Consolidation of Results

Deliverable D 12.1

Document Identification	
Date	21/08/2015
Status	Final
Version	1.4

Related SP / WP	SP 1, SP 2, SP 3, SP 4, SP 5	**Document Reference**	D12.1
Related Deliverable(s)	D21.1 … D52.4	**Dissemination Level**	Public
Lead Participant	IFAG	**Lead Author**	Dr. D. Houdeau
Contributors	ATOS, EEMA	**Reviewers**	TUD, FhG

This document is issued within the frame and for the purpose of the FutureID project. This project has received funding from the European Unions Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 318424

1. Abstract

Report D 12.1 collects and summarizes the results from the Sub-Projects SP 2, SP 3, SP 4 and SP 5. In addition D 12.1 consolidates the results and pinpoints the outcomes. This lays the basis for identifying insights and lessons learned as well as for forming them into general design requirements and principles that will augment those already identified in WP 2.2 and WP 2.3. D 12.1 helps the subsequent evaluation tasks to navigate the project and provides an important alignment of results obtaining from theory, research and experiences from practical application of the methodology.

2. Document Information

2.1 Contributors

Name	Partner
Dr. Detlef Houdeau, Julia Ertl	Infineon Technologies
Jon Shamah	EEMA
Charles Bastos, Nuria Ituarte Aranda	ATOS

2.2 History

Version	Date	Author	Changes
0.1	29/05/2015	Dr. Detlef Houdeau	Initial version
0.11	03/06/2015	Julia Ertl	Structure, 1st draft
0.2	13/07/2015	Julia Ertl	Content chapter 8
0.3	20/07/2015	Julia Ertl	Appendix
0.4	27/07/2015	Julia Ertl	Content chapter 6, chapter 9, List of Tables
0.5	28/07/2015	Dr. Detlef Houdeau	Table of Acronyms
1.0	29/07/2015	Julia Ertl	Clean version
1.1	30/07/2015	Jon Shamah	Quality check
1.2	31/07/2015	Julia Ertl	Final version
1.3	10/08/2015	Jon Shamah Nuria Ituarte Aranda Julia Ertl Dr. Detlef Houdeau	Chapter 7.1 Chapter 7.4 Chapter 7.2, 7.3 Chapter 7, 9
1.4	21/08/2015	 Jon Shamah Nuria Ituarte Aranda Julia Ertl Dr. Detlef Houdeau	Final reviewed with the following changes: Overwork chapter 7; Merge chapter 7&8; Overwork chapter 6; Overwork chapter 8 (previous 9)

3. Table of Acronyms

AF	**A**ccess **F**ilter
AIS	**A**pplication **I**ntegration **S**ervice
APDU	**A**pplication **P**rotocol **D**ata **U**nit
API	**A**pplication **P**rogramming **I**nterface
APS	**A**uthentication **P**rotocol **S**pecification
B2B	**B**usiness to **B**usiness
B2C	**B**usiness to **C**ustomer
CAdES	**C**MS **A**dvanced **E**lectronic **S**ignature
CAPS	**C**redential specific **APS**
CEN	**C**omité **E**uropéen de **N**ormalisation
DNSSEC	**D**omain **N**ame **S**ystem **S**ecurity **E**xtension
DSS	**D**igital **S**ignature **S**ervice
EAC	**E**xtended **A**ccess **C**ontrol
EAI	**E**nterprise **A**pplication **I**nfrastructure
EC	**E**uropean **C**ommission
eIDAS	**e**lectronic **ID**entification, **A**uthentication and **S**ignature
epSOS	**e**uropean **p**atients **S**mart **O**pen **S**ervices
eSENS	**e**lectronic **S**imple **E**uropean **N**etworked **S**ervices
ETSI	**E**uropean **T**elecommunication **S**tandards **I**nstitute
EU	**E**uropean **U**nion
GUI	**G**raphic **U**nit **I**nterface
G2C	**G**overnment to **C**ustomer
HCM	**H**uman **C**omputer **I**nterface

HMAC	**H**ash **M**essage **A**uthentication **C**ode
HSM	**H**igh **S**ecure **M**odule
HTML	**H**yper**t**ext **M**arkup **L**anguage
I/O	**I**nput**/O**utput
ICT	**I**nformation and **C**ommunication **T**echnologies
IFD	**I**nter**f**ace **D**evice
IL	**I**ntermediate **L**anguage
JCA	**J**ava **C**ryptography **A**rchitecture
JEE	**J**ob **E**xecution **E**nvironment
NIST	**N**ational **I**nstitute for **S**tandardization and **T**echnologies
OASIS	**O**rganization for the **A**dvancement of **S**tructured **I**nformation **S**tandards
PACE	**P**assword **A**uthenticated **C**onnection **E**stablishment
PAdES	**P**DF **A**dvanced **E**lectronic **S**ignature
PC/SC	**P**ersonal **C**omputer/**S**mart **C**ard
PEPS	**P**an **E**uropean **P**roxy **S**ervice
PKCS	**P**ublic **K**ey **C**ryptography **S**tandards
REST	**R**Epresentational **S**tate **T**ransfer
SAML	**S**ecurity **A**ssertion **M**arkup **L**anguage
SCT	**S**imple **C**redential **T**ransformer
SICCT	**S**ecure **I**nteroperable **C**hip **C**ard **T**erminal
SOAP	**S**imple **O**bject **A**ccess **P**rotocol
STORK	**S**ecure Iden**t**ity Acr**o**ss Bo**r**ders Lin**k**ed
SSO	**S**ingle **S**ign **O**n
TLS	**T**ransport **L**ayer **S**ecurity

UAS	**U**niversal **A**uthentication **S**ervice
UI	**U**nit **I**nterface
URL	**U**niform **R**esource **L**actor
WP	**W**ork **P**ackage
XAdES	**X**ML **A**dvanced **E**lectronic **S**ignature
XSLT	E**x**tensible **S**tylesheet **L**anguage **T**ransformation

4. Table of Contents

5. List of Tables

6. Introduction

For the review of the key findings, the status of work and for the navigation of the whole project a consolidation of the results along SP 2, SP 3, SP 4 and SP 5 is needed. D12.1 reflects the status of FutureID at t=33 months.

Chapter 7, Summary Results starts with a complete brief description of the summary of each task and deliverable, the key findings and the key results along the work packages. In the chapter 7.1, 7.2, 7.3 and 7.4 all deliverables are shown in a table format with the scope of the deliverable, the targets, the results and the current status of the deliverable in a traffic light mode. If more detail of the deliverable is needed, it is indicated to have a look on the chapter conclusion of the requested deliverable.

In the appendix (chapter 9) a complete picture of all deliverables in a traffic light mode is painted.

7. Summary Results

The deliverables in **SP2** provide a complete understanding of the methodology used and the rationale behind FutureID. This is a fundamental exercise to ensure that all partners concur on the design principals and architecture of the project. It describes the architecture and guidelines to be used in the other subprojects in FutureID and ensures that there are no misunderstandings between partners or divergence of components.

WP21 (Vision, Approach and Inventory) established the baselines needed to reach a successful project conclusion. The aim of Task 21.1 was to provide the project, and also the community at large, with a common understanding of terminology used in identity related activities. Analysing current and past EU projects, it was found that there was a significant variance in the use of these terms. Terminology development resulted in 150 terms being defined and used as a base terminology. Tools were established to facilitate their use. Task 2.2 surveyed the state of the art technologies available at the start of the project (1998-2012) and those expected to be developed through research after the project started (2012-2015).Tasks 21.3 and 21.4 established the high level vision of the FutureID and then, through a series of evolutions, determined a reference architecture to meet the business and technical requirements from other tasks and work packages. Due to the resultant complexity of the architecture, it was decided to extend Task 21.4 through the length of the FutureID project to allow for evolution of the reference architecture to meet the requirements of other technical and non-technical work packages. Task 21.5 looked at business and use cases, as real application scenarios, to determine where FutureID can improve existing services with the integration of eID solutions. Five business use case models for evaluated, as well as the two use cases that were planned to be implemented as pilots.

The WP22 (Requirements Analysis) role was to analyse the requirements of the FutureID infrastructure components and utilization. They looked at the technical, security, privacy, usability, socio-economic, legal, and accessibility/inclusion aspects of FutureID and documented the conditions and circumstances that FutureID would need to meet to be compliant with these domains. There was extensive consideration of results from the SSEDIC and SKIDentity projects. Task 22.1 (Technical) made a detailed analysis of the requirements for the eID service, the eSignature service, the client platform, and browser, as well as user-side and back-end server integrations. Task 22.2 (Security) used a similar approach to that taken in the Common Criteria methodology and a careful definition of system boundaries was needed to be undertaken to address the specific security aspects of the FutureID distributed architecture. A crucial outcome of this analysis was that new system-level requirements, due to this distributed architecture would need to be addressed. Task 22.3 (Privacy) collaborated with Task 22.6 (Legal) to ensure that FutureID exceeded the legal requirements in providing a ‘privacy-friendly’ solution, while also considering transparency. Special attention was paid to data-protection, e-

commerce and eSignatures. Task 22.4 (User-requirements) took into account security and privacy and produced user requirements defined and structured along ISO standard definitions. Task 22.5 (Socio-economic) conducted a stakeholder analysis for the B2B and B2C markets taking into account relevant socio-economic theories for identity management. Ethical as well as tangible and intangible business requirements were considered and defined. This task focused on stakeholder and end-user acceptance, as key success factors, and alignment with business processes and constraints. Task 22.7 “Accessibility and Inclusion” looked at the current standards and guidelines for different groups of people with disabilities, including current EU regulations. This task ensured that services using FutureID should be built with accessibility and inclusion in mind. Task 22.7 worked closely with Task 22.4.

Questionnaires and example criteria for each domain were created to help evaluating the FutureID solution at the end of the project.

WP23 (Design Guidelines) was intended to be the basis of consistent work across all partners and to provide guidelines to foster high quality and good communication throughout the project. Task 23.1(User Interface) provided the guidelines for the user interface (UI) taking into account Human-Computer-Interaction (HCI) techniques in the form of design patterns. Design principles and usability heuristics were stated as recommendations and guidelines for the UI developers. Task 23.2 (Software Development) provided the guidelines to ensure a smooth integration of the developed components for FutureID. This ensures a synchronized development process in such a large project as FutureID. Task 23.3 (Software Documentation) built on current standards and provided guidelines for a consistent documentation of the project in terms of results, manuals, templates and communications.

WP24 (Protocols for future eID solutions) had the objective to develop protocols and tools that can be used in the next generation of privacy enhanced eID solutions. Task 24.1 (Extending Languages and Tools) had three avenues of research: 1) How non-cryptographic operators contribute to structure and format messages; 2) Sufficient conditions for vertical protocol composition. It established seven easy to check syntactic conditions for static vertical protocol composition; 3) automatic code generation for protocol implementations, starting with high level languages and automatically generating implementations and formal models. Task 24.2 (Methods and Languages for privacy goals) established a privacy analysis method, together with its formulation, and also semantics for claims languages to allow reasoning over them. The task introduced a new way of specifying privacy goals for formal verification (Alice and Bob) and extended this notion for transitional systems (privacy in the presence of an active intruder). Task 24.3 (Privacy friendly audit and data-handling) conducted research on data-handling mechanisms and on audits (privacy co-existing with transparency). The task designed appropriate cryptographic protocols for data disclosure minimization. Task 24.4 (Privacy-friendly revocation) dealt with credential revocation in advanced identity schemes that support pseudonymous authentication. A particular problem when transactions from the same user are intended to be unlinkable. This task developed a pairing-based group signature scheme with

controllable linkability. Task 24.5 (Usable Privacy) looked at previous approaches to usable privacy as pursued by the EU project PRIMELIFE and eID privacy in SEMIRAMIS. It has established a novel research approach called 'Cognitive Privacy'.

The **SP3** subproject defines and implements the specific client components required for the FutureID framework. This includes capability to utilise a wide range of devices, smartcards and other hardware and to perform the functions, for the end user, to take full advantage of the unique capabilities of FutureID.

The deliverables in SP3 provide a comprehensive picture of the high level design of the FutureID client. It is based on 6 pillars with the IFD, SAL, eSign, GUI trustworthy client platform and the dispatcher and transport, such as HTTP, SOAP, TLS and others.

WP31 (Interface Device Service) developed an open source reference implementation for a comprehensive Interface Device layer according ISO/IEC 25727-4. This implementation supports a range of client platforms and a variety of interface devices such as smart card readers (terminals) based on PC/SC as well as mobile devices which support NFC and/or OpenMobile-API. Task 31.1 displays the requirements for ubiquitously usable IFD services. This is needed to develop the interface and the module specification for the IFD services, as collected in task 31.2. In task 31.3 the open source reference implementation was done in Java. This supports the operating systems Windows, Linux, Mac and Android as well as the terminal interfaces PS/SC, NFC and Open Mobile API. Task 31.4 shows the system test to verify the IFD specification.

WP32 (eID Services) established the FutureID client, which runs as a middleware between the credential and the identity infrastructure. The implementation provides the user interface, the security and communication protocols and the application flow. The FutureID service supports authentication services based on various authentication tokens. Task 32.1 collects a survey and analysis of existing eID and credential systems as a start point for the requirement analysis, which is the scope of task 32.3, followed from the interface and module specification, as mirrored in task 32.3. The generic implementation includes a Service Access Layer (SAL) that provides a generic interface for common credential services as reported in task 32.4. Task 32.5 provides specific implementations such as EAC, generic crypto and the new protocol for ABC4Trust using Idemix/IRMA credentials. Task 32.6 displays the Java tool development, which reads ISO/IUEC 7816-15 structures on smart cards and creates CardInfo files according ISO/IEC 24727-3. Task 32.7 highlights CardInfo files from Austria (e-Card), from Estonia (eID Card), from Germany (Patient Card, Health Professional Card, light, HBCI-Card, S-Trust Signature Card, D-Trust Signature Card and Telesec Signature Card), from Belgium (BELPIG) and from Peru (eID Card).

WP33 (eSign Services) developed the FutureID eSign service, from the requirement analysis along the specific interfaces and eSign service module to the implementation. Task 33.1 was

intended to collect all requirements of the eSign service module in the FutureID client framework, based on the DSS protocol and the signature formats, such as CAdES, XAdES and PAdES. Along task 33.2 the interfaces of the eSign service module and the underlying framework were defined, including the data types. In task 33.3 the implementation has fully reached all objectives, e.g. the framework for a local host signature server. This supports the relevant subset of the OASIS DSS specification and several signature formats. In task 33.4 the integration of the localhost signature gateway into the FutureID client and the testing have fully reached all objectives. The system test was done with the sending of predefined OASIS DSS requests to the FutureID client.

WP34 (User Interface) has the objective to realize the open source reference implementation for the user interface for the different supported platforms and to optimize the user interface with respect to usability and accessibility, as displayed in task 34.1, 34.2 and 34.3. Java and Swing have been identified as the most suitable technology base for the user interface implementation due to the cross platform availability and adherence to the host system's look and feel. For Android the implementation needs a bridging technology as described in task 33.4. The task 33.5 shows an implementation which assumes client-server architecture with XHR calls. The user interface is implemented in HTML using the JavaScript framework. The implementation is based on a RESET service prototype. In task 34.6 a GUI for the authentication process was created based on JSON messages that work with a prototype of a RESET-service.

WP35 (Trustworthy Client Platform) reflects secure applications and transactions in a close look of the different technologies such as TEE and built-in OS or MTM. Task 35.1 mirrors existing eID and credential systems and their trustworthy properties along a requirement analysis. In task 35.2 various methods and elements for secure execution environments and their access control properties were investigated. Due to the differences of mobile and stationary devices, several architecture options have been presented for both device classes and the achievable level of trustworthiness has been displayed.

In Task 35.3 the implementation is realized on OpenMobile API as a single interface for secure environment access on mobile devices. Several plug-in terminals have been implemented. It allows access via NFC and allows addressing USB hardware tokens. The tests have passed the Global Platform compliant test period. On the other side the resistance of the mobile client platform against SW attacks were shown. In task 35.4 two issues can be solved by letting the user of a credential locally store some backup data that is only needed for (re-)issuing of a credential: a) the credential and all information needed to use it are leaked to the adversary. The owner of a credential should be able to revoke a lost credential to prevent further damage; b) in a mobile environment, users often copy authentication credentials to portable devices and carry these credentials with them. This procedure generates a higher risk of losing the credentials together with a device.

WP36 (Browser Integration) provides an interface between the user's browser and the FutureID client. This is needed to get web site access with the FutureID client. Task 36.2 provides the architecture for the browser integration. Task 36.3 goes further with the implementation of the browser integration solution, including interfaces and browser specific modules. In task 36.4 the integration of other modules of the FutureID client in the framework is started.

WP37 (Client Testbed) provides a testbed for the developers in the FutureID project to make sure that a certain level of code quality is met and the SW is developed properly. Task 37.1 and Task 37.2 deal with two objectives, with the definition and provision of tools and methods for the testing of client components itself and the provision of the so called testbed to the project consortium. The work for the Tasks 37.3, 37.4 and 37.5 is ongoing.

The **SP4** subproject defines and implements the specific components required for the FutureID framework infrastructure to deliver the needed functionality. This includes capability to utilise a wide range of devices, smartcards and other hardware, and to perform the functions, for the end user, to take full advantage of the unique capabilities of FutureID. This includes the Identity Broker and the interfaces to the application services which are offering access to users via FutureID. Different modes of FutureID operation are taken into account, as are the varied enterprise application platforms that are currently available for service providers

SP4 provides different components aiming at making it easy for Service Providers to use various authentication and credential technologies. The set of components comprises the Identity Broker (WP 4.1), the Universal Authentication Service (WP 4.2), the Trust Service (WP 4.3), the Application Integration Service (WP 4.4) and the Server Testbed (WP 4.5).

WP 4.1 addresses the **Identity Broker** which is a central component within the FutureID infrastructure. It aims at making it easy for Service Providers to integrate various authentication and federation technologies. Furthermore the Identity Broker aims at improving usability and privacy by allowing the User to select and manage its identities and related trust issues.

Identity Broker consists of a central Broker Service, which integrates various Federation Services and Authentication Services, such as eID-services for the German eID card, authentication and proxy services (PEPS) from the STORK project, the Austrian MOA-ID service and authentication services provided by popular social networks, such as Facebook and LinkedIn for example. The Identity Broker allows the different authentication services to be available via standardized federation protocols, such as SAML (Web Browser SSO with Bearer Tokens or Holder of Key Binding).

Among the significant results created in WP 4.1 are the development of a technical specification for the central Broker Service of the FutureID Identity Broker, which allows integrating arbitrary Authentication Services and Federation Services. This component provides a generic Authenticate interface, which is an enhanced version of a web service based interface

standardized in CEN 15480-3. The enhancements and refinements may be brought back to CEN and could be integrated in a future version of the European Citizen Card specification.

The availability of the identity broker in dispatcher mode (T41.3) and in claim transformer mode is other significant result of this work package. This is the foundation to integrate authentication backends directly. The results of Task 41.3 are provided in the form of services accessible by authentication backends such as STORK-PEPS and federation services such as SAML IdPs. In order to simplify the integration of authentication backends, an additional service transforming the Authenticate interface into an easier to consume REST service has been created.

WP 4.2 addresses the **Universal Authentication Service** which provides an innovative approach of facilitating a universal Identity Provider, supporting arbitrary authentication mechanisms. The component introduces the Authentication Protocol Specification (APS) language which provides a high-level description language to specify authentication protocols. This avoids the implementation of all authentication protocols directly in a certain programming language, APS files will be executed by the Execution Environment which maps certain commands to Basic Services.

WP 4.2's first results are the list of requirements (among others functional and privacy requirements) for the Universal Authentication Service and the Authentication Protocol Specification (APS) language; the research on several protocol specification languages and verification tools; the formal definition of a suitable APS language (Future AnB). Future AnB extends AnB (Alice and Bob notation) with key features such as selection structures, database manipulation and formats as a novel way to distinguish messages with different semantics.

The specification of Basic Services and Execution Environment; the specification of EAC and PACE in Future AnB, the design of basic data structures for the implementation of Future AnB and the mentioned requirements are the basis of the Basic Services development (Task 42.4), which provide building blocks for authentication protocols running in the JEE (Job Execution Environment). The different authentication protocols are composed of a rather limited set of Basic Services. In this way, arbitrary authentication protocols can be supported by combining the limited set of basic functionality.

Basic Services provide functions that (D42.5):

- Enable the usage of common security algorithms and protocols (such as HMAC or Diffie-Hellman Key Agreement), invoke cryptographic operations for both symmetric and asymmetric functions (like encryption and signing) via an interface that wraps JCA.
- Enable the usage of HSMs to accelerate and perform signing and encryption via an interface for HSMs that is compliant with JCA.
- Generate APDUs for smart cards.

The implementation of the Basic Services assumes the reutilization of existing building blocks, such as the eID Services implemented in WP32 as well as existing on the market PKCS#11 providers.

Task 42.5 also defines and develops an intermediate language (IL), for the execution of APS files is the core part of the JEE. It influences the interfaces between the JEE, the basic services and the APS translation. The significant elements are as follows:

- Data Formats for exchange between APS, Basic Services and the authentication protocol
- Definition of Credential specific APS modules (CAPS)
- Job Execution with Java scripting mechanisms according to JSR-223

The main result in Task 42.6 is the implementation of the prototype translator from APS to Applied Pi Calculus (input language of ProVerif) and the implementation of the generator for Java/Javascript. A translator for AVISPA IF (OFMC)

D42.8 presents APS files for some selected protocols that are highly relevant to the electronic identity systems (eID) in general and FutureID in particular. By means of these examples, we show the effectiveness of encoding in APS a significant class of real-world protocols relevant to eID systems to define the message formats employed in these protocols, and formally verify by means of two state of the art verification tools (Proverif and OFMC)

WP 4.3 addresses the **Trust Services**. Within the FutureID ecosystem they are needed due to the necessity of many entities and protocols which require trust information. For example, during the connection to FutureID components the authenticity of these components has to be verified to avoid masquerading and man-in-the-middle attacks. Furthermore, it is often necessary to verify the result of a previous step in order to continue with the execution of a protocol, for example, verify a signature.

Trust Services will, hence, provide two basic services: (1) a trust repository based on ETSI TSL, which will collect all relevant trust service providers and trust services with their respective assurance levels; (2) a validation service that will be able to verify various kinds of assertions, for example electronic signatures or certificates.

The significant results for Trust services are the following:

- In D43.1, expectations towards FutureID were defined and linked with the main recommendations based on the functional aspects of trust.
- The description and requirements for the FutureID Trust Services module
- The trust infrastructure will be built using a new approach, which uses DNS with DNSSEC extension. This will lead to a globally scalable trust infrastructure that allows easy delegation of trust decisions and efficient querying of trust lists.

- The validation service is able to validate a broad range of state-of-the-art electronic signatures, like the advanced formats XAdES, CAdES and PAdES and their various long-term profiles, and also others like blind digital signature schemes.

WP 4.4 focuses on the integration of services and applications into the FutureID Infrastructure through the **Application Integration Services** (AIS) component.

The Application Integration Service is an access control system that handles authentication for the service provider's applications. The Application Integration Service covers two major roles in the FutureID architecture:

- Intercepting unknown users and requesting the FutureID infrastructure to authenticate them
- Receiving and validating credentials in order to set up authenticated sessions for users

WP 4.4 addresses the implementation of JBoss specific Application Integration Service that is oriented towards the epSOS pilot integration, which runs on JBoss server. In this way, this Application Integration Service is covering the market for Java applications servers.

Significant results achieved in WP 4.4 are the design of Application Integration Services architecture at a high level according to its integration into EAI solutions; the description of relevant requirements for AIS grouped in (technical, security, privacy, usability, socio-economic and legal); the FutureID AIS Architecture: a detailed description of every AIS module containing the functionality and operational environment of all them was described; the implementation of FutureID AIS component consisting of the Access Filter (AF) component which intercepts unknown users and requests the FutureID infrastructure to authenticate them and the Simple Credential Transformer (SCT) which receives and validates credentials in order to set up authenticated sessions for users. AF includes the FutureID client detection. Both components AF and SCT are sharing the SL component which manages the user session; the development of an assessment tool to increase success rates of FutureID integrations at a business level.

WP 4.5, **Server Testbeds**, addresses the Definition/Provision of tools and methods supporting development and the Provision of a testbed for backend evaluation. The Server Testbed is an essential part of the project as that will ensure that the components of the server part of FutureID are tested thoroughly. We decided to use the semantic wiki as a way to save requirements, test assertions and possibly also test cases. To make sure that also TAML tools can be used in future we try to orientate the structure of the assertions towards the Test Assertion Guidelines.

The WP provides a testbed which is available for the project consortium in order to validate the functions of the FutureID backend. It will be available as a virtual machine for the different developers of the project. However, it is also planned to have a public web service which shows the current start of the backend.

Significant results achieved in WP 4.5 are a first virtual machine with Jenkins and SoapUI has been setup and is available for project partners and the first version of the testbed wiki tools which is ready for use.

The **SP5** subproject's aim is to define and implement the two FutureID pilots that will demonstrate the use of the framework in real business cases. In the first business case, FutureID acts as the identity authentication component within the Large Scale Pilot ePSOS. In the second pilot, FutureID provides identity authentication for the Atos e-Learning Services for Enterprises. SP5 considers technical and trust implications and practical implementation.

WP 5.1 **Citizen Services**' objective is to demonstrate that the technologies developed in FutureID can be used and will provide a significant benefit for the provision of secure citizen services in Europe.

epSOS – a project co-financed by the European Commission for a panEuropean health data exchange cross border – was chosen to cover the citizen services pilot application of FutureID that will be extended with FutureID technology. Specifically, the FutureID technology is being integrated into the epSOS infrastructure in order to improve the currently developed enhanced security safeguards with eID based technology.

The use cases for WP 5.1 were defined in Task 51.1. This included a description of the epSOS architecture since it represents the deployment infrastructure of these use cases. The second task in WP51 breaks down the business use cases to a detailed conceptual and technical level.

The main results till now are the description of the epSOS architecture since it represents the deployment infrastructure of these use cases. The use cases to consider are: (1) patient authentication via FutureID in the country of affiliation, (2) securely exchange medical data in an end-to-end manner, and (3) signing patient consent via FutureID in the country of treatment. The architecture definition of the integration has been also specified.

WP 5.2 **Atos e-Learning Services for Enterprises**, focuses on the integration from Future-ID with Atos e-Learning Services for Enterprises with special focus on business viability.

D52.1 studied the requirements for identity management in the context of Atos e-Learning Services for Enterprises, and business environments, with special focus on business viability. It shows the integration of Atos e-Learning Services for Enterprises and establishes the integration architecture. It shows the scope of the integration of Atos e-Learning Services for Enterprises with FutureID and the general authentication process. It also describes the business scenario in which Atos e-Learning Services for Enterprises is possible. The overall scenario is a company that buys Atos e-Learning Platform and installs it in its servers. The document shows an overview of Atos e-Learning Services for Enterprises to demonstrate the viability of FutureID components in business scenarios, specifically with regards to Internet of Services. The services offered by this platform, its technical features and architecture are also described.

D52.2 establishes the technical specifications of Apache specific AIS implementation (it's the AIS implementation that will be used with this pilot) and the Atos e-Learning integration interface with FutureID infrastructure. It specifies how the Service Provider (Atos e-Learning) and the Identity Provider (STORK) interact with the FutureID infrastructure through the Apache specific AIS implementation. It provides a detailed description of the integration components and establishes the flow of messages between the different components. This document is the reference for the development of Apache specific AIS implementation and the integration of Atos e-Learning into FutureID infrastructure.

Significant results within WP 5.2:

- Provision of a technical blueprint of the Atos e-Learning Services for Enterprises requirements for FutureID components in Business Scenarios.
- Description of the requirements for identity management in the context of Atos e-Learning Services for Enterprises, and business environments, with special focus on business viability.
- Presentation of the integration architecture of Atos e-Learning services for Enterprises with FutureID. Specification of the interfaces and modules for the integration of Atos e-Learning Services for Enterprises into FutureID infrastructure.

Description of the technical specifications of Apache specific AIS implementation and the Atos e-Learning integration interface with FutureID infrastructure

The following chapters give an overview on all deliverables along the scope, the targets, the results and the status of the deliverable in a traffic light mode.

Short description of the used colors:

- Green = task successful achieved
- Orange = task in progress
- Red = task not started

7.1 SP 2 – Methodology

Summary - Target	Comments - Results	Status
D21.1	Terminology	
Good terminology is important for efficient communications in a project such as FutureID. This deliverable first develops requirements for the development and management of terminologies.	This deliverable describes the current state of affairs, intending the terminology work to span the whole period of the project. There is a very heterogeneous use of terms and understanding in identity management. This deliverable establishes a common understanding of terms and is useful for projects other than FutureID.	
D21.2	Technology Inventory	
The technology inventory is an extensive survey of the state of the art technologies, standards and implemented programs on eID in the public domain. It mirrors the development of international standards, technologies and implementation in states in Europe in the time window 1998 – 2012.	The deliverable gives an overview on all relevant international public funding programs as known today and finishes with the relevant trends in the context of FutureID. These trends include privacy enhancing technologies such as ABC4Trust, and components and efforts to improve interoperability, such as those in STORK and eSENS and other Large Scale Pilots sponsored by the European Commission.	
D21.3	Vision	
The FutureID project builds a comprehensive, flexible, privacy-aware and ubiquitously usable identity management infrastructure for Europe. By combining expertise, experience and skills of partners with multidisciplinary and complementary competencies, the projects' outcome will provide benefits to all stakeholders involved in the eID value chain.	Document displays the long term vision of the FutureID infrastructure for Europe. This addresses an open source eID client, and running on arbitrary desktop PCs, tablets and smart phones. The infrastructure provides additional value across B2C, G2C and B2B.	

D21.4	Reference Architecture	
The reference architecture constitutes a high level plan on how the project vision can be technically implemented. It also provides guidance to component implementers who need to understand how their piece fits into the whole.	Document shows a high level plan on how the project vision can be technically implemented and captures all relevant aspects, such as a wide variety of stakeholder types, existing services, infrastructures, technologies and credentials.	
D21.5	Business and Use Case Analysis	
The use cases show how the FutureID technology can be of use in Business to Business (B2B) and Business to Consumer (B2C)-Scenarios. This is an important task in the FutureID project, as it has been concluded that the lack of use cases and real application scenarios so far has been an important barrier to a wider success of eIDs.	This deliverable collects, analyses, and evaluates use cases for different application scenarios for the FutureID technology. It highlights the advantages of the innovative FutureID technology and how it can contribute to build a competitive approach for the European information industry, providing additional value across Europe's B2C and B2B sectors.	
D22.1	Technical Requirements	
The technical requirements of FutureID are very important for the success of the whole services and the underlying infrastructure. This document has successfully described the most important requirements on a high level.	Documents highlight the most important requirements on high level. Following these requirements allows flexible and ubiquitously usable infrastructure for secure authentication across border.	
D22.2 (v1.1)	Security Requirements	
For FutureID to act as a trusted service it is essential to establish a high level of security for all FutureID services. Therefore, this deliverable has analyzed the security problem definition of the FutureID system components, their distributed interaction and the threats to the overall system performance.	Document displays analysed security problem definitions of the FutureID system components, their interactions and the threats to the overall system performance. The common criteria methodology was used.	

D22.3	Privacy Requirements	
The FutureID project aims at developing and building solutions for an identity management infrastructure for Europe. Among others, privacy criteria will play a vital role for conceptualizations, implementation and operation of the FutureID system.	Document shows privacy criteria, which play a vital role for conceptualisation, implementation and operation of the FutureID system. Privacy by design concept, privacy protection goal approach, multilateral security paradigm and many other methods were used.	
D22.4	Usability Requirements Analysis	
Usability and user experience of a product can play a major role in determining its market success. These overall objectives, the scope and concrete tasks associated with the usability of the FutureID client are therefore critical.	Established usability standards were presented, which address privacy and security aspects for the client, conformity with user expectations, self-descriptiveness and controllability.	
D22.5	Socio-Economic-Requirements	
As FutureID's outcome provides benefits to all stakeholders involved in the eID value chain, it is essential to determine the requirements of these stakeholders to be able to consider them in the design process and to evaluate the end result.	Document is a guideline on social-economic requirements for the development of FutureID infrastructure. This document can be used at the end of the project as a tool to evaluate the project	
D22.6	Legal Requirements	
Legal requirements affect the development of the FutureID infrastructure and the provisioning of FutureID services. Considerable emphasis must be placed on data protection laws, as well as other areas of EU regulation which may impact the development of FutureID.	Report addresses the data protection directive and reflects the data protection law which is in overwork procedure on EU level. The second regulation is the eIDAS topic, which is along some delegated acts under progress.	
D22.7	Accessibility and Inclusion Requirements	

Accessibility and inclusion must be taken into account when developing different prototypes in the FutureID project. Project partners need to be informed about different aspects of accessibility when dealing with ICT.	Document displays the accessibility and inclusion requirements on ICT in general and in the FutureID in specific. It reflects different types of users, assistive technology. Document can be used for developing and testing.	
D23.1	Common User Interface Design Guidelines	
Some aspects of usability and usability engineering are universal. However, the developed software client and associated tools are required to account for the special needs of security tools.	Report shows user interface guideline in order to accomplish intuitive and usable demonstrators. It is guidance for general development based on concrete rules, examples and references.	
D23.2	Software Development Guidelines	
This guideline is very important, because FutureID works with real existing eID cards. In FutureID person related data of card holders is handled. Software development in FutureID underlies two partially contrary boundary conditions. Certification policies and partial Open Source issues and so require a large amount of flexibility.	Along with the Open Source initiatives the document addresses collaborative software engineering based on current state-of-the-art software engineering, and on change management. Document displays a broad range of tool kits.	
D23.3	Documentation Guidelines	
It is essential to provide guidance on the documentation required for certification and standards, such as Common Criteria, and the EuroPrIse.	The deliverable provides guidelines for documenting the artefacts created throughout FutureID. It is structured as a toolbox, consolidating guidelines for general requirements to documentation, user documentation, media used, and especially software documentation.	
D24.1	First report on research on protocols and tools for future eID solutions	

1st report on research on protocols and tools for FutureID solution	Report mirrors a toolbox of formal methods for modelling and verification of complex, composed protocols and their privacy features. The focus is on extending languages and tools for compositional reasoning.	
D24.2	Second report on research on protocols and tools for future eID solutions	
2nd report on research on protocols and tools for FutureID solution	Report mirrors a toolbox of formal methods for modelling and verification of complex, composed protocols and their privacy features. The focus is on establishing methods and languages for privacy goals.	
D24.3	Interim internal report on research on protocols and tools for future eID solutions	
Report conducts research data-handling mechanisms	Report mirrors a toolbox of formal methods for modelling and verification of complex, composed protocols and their privacy features. The focus is on privacy-friendly audit and data handling mechanisms.	
D24.4	Third report on research on protocols and tools for future eID solutions	
3rd report on research on protocols and tools for FutureID solution	Not Completed	

Table 1: SP 2 – Methodology, summary result

7.2 SP 3 – Client Development

Summary	Comments	Status
D31.1	Requirement report	
This report will start with a detailed examination of the various standards for interface devices on PC-based and mobile platforms in order to provide a well-engineered specification of the requirement of the IFD Service of the FutureID client.	Report addresses IFD service and the existing standards like NFC, Open Mobile API, PC/SC, SICCT and ISO/IEC 24727, which can be used for the IFD service for the FutureID client.	
D31.2	Interface and module specification and documentation	
This report provides an interface and module specification of the IFD layer according ISO/IEC 24727-4. The design of the IFD services shall contain appropriate extension mechanisms, which allow integrating a variety of technologies to realize the identification of users.	IFD service provides an interface for communication to arbitrary devices and has a sophisticated architecture. IFD includes a common module, containing generic data structure, and provides convenience function. It is based on ISO/IEC 24727 and TR 03112-6.	
D31.3	Implementation of the IFD service for selected platform	
This report will implement the modular IFD layer as specified in D31.2 such that the selected smart cards and secure elements can be used by the eID service layer as specified and implanted in WP3.2.	Document describes the implementation of the IFD layer of the FutureID client, based on Java and supporting Windows, Linux, Mac and Android. IFD supports Java Smart Card I/O and the Transport API of the Open Mobil API and comprises an IFD Proxy.	
D31.4	Test report	
This report will integrate the work of task 31.3 into the FutureID client and provide input for the client test bad to evaluate the correctness of the solution.	Report describes a broad basis for the actual test implementation of WP37 and allows verifying of the IFD functionality with a high trust level. Tests were shown for representative test causes such as nPA/Germany and	

	eCard/Austria.	
D32.1	Survey and analysis of existing eID and credential systems	
This report will provide a detailed survey and analysis of existing eID and credential systems such that the different concepts can be smoothly integrated into a common FutureID architecture. This survey will provide a solid foundation for the forthcoming tasks in this work package and the entire FutureID project.	Report is a comprehensive overview of the current eID landscape in Europe. It shows the functionalities, the technical protocols, the infrastructures and the smart card concepts in 15 states.	
D32.2	Requirement report	
This report will start a detailed examination of the various standards, authentication protocols, credential formats and available components in order to provide a well-engineered specification of the requirements for the eID service of the FutureID client.	A list of concrete requirements from six different use cases for the FutureID client is given along a broad range of standards, such as CEN TS 15480, ISO/IEC 24727, Open Mobile API, SAML, OpenID and ISO 20022.	
D32.3	Interface and module specification for eID services	
This report will develop a specification of the interfaces and modules of the eID services of the FutureID client, which will be based on existing standards such as CEN 15480, ISO/IEC 24727 and SAML for example.	The report shows the framework to support credentials through CIFs, authentication protocol through SAL plug-ins, enhancements of the functionality through application plug-ins, authentication protocol through IFD plug-ins and enhancements of the functionality through application extensions.	
D32.4	Implementation of basic modules	

This report will implement the basic and generic eID service modules as specified in D32.3.	The eID services are a key component for the FutureID client. Service comprises an Event Manager which handles credential-inducted events and performs card recognition to determine the credential.	
D32.5	Implementation of protocol specific modules for selected protocols	
This report will implement protocol specific services for selected authentication protocols and credentials as specified in D32.3.	In this deliverable protocol-specific modules to support Extended Access Control, generic cryptographic operations, ABC4TRUST, IRMA, PIN management, eID activation, eID status inquiry, signing and PKCS #11 have been described.	
D32.6	Development of tool for efficient creation of CardInfo files	
This report will paint a tool specification, implementation and testing for the efficient creation of CardInfo files.	To create CardInfo structures according to CEN 15480 for a large variety of smart cards there need to be appropriate tools, which analyze the internal structure of the smart cards by sending an appropriate sequence of commands to the card and analyzing the responses to create the standardized CardInfo structures. This report displays a description.	
D32.7	Creation of CardInfo files for selected cards	
This report will produce CardInfo files for selected cards, which are or will be rolled out in large volume across Europe.	Not yet delivered	
D33.1	Requirement report	

This report will formulate the requirements for the eSign service module. This includes the coordination with the FutureID client framework and the detailed examination of the DDS protocol as well as the signature formats with regards to the to-be-defined module framework.	Report is focused on the functionality to sign digital documents, based on current and on future electronic signature standards in the frame of OASIS-DSS. This captures ETSI and CEN as well as the M/460 mandate.	
D33.2	Interface and module specification and documentation	
This report will define the interfaces of the eSign service module as such as well as the interface of the underlying framework.	Report provides a brief overview of interfaces of eSignature Service Module as such and also the interfaces of the underlying framework in the FutureID infrastructure.	
D33.3	Implementation of the work	
This report will show the implementation of the module framework, followed by the implementation of the OASIS-DSS protocol and the various signature formats.	In this document the design decision as well as the implementation aspects of the eSign services is reflected. This service is an add-on, conforming to add-on framework of the FutureID client.	
D33.4	Test report	
This report will integrate the work of Task 33.3 into the FutureID client and provide input for the testbed to evaluate the correctness of the solution.	The scope is on testing of the eSign services and their integration into the FutureID client. It addresses testing of single components, testing of a system with test vector to ensure the successful integration and final, the requirements of D33.1 were reviewed.	
D33.5	Provision of selected evaluation documents	

This report will provide selected documents, which are required for a formal evaluation and certification of an electronic signature component as suggested for required by the national implementations of the European Directive on electronic signature.	Not delivered yet	
D33.6	Legal analysis of eSignature services	
This report produces a comprehensive analysis of the legal framework surrounding the provisioning of FutureID eSignature services.	Not delivered yet	
D34.1	Requirements analysis	
This report will transfer the functional and technical requirements derived from the use cases and scenarios into user interface requirements.	This document contains several analytical aspects of the FutureID client user interface. A user-task matrix, where the analytical components are joined and a conclusion is drawn. This is the basis for the prototypes.	
D34.2	Design mockups	
This report will focus on the core user interface concept of the FutureID client developed in SP3.	The mockup for three client devices, desktop, tablet and mobile along common use cases and user tasks of the FutureID client is painted. This would be the basis for user evaluation tasks.	
D34.3	Analysis of mockups	
This report provides a usability analysis of the client UI design mock-up.	Paper shows the results of a usability evaluation of the FutureID client prototypes for desktop, tablet and mobile versions. The focus is on easy understandable and easy use.	
D34.4	Technical specification of UI	

This report will develop a technical design of user interface of the FutureID client.	Report discusses the technical details of user interface design for the FutureID Client. It identifies which platforms will be supported by the client. It outlines the specific characteristics of those platforms.	
D34.5	Implementation of FutureID client user interface	
This report will implement the basic client user interface as specified in D34.4.	Document reflects the implementation related aspect of the FutureID client user interface. The focus is on dynamic HTML design and on a remote procedure called GUI module.	
D34.6	Integration test	
This report will ensure the conceptually correct implementation of the user interface in the technical implementation of the client.	Report contains a technical and usability analysis for the FutureID graphical user interface client. It is based on three pillars: a) unit tests, b) system tests and c) a usability walkthrough.	
D35.1	Requirement report	
This report will investigate existing eID and credential systems and their trustworthy properties and specify necessary requirements for both mobile and PC based trustworthy eID platforms.	This report discusses the relevant aspects of a trustworthy client platform and develops the requirements for FutureID. The focus is on the level of trust and security. Five levels were shown.	
D35.2	System architecture trustworthy client platform	
This report will develop architecture as well as protocols for real-time trust measurements and trust establishment in heterogeneous dynamic environments.	System architecture for a trustworthy client platform is displayed, based on various methods and elements for secure execution environments and their access control, with the focus on stationary and mobile devices.	

D35.3	Implementation of security relevant modules	
This report will implement the security relevant modules for selected platform.	Not delivered yet	
D35.4	Secure back-up and recovery mechanisms	
This report will develop protocols for secure back-up and for recovery mechanisms.	This report develops protocols that can be applied to existing credential technologies such as Idemix, U-Prove or alternative credential technologies and allows the legitimate user to prepare for the loss of data/services such that he can securely retrieve/restore his credentials if necessary.	
D35.5	Evaluation and integration	
This report will paint the proof of concept for the design specifications as well as for the implementation of the components specific for the integration.	Not delivered yet	
D36.1	Requirement report	
This report will first provide an overview of different proven concepts and interfaces for browser integration and discuss pros and cons of the various techniques.	This report provides a link between the FutureID client and programs running inside the browser, to allow service providers to integrate the advanced authentication and identification mechanisms of the FutureID into their web application.	
D36.2	Interface and module specification and documentation	

This report will provide specifications for the modules required to adapt these interfaces to the different browser-environments.	The architecture and the interface specification are painted to implement a flexible and platform independent communication link between the FutureID client and the browser. It supports arbitrary authentication protocols and credentials.	
D36.3	Implementation	
This report will implement the framework for browser integration, including interfaces and browser-specific modules.	This document illustrates a solution of the whitelist-filter to restrict access from websites and the internationalization of Http-responses.	
D36.4	Integration / Testing	
This report will integrate the other modules of the FutureID client into the framework developed in task 36.3 and perform operational tests in order to validate the solution.	Not delivered yet	
D37.1	Requirement report	
This report specifies the detail requirements for the client testbed.	Based on the W-model an early automated and central testing is displayed.	
D37.2	Test strategy	
This report will define an overall test strategy and an appropriate set of test toolkits to cover the various platforms supported by the FutureID client.	This report concentrates on setting up a suitable testbed with different kinds of test tools, which are all integrated into the central continuous integration tool Jenkins.	
D37.3	Specification and implementation of tools for module tests	

This report will gather all unit and module test cases from the other implementation-related work packages including documentation in order to build.	Not delivered yet	
D37.4	Specification and implementation of tools for integration tests	
This report will continuously gather and publish a reference implementation of the different FutureID components as a test framework for implemented modules.	Not delivered yet	
D37.5	Specification and implementation of tools for acceptance tests	
The report will specify individual acceptance criteria for the FutureID client and provide corresponding test tools.	Not delivered yet	

Table 2: SP 3 – Client development, summary result

7.3 SP 4 – Backend integration

Summary	Comments	Status
D41.1	Requirement report	
This report will start with a detailed examination of the relevant standards for federated identity management and the interfaces for the existing and emerging authentication and identity services as identified in the inventory.	This document specifies the requirements for the identity broker within the FutureID infrastructure.	
D41.2	Interface and module specification and documentation	
This report will develop a specification of the interfaces and modules of the identity broker which will integrate existing identity and authentication services and support	Document contains the specification of the interfaces and modules of the broker service, which will integrate existing and emerging identity and authentication	

internationally acknowledged identity federation standards such as SAML and WS-Trust.	services and support the internationally acknowledged identity federation standard.	
D41.3	Implementation of the identity broker in dispatcher mode	
This report will implement the identity broker in dispatcher mode as specified in D32.3.	Report describes the implementation of the broker service in dispatcher mode, based on the integration with and without the federation service and the description of the interfaces between the broker core and the authentication backend.	
D41.4	Implementation of the identity broker in claims transformer mode	
This report will implement the identity broker in claims transformer mode as specified in D32.3.	This report provides details about the implementation of the identity broker in claims-transformer mode. Based on updated URLs a constant update is available in wiki.	
D41.6	Legal analysis of the identity broker	
This report will produce a comprehensive analysis of the legal framework surrounding the provisioning of identity broker services, with particular attention to issues of data protection and intermediary liability.	Not delivered yet	
D42.1	Requirements	
This report will start with a detailed analysis of the authentication procedures supported by the relevant eID tokens identified in the technical inventory D21.2.	Report contains the specific requirements for the UAS and the APS language which is to be interpreted by the UAS. This is a preliminary document.	
D42.2	Interface and module specification and documentation	

This report will provide a modular design of the universal authentication service starting at the architectural overview.	The job execution environment is the core component of the UAS and is capable of running APS-specified authentication protocols. Two interfaces were shown, the universal service interface and the authentication interface.	
D42.3	APS definition	
This report will define domain specific language for authentication protocol specifications (APS).	The document provides the design of APS language	
D42.4	Specification of basic services	
This report will specify the basic services which are required for the execution of authentication protocols specified in APS files.	This document provides a specification of the Basic Services as a component of the Universal Authentication Service (UAS). The second component of the UAS, that the Basic Services interact with, is the Job Execution Environment (JEE). The deliverable deals with the requirements analysis and specification of the functions invoked by the Job Execution Environment and executed by the Basic Services.	
D42.5	Implementation of basic services	
This report will provide a specification and implementation of a generic execution environment, which is capable of executing arbitrary authentication protocols, which are described by APS-files, which refer to basic services.	Basic services focus on providing functions that enable the usage of common security algorithms and protocols, enable the transparent usage of HSMs and generate APDUs for smart cards.	
D42.6	Specification of execution environment	
This report will first select appropriate authentication protocols, which are supported by the FutureID client and specified in the APS language to be	This report captures JEE and CAPS. It is recommended to do the implementation of the JEE according to the specification.	

executed by the universal authentication service.		
D42.7	Implementation of execution environment	
This report will integrate the different parts of the universal authentication service and will perform appropriate tests to guarantee the required level of quality as specified in D42.1.	Document shows execution of scripts and CAPS packages.	
D42.8	APS files for selected authentication protocols	
This report will provide the APS files that are used for validating the APS language as well as first versions of the execution environment.	Report presents the APS files for some selected protocols that are highly relevant to the eID systems in general and FutureID in particular. The focus is on EAC, PACE, TLS and ISO 9798-4.	
D42.9	Testing report	
This report will contain the results of the operational tests, in which the various authentication protocols described by APS-files are tested in conjunction with the execution environment.	Not delivered yet	
D43.1	Analysis of trust aspects	
This report will provide a solid analysis of the different aspects of trust, which are relevant for the identity broker, the associated trust services and the entire FutureID infrastructure.	This report starts with the meaning of trust followed by an examination of trust in identification, authentication and non-repudiation mechanisms. The report refers to STORK QAA and NIST.	
D43.2	Requirements analysis	
This report will examine existing sources of trust and trust status information and identify the gap between provided and required information.	Document gives a brief overview of relevant standards and infrastructures for trust services and specifies requirements of the trust service of the FutureID infrastructure.	

D43.3	Interface and module specification	
This report will define and document the interfaces of the trust services module with the FutureID infrastructure.	An overview of the trust infrastructure and the trust service architecture in FutureID is displayed.	
D43.4	Implementation	
This report will at first implement the framework of the trust services module and the validation service, followed by the implementation of a set of collects for the supported sources of trust status information.	Document describes the implementation work done in WP43. This captures delegated signature validation, TSL integration and the trust status provisioning via DNSSEC.	
D43.5	Test report	
This report will integrate the trust services module into the FutureID backend framework and performs operational tests in order to validate the solution.	An overview is given of the integration of trust services into the FutureID backend, based on the description of test vectors.	
D44.1	Description of EIAs and their requirements of integration	
This report will review current industrial EAI infrastructures as well as their need for federated identity management.	The study provides a vision of the EAIs, the uses, the technologies and protocols as well as the type of EAIs. It also provides the study of the integration of Federated Identity Management (FIM) into EAI infrastructures and a description of the inclusion of EAI in FutureID infrastructure	
D44.2	Application integration service requirements	
This report will elicit relevant requirements for integration of applications into the FutureID infrastructure, including elements of commercial SaaS platform and considering too requirements from cloud-	Report shows a breakdown of requirements for the AIS of FutureID, from different perspectives. This is the input for D44.3 and D44.4.	

based identity and attribute management as a service functionality across domains and borders.		
D44.3	Technical specification for the application integration services	
This report will develop a technical specification for the application integration service.	Deliverable display based on the technical specification and address recommendations for the integration of AIS with the epSOS pilot in FutureID. The document also maps the requirements included in documents D22.x concerning AIS witih requirements with the AIS component which will perform the functionality. It also presents FutureID AIS Architecture: a detailed description of every AIS module containing the functionality and operational environment of all them was described.	
D44.4	Implementation of application integration services	
This report will implement the application integration service based on the specification from task 44.3.	This deliverable describes The implementation of JBoss specific Application Integration Service is oriented towards the epSOS pilot integration, which runs on JBoss server. In this way this Application Integration Service is covering the market for Java applications servers. The document describes implementation of the three different components of the Application Integration Service: • The Access Filter and the FutureID client detection module; • The Simple Credential	

	Transformer; • The Session Library. This deliverable also provides the environment used for the implementation and some guides for deployment and configuration of the different Application Integration Service building blocks.	
D44.5	Integration / Testing	
This report will integrate the application integration service implemented in task 44.4 as a module into the FutureID backend framework and perform operational tests in order to validate the solution.	Not delivered yet	
D44.6	Integration of selected application	
This report will integrate exemplary applications into the FutureID infrastructure using the application integration service implemented in task 44.3.	Not delivered yet	
D45.1	Requirement report	
This report will provide a summary of the most relevant testing requirements.	Report refers to different types of requirements for the server testbed. General requirements address the testing; specific requirements capture security, automated vulnerability testing, server communication and test environment requirements.	
D45.2	Test environment	
This report provides the test environment and corresponding test vectors that ensure that all given conditions are met by the various servers.	The outline of the reference test environment for the FutureID server infrastructure is painted; based on a toolset for testing the functionality and communication between the FutureID	

	components.	
D45.3	Server testbed	
This report will provides a first version of the online server test bed for testing purposes in other work packages and sub-projects.	Deliverable gives an overview of the tools, which are available for server testing of the backend component.	
D45.4	Test assertion	
This report will provide the derived assertions from the produced and referenced specifications.	Report gives valuable directions to any interested partner on how to use the server testbed, based on the implanting in SoapUI.	
D45.5	Transformation tools to present test results	
This report will provide the tools that will be used to transfer test results into the presentation format.	Report mirrors some insights into test result presentation within FutureID, based on XSLT engines.	
D45.6	Online server testbed	
This report will provide the final release of the server testbed as a public web service.	Not delivered yet	

Table 3: SP 4 – Backend integration, summary result

7.4 SP 5 – Pilot application

Summary	Comments	Status
D51.1	Requirements and inventory report	
One target of FutureID is the bridging of the FutureID infrastructure with the ICT LSP epSOS Infrastructure. This report provides a compact inventory of the relevant documents and components, which are relevant for the integration of FutureID into the existing epSOS infrastructure.	Report addresses an extended security safeguard along the large scale pilot epSOS which FutureID infrastructure should support. It's based on two pillars a) direct password-based encryption and message authentication mode ad b) PACE-based key exchange mode.	
D51.2	Technical module and interface specification	
Report reflects the cross work of FutureID and epSOS, the EC project for a pan-European health data exchange.	This document defines technical modules and interface specifications for epSOS use cases that demonstrate the capabilities of FutureID technology. Four use cases complete existing epSOS functionalities and are defined in the report. This deliverable defines necessary technical modules and interfaces that accomplish these use cases.	
D51.3	Citizen service implementation	
This prototype will provide a proof of concept implementation for the selected use case. The deliverable will include the components specification and the development environment.	Not delivered yet	
D51.4	Technical Test Report	
This report will contain the results of the operational tests in order to validate the solution.	Not delivered yet	
D51.5	Legal aspects and evaluation of	

	implemented citizen services	
This report will identify and resolve potential legal issues resulting from the applications scenarios. This will in particular involve the assessment of selected privacy aspects and the drafting of the consent forms and privacy policies.	Not delivered yet	
D52.1	Requirements for FutureID components in Business Scenarios	
Considering the integration of Future-ID with Atos e-Learning Services for Enterprises as the main objective, this initial report is very important to show the abstract technical blueprint of Atos e-Learning Services for Enterprises to the technical work packages. Especially important is the requirements description for identity management in order to establish the integration with Future ID components specified in other work packages.	The report describes the Atos e-Learning services for enterprises as a business scenario. It describes the requirements for identity management in the context of Atos e-Learning Services for Enterprises, and business environments, with special focus on business viability. It shows the integration of Atos e-Learning Services for Enterprises and establishes the basic integration architecture.	
D52.2	Technical Specification including Description of IdP / SP and Identity Token Formats	
The main objective of this deliverable is to specify the interfaces and modules for the integration of Atos e-Learning Services for Enterprises into the FutureID infrastructure.	The report reflects the technical specifications of Apache specific AIS implementation and the Atos e-Learning integration interface as well as the basic AIS architecture description, explaining the components and both their functions and their connection with FutureID elements. It shows the Atos e-Learning architecture and the breakdown components. It also explains how the communications established between the AIS components and the FutureID elements takes place. It describes the	

	used technologies and the supported multimedia contents by Atos e-Learning.	
D52.3	Proof-of-Concept Implementation of a Hosted Service for the FutureID Framework	
This prototype will provide the technical testbed landscape and relevant example service implementations within e-Learning Services for Enterprises and will demonstrate how to deploy and consume FutureID identity services in this context. The deliverable will include the components specification and the development environment.	Not delivered yet	
D52.4	Technical Test Report	
This report will provide the results of the technical tests in order to validate the solution integrating the e-Learning Services for Enterprises platform with the FutureID backend framework.	Not delivered yet	
D52.5	Legal aspects and evaluation of business scenarios	
This report will identify and resolve potential legal issues resulting from the applications scenarios. This will in particular involve the assessment of selected privacy aspects and the drafting the consent forms and privacy policies.	Not delivered yet	

Table 4: SP 5 – Pilot application, summary result

In chapter 9 – Appendix – a complete picture of all deliverables in a traffic light mode is shown.

8. Conclusion

FutureID, started on 1st of November 2012 looks after T=33 months in a good shape. The consortium under the project coordination of Fraunhofer IAO has deployed a homepage (www.futureid.eu), a livelink for documents and deliverables and a wiki as a workbench for all activities. After 33 months runtime in total 72 reports plus two project management reports were created, collected and submitted. A major blocking point was not identified. The project progress is almost in the milestone plan of the DoW.

As final results, the FutureID project will mainly provide a standardized, trustworthy and ubiquitously usable eID client, as well as a usable identity management infrastructure. The FutureID project is developing two pilot applications by integrating FutureID into the European eHealth Project epSOS2 and into the Atos e-Learning Services for Enterprises. This demonstrates the usability of FutureID for providing identity management to real Service Providers.

Besides this, FutureID will contribute to standardization efforts and will develop a new set of protocols for future eID solutions.

The FutureID infrastructure will provide benefits to all stakeholders involved in the eID chain including users, service providers, e-government businesses and identity service providers. FutureID eID client will benefit the users to access different services supported by FutureID infrastructure. Application Integration Services will provide easy integration of existing services into the FutureID infrastructure avoiding making large up-front investments in eID technologies to service providers. The Identity providers will benefit from the increasing set of potential customers of their services.

9. Appendix

SP 2				SP 3							SP 4					SP 5	
21.1	22.1	23.1	24.1	31.1	32.1	33.1	34.1	35.1	36.1	37.1	41.1	42.1	43.1	44.1	45.1	51.1	52.1
21.2	22.2	23.2	24.2	31.2	32.2	33.2	34.2	35.2	36.2	37.2	41.2	42.2	43.2	44.2	45.2	51.2	52.2
21.3	22.3	23.3	24.3	31.3	32.3	33.3	34.3	35.3	36.3	37.3	41.3	42.3	43.3	44.3	45.3	51.3	52.3
21.4	22.4		24.4	31.4	32.4	33.4	34.4	35.4	36.4	37.4	41.4	42.4	43.4	44.4	45.4	51.4	52.4
21.5	22.5				32.5	33.5	34.5	35.5		37.5	41.6	42.5	43.5	44.5	45.5	51.5	52.5
	22.6				32.6	33.6	34.6					42.6		44.6	45.6		
	22.7				32.7							42.7					
												42.8					
												42.9					

Table 5: Complete overview