E-vote 2011

SSA-U Appendix 4

Project and progress plan

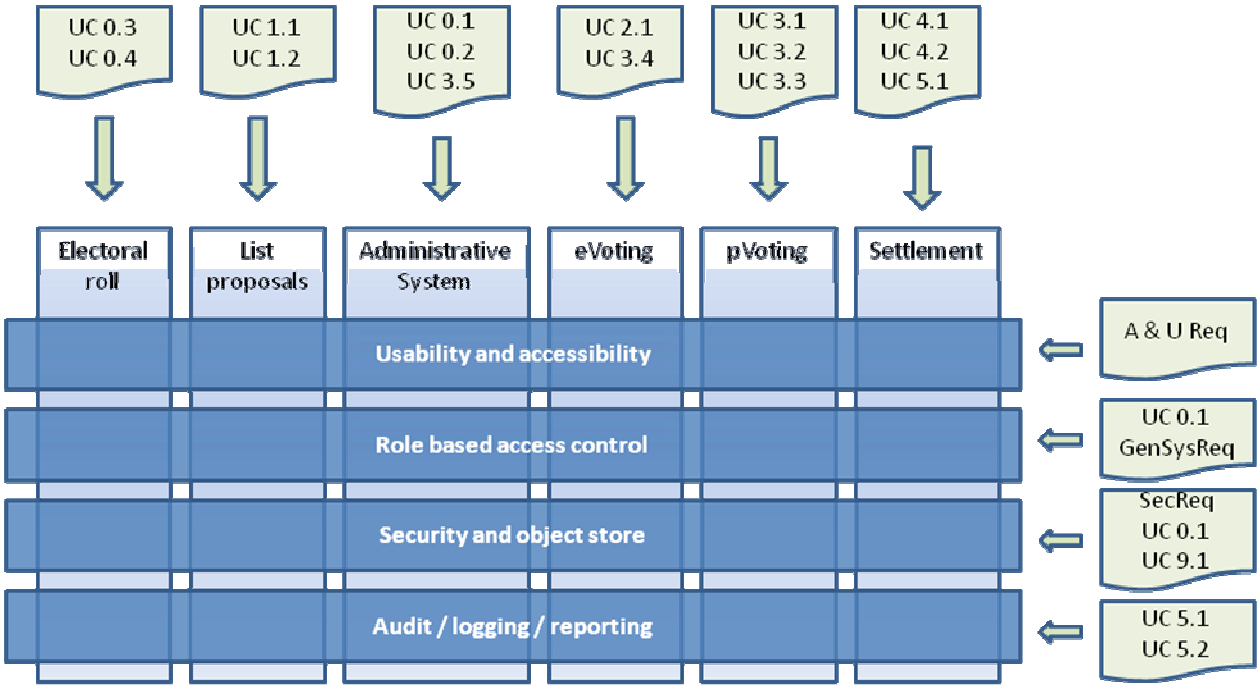
Project: E-vote 2011

CONTENT

1. Implementation methodology and customer involvement

1.1. System components and requirements from KRD

The drawing below shows the general concept of the system that is to be developed and how the different requirements apply to different parts of the system.



As shown in the drawing, the system is constructed with horizontal layers of basic functionality which the vertical functional groups of end-user functionality are based upon. The drawing also indicates where the different parts of the requirements provided by KRD apply.

The system will be delivered in iterations. This is because the vertical subsystem contains horizontal layer components which will come in new versions throughout the project as the development of the horizontal layers progress. However, the functionality as seen from the end-user can remain the same throughout the iterations. Obviously, there will also be iterations based on the results from usability and accessibility test and enhancements proposed from those tests.

Based on our experience in developing these types of systems, ErgoGroup/Scytl propose our development methodology which is based on an iterative development process. This development methodology has several advantages:

- The development process is made transparent to the customer (i.e. KRD)
- End user functionality may be verified by the customer (i.e. KRD) at early stages in the development process.
- The actual progress may be easily monitored.
- The quality of the modules developed are automatically verified at every iteration
- Each iteration will assure that the total system converge to fulfill all vertical (functional) requirements as well as all horizontal (general) requirements in the final release.

1.2. Partial deliveries and Intermediate Releases

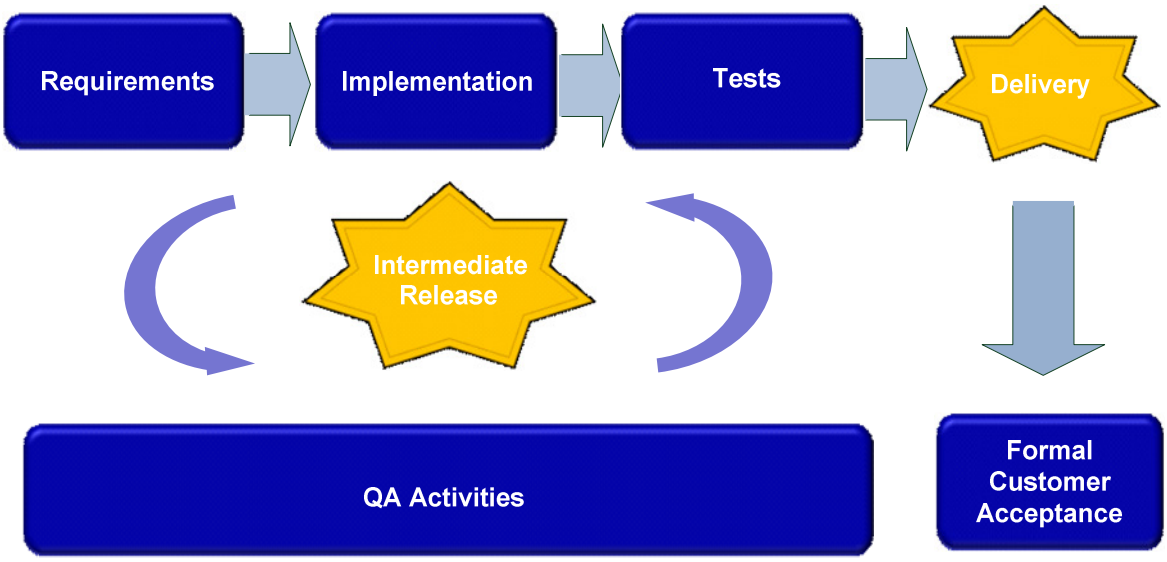
KRD has requested partial deliveries that should be formally handled according to SSA-U section 2.2 – 2.5. This will require a complete and final subsystem to be delivered including both the vertical components and a final version of each horizontal component described in section 3.1.

We are achieving this by combining two approaches:

- Proposing three partial deliveries as outlined in section 3 of this Appendix
- Using and implementation methodology that allows for and encourages a much broader degree of customer and user involvement

Combining these techniques we have three formal deliveries of the voting system formed by multiple subsystems ready for piloting or evaluation. At the same time the customer can access partial iterations of the subsystems forming the final delivery. The delivery in each iteration will be available for individual testing and evaluation.

The drawing below shows our iterative development methodology with intermediate releases. There are several intermediate releases within a formal delivery.



Intermediate Releases support the concept of iterative development where system development is time boxed in iterations of agreed length in time, usually around 4 weeks. The specific length will be agreed and specified in the project plan during the project initiation.

Intermediate releases will contain two types of components:

(1) Components internal to the Election system that will evolve throughout the project (horizontal components in the drawing in section 2.1)
(2) Components that provide end user functionality that can be completed and formally accepted at the end of the iteration (vertical components in the drawing in section 2.1)

Components in (1) may not be testable for the customer at the end of the iteration. This will usually be intrinsic functionality, for instance within the auditing and security mechanisms. These components may need frequent changes throughout the project and should not be placed under formal acceptance and change control until a final stage.

Components in (2) are testable by the Customer and can be formally accepted and placed under change control with regards to end-user functionality. However, even these components may be object to iterations as part of the accessibility and usability testing and progress.

This development methodology may be combined with a formal regime for customer acceptance with regard to end-user functionality on components in (2). This ensures that the Customer will be guaranteed that the functionality will be provided as agreed on, and that functionality will not change unless a formal change order is approved by the Customer.

1.3. Method of customer involvement in the development process

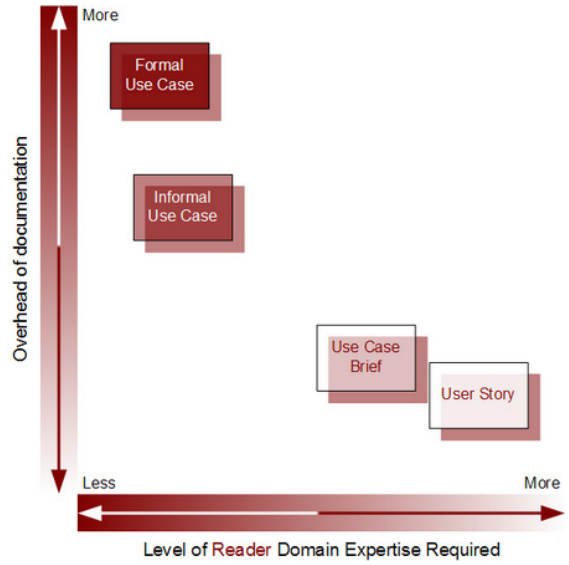
The actual content of each intermediate release is defined in the planning of these releases. This is done in cooperation between the Customer and the Contractor. Typically, this planning also involves assigning end-user functionality to each intermediate release. However, to demonstrate end-user functionality within the scope of an intermediate release, end user functionality must be broken down to atomic functional requirements that may be verified independently (please refer section 2.4 User Stories).

Thus, each iteration consists of the following three phases:

1. The planning phase were atomic pieces of end-user functionality is assigned to the intermediate release. Also new iterations of already delivered end-user functionality may be assigned to an intermediate release.
2. The development phase which ends with an intermediate release (usually every 4th week). This intermediate release includes our Quality Assurance methodology outlined in Appendix 5 to the SSA-U contract, including automatic regression tests.
3. The functional acceptance test
    a. Customer will (only) be able to test user functionality and not intrinsic components. These components are agreed upon in the planning phase described above.
    b. Formally accepted functional components are placed under change control thereafter.

1.4. User Stories

The KRD requirements describe the functionality that shall be delivered in a formal manner. The final system acceptance test will be based on these requirements. However, several of the requirements (and use cases) are dealing with vertical end user functionality as well as horizontal basic functionality (refer the drawing in section 2.1).



[1]

The drawing to the left shows the common approach to user stories and use cases.

Use Cases require more formal documentation and cover all aspects of the functionality including context, technical environment and other.

User Stories describe perceived end user functionality. Usually, user stories require more domain expertise from the developing organization.

Often we base our dialogue with the customer on creating user stories. Planning of iteration in our development cycle will involve assigning new user stories to an iteration.

[1] TynerBlain

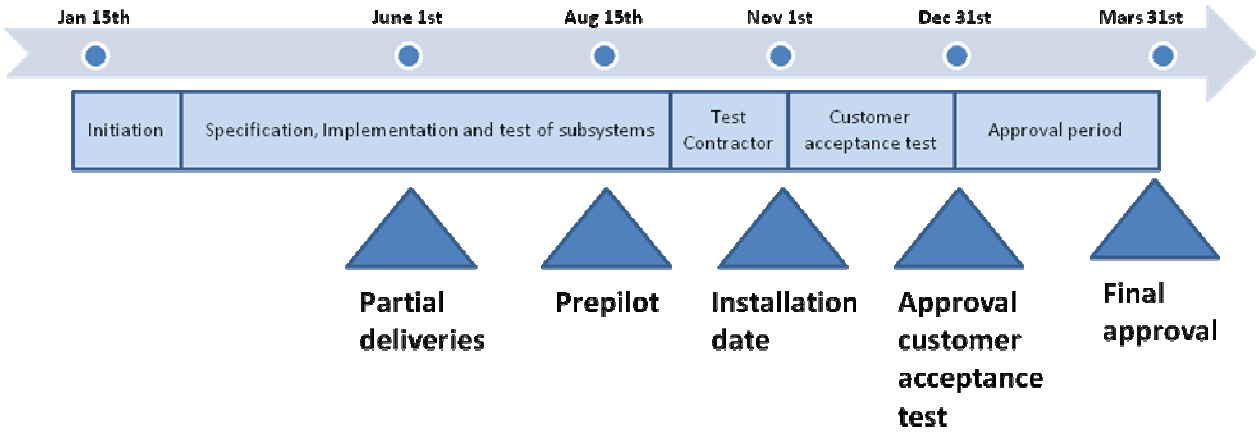
User stories are always expressed in the form:

As a [**person in a role**] I want to [**perform some activity**] so that [**some goal is achieved**].

User stories shall always be broken down into an atomic form. Thus, if it is possible after verification to say that a user story is partially demonstrated, it has not been broken down into sufficient detail.

When the functional acceptance test after each intermediate release has been performed, or more precisely when the end user functionality described in the agreed user stories has been verified, the end user functionality is placed under change control.

2. Formal Project progress plan and milestones



The project plan is made with the assumption that contract is signed December 15th. 2009. The installation date is November15 th.2010 with approval of customer acceptance test Dec 31st.2010. There is no need for conversion of data in order to meet the requirements.

Milestones

ID	Finish date	Milestone	Deliveries
MP1	15/12-09	Contract signed	
MP2	29/1-10	Detailed project plan finished	• Detailed project plan • Risk management plan • Quality management plan
MP3	26/02-10	Design documents finished	• Customer approved system design document

			• Customer approved design templates
MP4	09/04-10	Intermediate release-1	Evoting: • Use case 2.1 accessibility and usability, multiple languages and selected browsers <u>Adm.system</u> • Use case 0.1 Role based access control – database and service interface functionality • Use case 0.3 Electoral roll <u>Pvoting:</u> • Use case 3.3 Electronic counting – initial functionality
MP5	30/04-10	Intermediate release-2	Evoting: • Use case 2.1 secure voting protocol, storing of votes and multiple browser support <u>Adm.system</u> • Use case 0.1 Role based access control – management application (user interface) • Use case 0.2 Configuration – create and store configurations, selected configuration options available • Use case 0.4 Exception process ER • Use case 1.1 Submission list proposals - user interface • Use case 9.1 Authentication – Permanent credentials <u>Pvoting:</u> • Use case 3.1 Reg. of pvotes in ER • Use case 3.3 Electronic counting – main functionality
MP6	28/05-10	Intermediate release-3	Evoting: • Use case 2.1 Improvements evoting client, integration of electoral roll • Use case 3.4 Counting evotes – initial functionality cleansing and mixing <u>Adm.system</u> • Use case 0.2 Configuration – detailed configurations and approvals • Use case 1.1 List proposals - approvals, notifications and publishing • Use case 1.2 Processing list proposals – presentation and approvals • Use case 4.2 Settlement – Merging of votes • Use case 9.1 Authentication <u>Pvoting:</u> • Use case 3.2 Man.reg. of vote results – reg. of

			results • Use case 3.3 Electronic counting – final improvements
MP7	01/06-10	Partial delivery electoral roll and electronic counting pvotes	Delivery of use case 0.3 electoral roll and use case 3.1 Registration of pvotes in electoral roll. Delivery of use case 3.3 electronic counting pvotes. Ready for part acceptance test.
MP8	25/06-10	Intermediate release-4	Evoting: • Use case 3.4 Counting evotes – implementation of counting process with audit support <u>Adm.system</u> • Use case 0.1 Role based access control • Use case 1.1 Submission list proposals • Use case 3.2 Man.reg. of vote results – reg. data for election protocol • Use case 3.5 Approval pvotes, ballots – ER mark off and verification • Use case 4.2 Settlement – Merging of votes • Use case 5.1 Reporting – select and execute reports • Use case 5.2 Auditing – full logging functionality, create monitors
MP9	23/07-10	Intermediate release-5	Evoting: • Use case 3.4 Counting evotes <u>Adm.system</u> • Use case 0.2 Configuration • Use case 1.2 Processing list proposals • Use case 3.2 Man.reg. of vote results • Use case 3.5 Approval pvotes, ballots • Use case 4.1 Reporting to SSB • Use case 4.2 Settlement • Use case 5.1 Reporting • Use case 5.2 Auditing
MP10	15/08-10	Prepilot ready for customer acceptance test.	Prepilot with functionality according to section 2.4, ready for part acceptance test.
MP11	01/11-10	Installation date	Installation date reached. Customer acceptance test starts.
MP12	31/12-10	Approval customer acceptance test	Approval reached.
MP13	01/01-11	Ready for approval period	Approval period starts.
MP14	01/03-11	System configured	System configured and ready for use by local communities.
MP15	31/03-11	Delivery date	End of approval period
MP16	01/04-11	Warranty period	Warranty period starts
MP17	01/04-12	End of warranty period	

2.1. Customers allocation of resources

The customer is expected to allocate resources to fulfill their responsibility according to this contract, e.g. project management, test and approval. In addition to the general fulfillment of contract, the contractor estimates the following minimum direct involvement from customer side.

ID	Finish date	Milestone	Allocation of resources
MP1	15/12-09	Contract signed	
MP2	29/1-10	Detailed project plan finished	**Establish project procedures: 20 man days**
MP3	26/02-10	Design documents finished	**Approval of design documents: 20 man days**
MP4	09/04-10	Intermediate release-1	**Approval of specifications: 10 man days** **Cooperation and approval of usability and accessibility: 10 man days**
MP5	30/04-10	Intermediate release-2	**Approval of specifications: 10 man days** **Cooperation and approval of usability and accessibility: 10 man days**
MP6	28/05-10	Intermediate release-3	**Approval of specifications: 10 man days** **Cooperation and approval of usability and accessibility: 10 man days**
MP7	01/06-10	Partial delivery electoral roll and electronic counting pvotes	**Approval of specifications: 25 man days** **Cooperation and approval of usability and accessibility: 25 man days**
MP8	25/06-10	Intermediate release-4	**Approval of specifications: 10 man days** **Cooperation and approval of usability and accessibility: 10 man days**
MP9	23/07-10	Intermediate release-5	**Approval of specifications: 10 man days** **Cooperation and approval of usability and accessibility: 10 man days**
MP10	15/08-10	Prepilot ready for customer acceptance test.	**Approval of specifications:20 man days** **Cooperation and approval of usability and accessibility: 20 man days**
MP11	01/11-10	Installation date	**Approval of specifications:20 man days** **Cooperation and approval of usability and accessibility: 20 man days**

MP12	31/12-10	Approval customer acceptance test	**Acceptance test: 100 man days**
MP13	01/01-11	Ready for approval period	
MP14	01/03-11	System configured	**Election configuration: 10 man days**
MP15	31/03-11	Delivery date	
MP16	01/04-11	Warranty period	
MP17	01/04-12	End of warranty period	

The customer is expected to be responsible for the following:

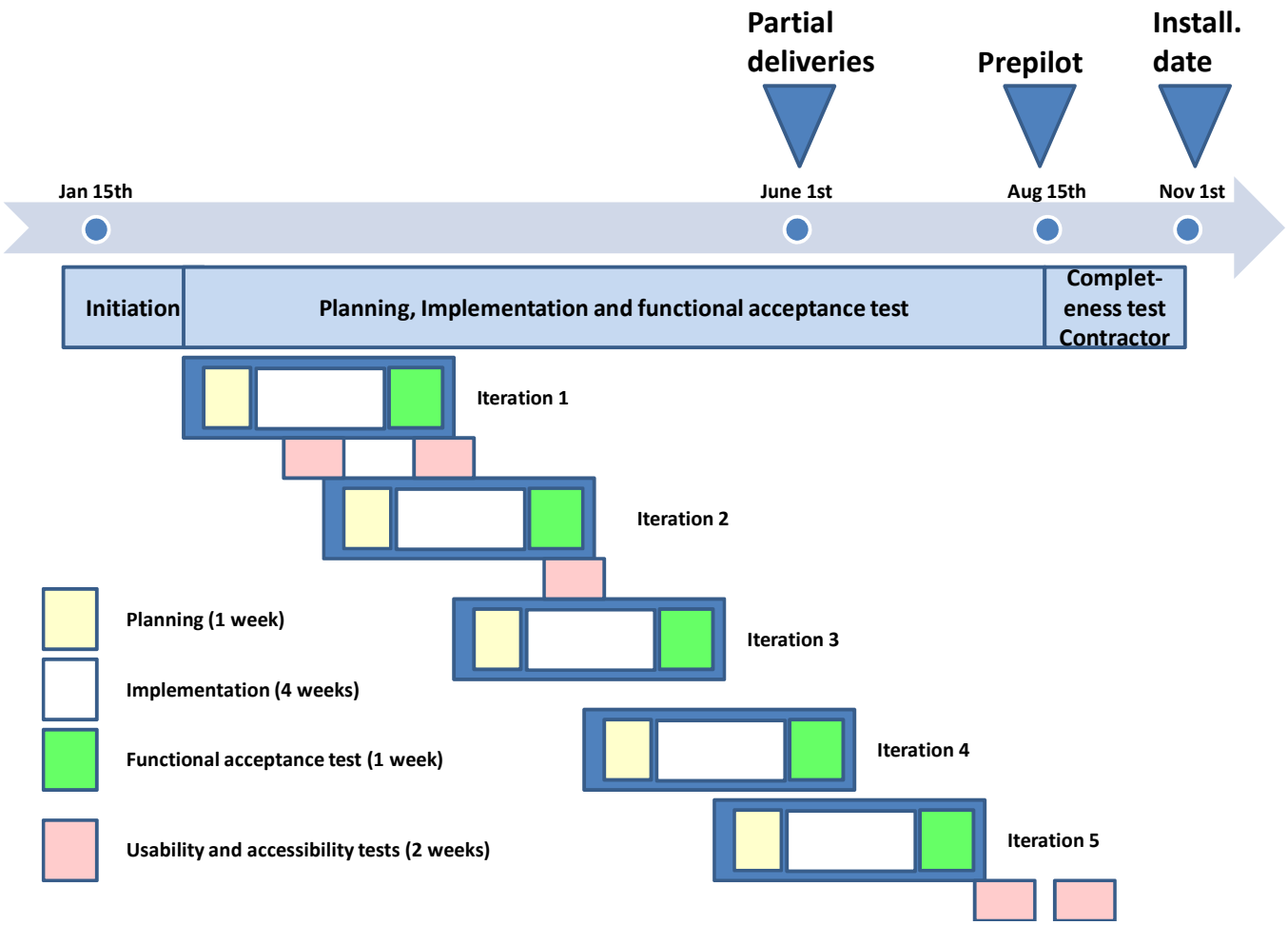
- To provide test data for all necessary test activities, herein any costs related to this responsibility.
- Approval of specifications on other documents that need to be confirmed by the customer.
- To keep the contractor informed of any deviation in associated projects or processes that may influence the project progress for this contract.
- To provide necessary public information.
- Handling of media and public officials.

2.2. Plan customer and user involvement

Customer involvement in all iterations is essential to ensure that requirements are implemented according to customer expectations. As already explained, each iteration consists of three phases:

- The planning and detailed specification phase
- The development phase which output is an intermediate release
- The functional acceptance test, ending the iteration

The drawing below illustrates customer involvement in each iteration of the development process. The customer is involved in the planning phase and the functional acceptance test phase. In addition the customer will be involved in the usability and accessibility tests.



The planning phase will initiate every iteration. The activities in this phase will continue for approximately one week and involve both parties. The quality of cooperation is essential to make necessary progress in each iteration. The result of the planning phase is the chosen user stories to be implemented in the intermediate release.

The functional acceptance test phase ends every iteration. The objective is to test and accept the implemented functionality. The test activities will continue for approximately one week. The customer may formally accept functional components based on their own tests. The contractor will facilitate the test environment for the functional acceptance tests.

2.1. Liquidated damages in the case of delay, refer main contract body section 11.5.2

Liquidated damages in the case of delay, apply to the milestones with id MP10 and MP11.

2.2. Commissioning

The duties of the parties in connection with commissioning are specified in Appendix 7.

2.3. Functionality for pre pilot, milestone MP10

The pre pilot will include functionality to conduct a limited referendum. The following functionality should be available for referendum to the extent necessary for referendums:

- Use case 0.2 Configuration of the election system
- Use case 0.3 Electoral roll
- Use case 2.1 E-voting. The cryptographic protocol for vote verifications may not be included, if major changes to the protocol are introduced
- Use case 3.1 Registration of p-votes in electoral roll
- Use case 3.2 Manual registration of p-vote results
- Use case 3.3 Electronic counting of p-votes
- Use case 3.4 Counting e-votes

3. Quality control

3.1. Elaboration of requirement ST4.1

The contractor is fully certified according to the ISO-9001:2000 quality standard. This also applies to our Application Development and Application Management Services. (Please refer certificate below).

KOPI

DNV

DET NORSKE VERITAS

MANAGEMENT SYSTEM CERTIFICATE

Sertifikat nr. 2001-OSL-AQ-7041

Med dette sertifiseres at
KVALITETSSYSTEMET
ved

ErgoGroup AS

adresse:
Nydalsveien 28, 0402 Oslo

er funnet å være i overensstemmelse med følgende standard for kvalitetssystem
NS-EN ISO 9001 : 2000

Dette sertifikatet er gyldig for følgende produkter eller tjenester:

Tjenesteutvikling.
Salg og markedsføring.
Konsulenttjenester, rådgivning og prosjektledelse.
Utvikling, implementering, drift og forvaltning av IT-løsninger.

Første sertifikat gyldig fra:
1994-10-19

Sertifikatet med vedlegg er gyldig til:
2009-12-03

Jan Roald Brembo
Revisjonsleder

NORWEGIAN ACCREDITATION
QUAL 002

Sted og dato:
Høvik, 2007-01-12

For akkreditert enhet:
Det Norske Veritas Certification AS

Jøran Laukholm
Ledelsens representant

Brudd på forutsetningene for sertifikatet, slik det fremgår av vedlegget, kan gjøre sertifikatet ugyldig.

DET NORSKE VERITAS CERTIFICATION AS, Veritasveien 1, N-1322 Høvik, Tel.: +47 67 57 99 00, Fax: +47 67 57 99 11
AQN-23no 2005.03.03

The contractors Application management Service is built on ITILs Application Management framework. This ensures proper management of the applications through the complete lifecycle. The contractor performs professional AM-services on applications developed by ourselves as well as applications developed by other companies.

The subcontractor Scytl is ISO9001:2000 certified, following the ISO/IEC 90003 guidelines for software engineering for electronic voting processes. ISO/IEC 90003 provides guidance in the application of ISO

9001:2000 to the acquisition, supply, development, operation and maintenance of computer software and related support services. It frequently references the ISO/IEC JTC 1/SC 7 software engineering standards: in particular ISO/IEC 12207, ISO/IEC TR 9126, ISO/IEC 14598, ISO/IEC 15939 and ISO/IEC TR 15504. According to ISO requirements, the quality system is documented which includes project management, delivery, software development processes and software quality assurance, among others.

Applus

CERTIFICATE

Number EC-3469/08

LGAI Technological Center, S.A.
certifies that the Quality Management System of the organization:

SCYTL SECURE ELECTRONIC VOTING, S.A.

C/ Tuset 20, 1-7
E-08006 BARCELONA

For the following activities:

Software engineering for electronic voting processes, according to UNE-ISO/IEC 90003 guidelines,

is in accordance with the requirements of the standard
ISO 9001:2000

ENAC CERTIFICACIÓN Nº 02/C-SC032

This certificate is valid until March 20, 2011
Cerdanyola del Vallès , March 20, 2008

General Director: Ramon Capellades i Font

Systems Certification Technical Manager: Miquel Sitjes Cabanas

This certificate shall be valid provided that all the conditions of the contract are fulfilled.
LGAI Technological Center, S.A. Campus U.A.B., s/n, 08193 Bellaterra, Barcelona

Ed. 1

3.2. Elaboration of requirement ST4.2

In SSA-U Appendix 6, Administrative provisions, section 3.1 we have attached ErgoGroup certificate for our ISO 27001:2005 certification. Please note that this certificate covers our data centre operations as well as our System Development Process i.e. "Programvareutvikling".

ErgoGroup states that the development of the system will be developed to satisfy the requirements of a data centre operator certified to or working according to ISO/IEC 20000-1 and ISO 27001.

3.3. Elaboration of requirement ST4.3

The contractor is certified for ISO 9001:2000 and ISO 27001:2005, and consider further certifications. ISO 20000-1 is highly actual for future certifications. The contractor´s processes and process architecture for Application Management comply with ITIL and ISO 20000-1.

To ensure continuously improvements of process orientation, the contractor use Det Norske Veritas for revision. The revisions focus on the most critical processes and the cooperation between them. The result of revisions is attention and priority in all business levels.