information Cards and CardSpace for different users

#### **Progress:**

- Finished implementation of the website with support for Information cards and user-name/password
- Finished prototype of Identity Provider
- Hosted website with Security Certificate

#### **Supervisor's Comments:**

- Create a questionnaire to evaluate the technology
- To make the evaluation with different users
- Create a conclusion for every chapter

**NAPIER UNIVERSITY**

**SCHOOL OF COMPUTING**

**PROJECT DIARY**

**Student: Antonio J. Fernandez Sepulveda**

**Supervisor: Bill Buchanan**

**Date: 7 November 2008**

**Last diary date: 10 October 2008**

**Objectives:**

- Write up the abstract and introduction
- Write up the conclusion
- Review the document

**Progress:**

- Finished quantitative evaluation of CardSpace
- Finished qualitative evaluation of the Information cards versus username and password
- Finished prototype of Identity Provider

**Supervisor's Comments:**

- Improve format of the document
- Review grammar in the document



# 10 Appendix C

## 10.1 Evaluation Questionnaires

---

Please rate the following systems, with respect to usability and the level of confidence you have in security and privacy.

	<b>Bad</b>	<b>Average</b>	<b>Good</b>	<b>Excellent</b>
<b>Username and password</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>CardSpace</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Digital Certificate</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Face Recognition</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Fingerprint</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Hand Reader</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Iris Scanning</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>DNA Identification</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Keystroke Typing</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Signature Detection</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Voice Verification</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

### Username and password

Username and password is a simple authentication method that is used to gain access to a service for example in a website. The user has to remember their username and password in order to access the service.

Comments:

### CardSpace

CardSpace is an authentication system based on the use of information cards to provide the user's credentials. CardSpace provides a client interface called Identity Selector, where the user can make a number of operations such as creation, storage, sending, and so on. When a user wants to obtain a service, he/she can select the Information Card that contains the claims requested from the service.

Comments:

### **Digital Certificate**

Digital Certificate is an authentication method based on the use of public and private key cryptography and SSL. The digital certificate is issued by independent, recognized and trusted third party CA, guaranteeing the service to be what it claims to be. The digital certificate binds this service with its public key. The digital certificate contains the service's name, the digital signature, the service's public key, expiration date, and so on. When digital certificates are in order, the browser establishes the secure connections.

Comments:

### **Face Recognition**

Face recognition is an authentication method based on the recognition of the user's face by means of a digital image. The system compares the image obtained as input with an image of the user recorded in the database.

Comments:

### **Fingerprint**

Fingerprint is an authentication method based on the impression of the bottom of the user's finger. The authentication system scans the fingerprint of the user and compares this with the user's fingerprint stored in the database.

Comments:

### **Hand Reader**

Hand Reader is an authentication method based on the geometry of the user's hand. The authentication system scans the unique geometry of the user's hand and compares this with the template stored in the database.

Comments:

### **Iris Scanning**

Iris Scanning is an authentication method based on a high-resolution image of the user's iris. The authentication system uses a camera with infrared illumination to obtain an image of the user's iris, this image is converted to a digital template and is compared with the previous template of the user stored in the database.

Comments:

### **DNA Identification**

DNA Identification is an authentication method based on the structure present in every human cell. The authentication system obtains a sample of the user's DNA (the sample can be obtained from blood, saliva, hair, semen or tissue) and compares this to the previous template of the user stored in the database.

Comments:

### **Keystroke Typing**

Keystroke Typing is an authentication method based on measuring the time a key of the keyboard is held down and the duration between taps when the user writes their authentication. When the user tries to write their authentication, the system measures their results with the pattern stored in the database.

Comments:

### **Signature Detection**

Signature Detection is an authentication method based on the user's handwritten signatures. The authentication system scans the user's signature and compares this to the image signature stored in the database. Advanced Signature Detection systems can check rhythm, acceleration and pressure of the user's writing.

Comments:

### **Voice Verification**

Voice Recognition is an authentication method based on the user's voice tone. The authentication system compares the user's voice to the pattern stored in the database.

Comments:

Please evaluate the use of Information Card authentication. Please rate your experience on the scale bad to excellent for the following questions. If you have any comments, please provide these in the space provided.

	<b>Bad</b>	<b>Average</b>	<b>Good</b>	<b>Excellent</b>
<b>Would you consider it easy to find the pages for info card and password login?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it easy to register on the website with Information Cards?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it easy to login and log-out with Information Cards?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you consider login with Information card to be a fast authentication?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you consider it good to encrypt your identity information with a Security Certificate?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it easy to use Windows CardSpace?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it secure to use an Identity Provider to handle your identity information?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it easy to create an information card?</b>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Comments:

Please rate your experiences of using Information Cards as opposed to the traditional username and password to security. Please add any comments at the end of the questionnaire.

	Yes	No
<b>Do you consider Information Card easier to handle than Username and Password?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you consider Information Card to be a faster method than Username and Password?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it easy to login and logout with Information Cards?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you consider it good to encrypt your identity information with a Security Certificate?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find CardSpace easy to use in different browsers?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it easy to use Windows CardSpace?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you consider Information Card technology to be a more secure method than username and password?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you consider encrypting the identity information with a Security Certificate to be a secure communication?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it useful to have different cards for different services?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you think it is better to use one card for different services than to remember different usernames and passwords?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you find it easy to set up Information Cards in your computer?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you think it is easier to use an information card than to remember a username and password to login to a service?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you consider Security Token to be a secure method to exchange information?</b>	<input type="checkbox"/>	<input type="checkbox"/>
<b>Do you think that Information Card would be a good replacement for username and password authentication?</b>	<input type="checkbox"/>	<input type="checkbox"/>

Comments: