Bundesamt für Sicherheit in der Informationstechnik

E-Government Phase Plan

Phase 5 "Realisation and test"

The current text forms a module of the

E-Government-Manual

http://www.e-government-handbuch.de

Editorial staff: E-Government project group of the
Bundesamt für Sicherheit in der
Informationstechnik (BSI)

Contact details: **egov@bsi.bund.de**

Information on this module

Status	BSI contribution
Author	Dr. Blum (BSI)
Point of contact / contact details	Herr Dr. Blum (BSI), mailto:egov@bsi.bund.de

Amendment History

Date	Name	Change
03-18-2004	Horn	Integration of recommended changes
08-06-2003	Horn	Editorial revisions
02-05-2003	Dr. Blum	First version

© 2004
Bundesamt für Sicherheit in der Informationstechnik
Godesberger Allee 185-189, 53175 Bonn

Contents

5 Phase 5 “Realisation and test”

Input from the previous phases

The task of the first four phases of the phase plan was to identify the services of a public agency that it is feasible to implement online, then to optimise the processes and finally to map a process chain adapted to automated processing onto a DP model.

E-government service as process chain

At the end of Phase 4, a complete and detailed draft for the new e-government services is available in the form of the schedule of specifications. In this schedule all processes are described to the level for the so-called elementary processes (see Activity 4.6). These elementary processes form self-contained logical entities that although they can draw on a precisely defined (data) input from the results of other elementary processes, their internal operation is however independent of the internal workflow in other sub-steps. Furthermore, the detailed technical concept in the schedule of specifications includes an exact description of all the functionality of the new e-government application, the editing screens visible to the user on the screen as well as the interfaces to other technical procedures, e. g. to the Document Management System or budget monitoring.

Elementary processes

Detailed technical concept

This detailed plan of the process chain prepared from a technical point of view, but that is still in generally understandable language, i. e. not in a formalised DP language (e. g. pseudo code or similar), can now be implemented directly in a formal program structure plan (detailed DP concept) by the software developers. The detailed DP concept forms the basis for the program code for the e-government application.

Detailed DP concept

Along with the detailed flowchart, the detailed technical concept also includes the exact description of the data model (Activity 4.5), which forms the basis used by the DP developer for the definition of the design of the database(s) that form the backend of the e-government application. Among other aspects, the role-based access rights to the data stored in the backend are defined based on the security concept developed in Activities 4.7 and 4.10.

Databases

Further security requirements that relate to the protection of the exchange of data in terms of confidentiality, integrity and authentication, aspects that are particularly important for e-government applications, were defined in Activities 4.3 and 4.4. These must be applied in the subsequent implementation by means of the appropriate selection of the hardware and software. During this process, in particular the standards (SAGA) and basic components (see Activity 4.2) developed especially for e-government services must be observed.

E-government standards (SAGA), basic components

Objectives of Phase 5

As already expressed by the title, the objective of the fifth phase is the realisation of the previously developed concept for the new e-government services as well as the performance of the first tests under realistic working conditions. ”Realistic working conditions” are to be understood as a test environment at this stage, that although it is completely separated from productive operation, simulates productive operation as far as possible within financially justifiable limits.

Software production

The first activity in this phase comprises the preparation of the software that controls the overall application. As this is primarily the work of the programmers, during this task the role of the members of the E-Government Team is primarily a supporting or controlling role. They must ensure that the implementation actually corresponds to the requirements of the detailed technical concept and that the software prepared satisfies other formal criteria in relation to ease of use, interoperability, low maintenance etc.

Integration of standard software

Commensurate with the technical requirements from Activity 4.8, in Activity 5.2 the product-related definition of the hardware and software to be procured is prepared. In general the latter is standard software for extending the main application developed in Activity 5.1 (e. g. encryption software). During procurement and subsequent installation, the E-Government Team must ensure that the quality requirements defined are observed, similar to Activity 5.1. Both activities (5.1 and 5.2) are generally to be undertaken in parallel rather than one after the other.

Change management

The need for a controlled change request procedure was already highlighted in Activity 4.11. On the one hand, a desire for modifications to the overall concept prepared in Phase 4 can stem from a technical viewpoint, on the other hand, problems that only arise during the realisation phase and test phase can make such changes necessary. As part of Activity 5.3, here it is the task of the E-Government Team to establish controlled change management and to ensure that major delays in the progress of the project[1] are not caused by excessive requests for change.

Documentation

Complete documentation is both the prerequisite for the correct operation of the e-government application and the basis for training the users (on this topic also see Activity 4.13). The preparation of this documentation, which cannot be simply limited to a description of the hardware and software, but must also include information on the technical and organisational integration in the overall structure of the public agencies, staff regulations, work regulations and operating instructions for the staff, etc., is part of Activity 5.4.

Test and acceptance

The Activities 5.5 and 5.6 are the central task of Phase 5 from the point of view of those responsible for implementing e-government. On the one hand this involves the testing of the application completed by the contractor in relation to functionality required in the schedule of specifications, and also subsequent acceptance, which certifies that the contract has been correctly fulfilled by the contractor.

Adaptation of the IT security concept (product-related)

As during the procurement of hardware and software in Activity 5.2, new security-related aspects could arise due to the specific product selection, in Activity 5.7 the IT security concept must be adapted appropriately. Other modifications to the security concept can also result due to change management (Activity 5.3).

Phase conclusion

Phase 5 is completed with the provision of information to all concerned (Activity 5.8).

[1] It may be necessary to make use of change management during the entire realisation and test phase. This may then make it necessary to return to earlier activities (see the flowchart for Phase 5).

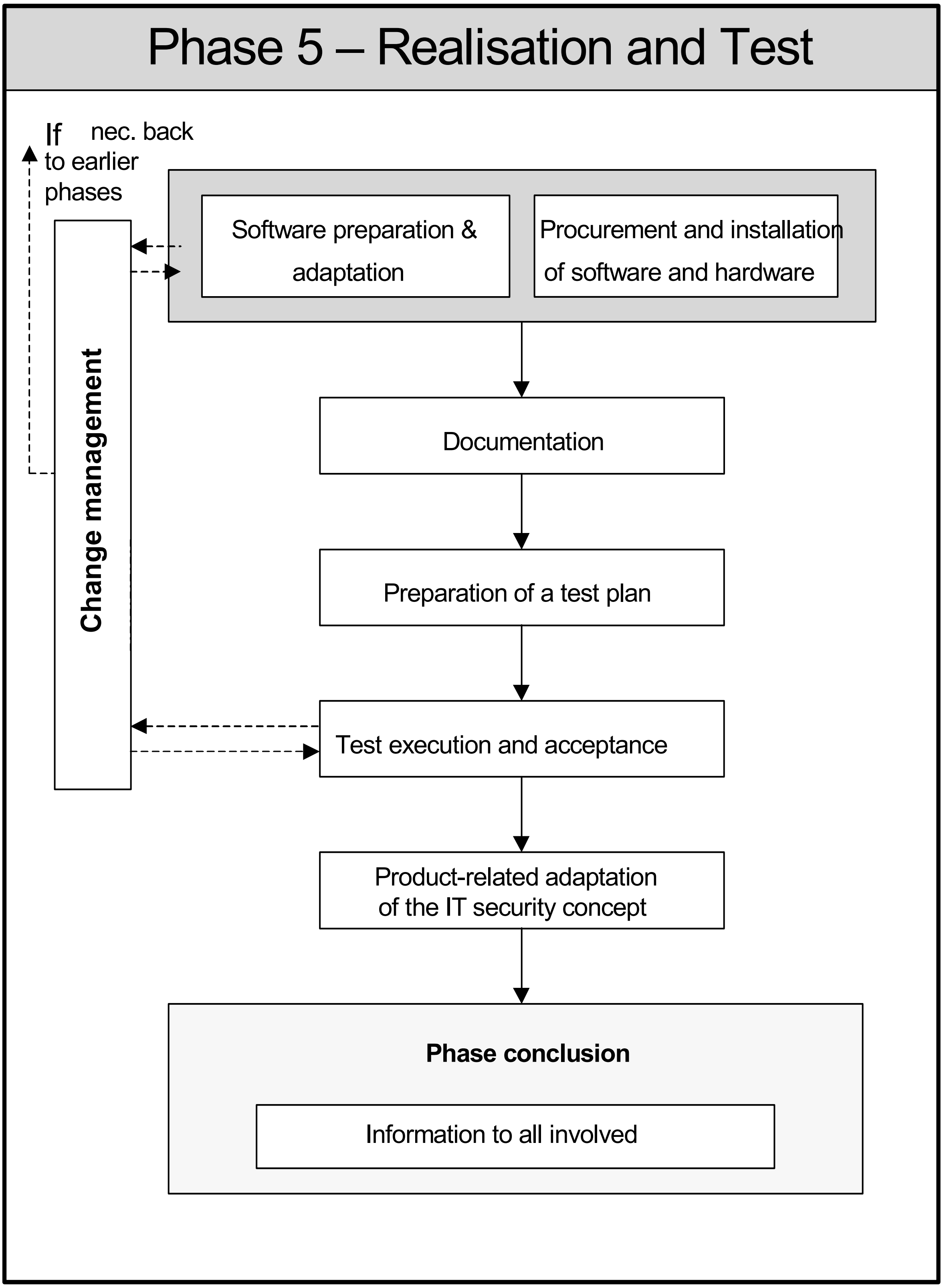
Phase 5 – Realisation and Test
If nec. back to earlier phases
Change management
Software preparation & adaptation
Procurement and installation of software and hardware
Documentation
Preparation of a test plan
Test execution and acceptance
Product-related adaptation of the IT security concept
Phase conclusion
Information to all involved

Figure 1: Flowchart for Phase 5 – Realisation and test

5.1 Activity "Software preparation and adaptation"

Initiation responsibility:	Head of the DP Development Team (contractor)
Implementation responsibility:	Programmers, Database specialists (contractor)
Support:	E-Government Core Team, IT system administrators, Persons in charge of the technical procedure (client)

In Phase 4 it was highlighted on various occasions (see e. g. activity 4.13) that there are different ways of realising the development of an e-government application depending on the know-how available in a public agency. Organisations with extensive personnel capacity in the IT area may be capable of planning and developing the new applications completely in-house. Public agencies on the other hand with relatively few IT personnel resources will be forced to award at least the realisation, and possibly also the planning of the e-government service to an external contractor.

In-house development versus contracting out

In the latter case those responsible for the initiation and implementation named in the header for this section are accordingly not staff at the public agency awarding the work. They are thus not subject to their direct authority, authority is defined via contractual agreements on compliance with e-government-specific standards and framework conditions (see next section).

It cannot be the objective of this phase plan to provide a compendium for the preparation of software in the e-government area or to define specific programming rules which would anyway be difficult to enforce for the above-mentioned reasons. Accordingly, the title of this Activity 5.1 should also not be taken with this meaning.

Control function of the E-Government Team

Instead this section briefly covers the tasks that arise for the E-Government Team during software development. These essentially comprise being available to programmers on the one hand as a contact partner for technical questions, for example on details of the different processes, and on the other hand to ensure that the product produced also actually corresponds to the draft laid down in the schedule of specifications.

Framework conditions

Compliance with standards

Attention especially to compliance with certain standards is more important in the e-government area than for "conventional" IT development projects due to the increased consequences. It is not to be expected that the customer will accept completely different designs for the appearance and functionality of the user interface for every public agency when the processes are in principle the same, e. g. making an application.

Style guides

In relation to the style elements to be used, for instance colour and font design or the use of graphics, logos etc., there exist binding style guides in many departments; these are to be adopted as appropriate to ensure as far as possible uniform design of the external form of e-government portals at least within the same business area.

Essential functionality of e-government services will in addition be defined by basic components, such as the virtual mail room, the form server, the payment transaction platform, etc., which will be developed centrally as part of the BundOnline 2005 initiative and be made available to the federal agencies (see Activity 4.2). This situation produces framework conditions for programming new e-government applications. Provided these directly utilise a basic component (e. g. the virtual mail room) as part of the overall system, it must be ensured that these components are securely integrated. Compatibility with these basic components must also be ensured on only indirect utilisation by means of external access (for instance to the payment transaction platform). In relation to this and other standards on the IT architecture of e-government applications to be observed, reference is again made to the SAGA paper already addressed in Activity 4.2.

Compatibility with basic components

Support to software development by the E-Government Team

Along with compliance with the e-government-specific standards defined by the SAGA paper, the E-Government Team should also ensure that programming is uniform and as far as possible standardised. Along with the use of established programming languages (e. g. C, C++) and database query languages (SQL), the software must also be as far as possible of modular design. The objective here is to breakdown the overall e-government application into program elements that are independent in terms of maintenance and operation and that are only connected to each other via exactly defined interfaces. This makes it easier to make any modifications, changes or additions to the programs that may become necessary later, even by other programmers. In this way dependence on one manufacturer, one member of staff or specific system hardware or software is to be avoided.

Uniform, standardised programming

Organisationally it is therefore recommended that regular meetings are held between representatives of the E-Government Team and the software developers during the realisation phase. A key milestone in this phase is the completion of the detailed DP concept as well as its checking and acceptance by the E-Government Team. In its property as a complete structure plan for programming the e-government application, the flow logic and all DP related properties of the source code are defined in detail in the detailed DP concept (on this topic see also the overview in the “DP design and realisation” section in Activity 5.4 “Documentation”). This is thus the best way to see how far the criteria described above for uniform and standardised programming have also actually been met.

Regular meetings

Detailed DP concept

The tasks of the public agency as the client for the development of an e-government application are thus certainly not completed with the award of the work to an external contractor. Instead the client has an obligation to be involved also during the realisation phase. A detailed discussion on the rules for the realisation of DP projects in the federal administration, which goes beyond the concise summary focussed particularly on e-government-specific aspects in this section, is given in Volume 5 of the series of publications from the Bundesbeauftragten für Wirtschaftlichkeit in der Bundesverwaltung[1]. Reference is also made to the

Client‘s involvement obligation

[1] “Datenverarbeitung in der Bundesverwaltung II”, Verlag W. Kohlhammer, Stuttgart (1993)

“Ergänzenden Vertragsbedingungen für die Beschaffung von IT-Leistungen” EVB-IT/ BVB[2] already addressed in Activity 4.12, here in particular “BVB-Erstellung”.

EVB-IT/ BVB

Even if the realisation of the project is not awarded to an external contractor, but the IT specialists within the public agency develop the project themselves, the control function of the E-Government Team is the same as described previously.

[2] These documents can be downloaded from the web site run by the federal government's co-ordinating and advisory body for information technology in the federal administration (KBSt) at www.kbst.bund.de.

5.2 Activity “Procurement and installation of software and hardware”

Initiation responsibility: Head of the E-Government Team

Implementation responsibility: IT specialists, Procurement department

Independent procurement versus solution “from a single source”

The procedure during the procurement and installation of hardware and software will certainly vary considerably in the different e-government projects. This is due on the one hand to very different hardware and software requirements depending on the type of e-government application, on the other hand the question of whether the project is realised externally or awarded externally also plays a role. Specifically, in the latter case there exists the possibility of a “solution from a single source”, i.e., the contractor supplies the hardware (and any necessary standard software) in addition to the application developed by the contractor and also undertakes all installation work. This has the advantage for the awarding public agency that its procurement tasks are reduced and in the case of a complaint it is only necessary to tackle the issue with one contractor. The disadvantage of such “package solutions” is, however, that these are in general very expensive and the client is also dependent on a single supplier.

In the following the alternative scenario is assumed, that the hardware is procured by the public agency independent of the application development[1]. As during this process, many procedures are similar to “normal” IT procurement, only a few e-government-specific aspects will be briefly addressed in the context of this activity.

Framework conditions

Market survey

During the planning for restructuring of the IT environment that will become necessary on the introduction of the new e-government services (Activity 4.8) the general criteria for the software and hardware to be procured were defined. However, this was initially performed independent of manufacturer and product. Accordingly in a second step a market survey is now to be performed by IT specialists and a list of products that satisfy the criteria stated is to be compiled. Special attention should be paid to the compliance of the products with the corresponding standards and architectures for e-government applications. These are defined in the SAGA paper (see Activity 4.2) already mentioned several times.

Criterion: compliance with SAGA

[1] If the software and hardware are procured by the contractor who also programs the application, then in the contractual agreements the contractor is also to be given the obligation to also comply with the binding standards (see SAGA paper) for e-government services also for these additional components.

Evaluation of the products

Product evaluation in agreement with the software developers

Based on the framework conditions resulting from the e-government standards, an initial selection can be made from the products that have been identified in Activity 4.12 as suitable for the new application from a technical point of view. In the following step these products must then be compared for strengths and weaknesses in relation to the requirements to be met. Such an evaluation should be performed in close agreement with the application developers, the regular meetings between representatives of the E-Government Team and the programmers can be utilised for this purpose.

Despite standardised and uniform programming, as required in the last activity so that it can be ensured among other aspects that the software will run on any system hardware, it has however been found useful in practice for the software developers to be aware of which hardware platform the application is to be installed on subsequently. The *capability* of running the application on any hardware does make any statement, for instance, on the performance of the software in the corresponding system environment. Often it is possible to considerably optimise the run-time behaviour on specific hardware by making minor modifications to the programs, without degrading the compatibility with other system platforms.

Legal stipulations related to procurement

Along with the technical aspects, financial aspects of course also play an important role in the selection of the hardware and software. As this is however not an e-government-specific topic, at this point reference is only made to the applicable regulations, as are to be found e. g. in the VOL in connection with the EVB-IT and the BVB often mentioned previously.

For certain system components, the result of the evaluation process can however be that products that satisfy both the technical requirements and the financial requirements are simply not available on the market. Such a case makes it necessary to return to Activity 4.8 where the planning for the restructured IT environment is to be modified such that it matches the conditions on the market.

In relation to situations as described above, it is again highlighted that the activities within the phase plan cannot (and should not) always be worked through in the order stipulated here. Depending on the situation in the specific case, it may be necessary to work through various activities in parallel or in a different order. Thus, for instance, the first two activities of Phase 5 should be performed either in parallel or at least closely co-ordinated.

Installation of the new hardware and software

After the new system components have been procured, they will be all the easier to integrate in the existing IT environment, the more carefully the planning in Activities 4.8 and 5.2 was performed. With the installation of the application developed in Activity 5.1, the e -government service is then available completely for the first time for test purposes. However, prior to the start of the test phase, two further important activities are to be performed.

5.3 Activity “Change management”

Initiation responsibility:	Head of the E-Government Team
Implementation responsibility:	E-Government Team, Persons in charge of organisation

With the completion of the schedule of specifications in Phase 4, the technical concept for the new e-government service should be actually complete. Major changes to this concept should, given careful prior planning, therefore not be necessary during the realisation phase.

Reasons for possible change requests

However, as has been found in practice, there is often a need with complex applications to make some modifications to the original concept after the planning phase has been completed. The reasons for this need may be for instance:

- Desirable extensions to the functionality
- Adaptation to new framework conditions, e. g. due to changes in the law or ordinances
- Unexpected difficulties in the realisation of the original concept

In principle when making changes the following guideline should be used: as much as absolutely necessary, as little as possible. In particular, the E-Government Team must ensure by means of a controlled change request procedure that the smooth progress of the project is not impaired by changes or is only impaired insignificantly.

Legal framework conditions related to the award of contracts

In § 5 para. 1 of the BVB-Erstellung[1] it is stated: “Der Auftraggeber kann bis zur Abnahme oder dem im Erstellungsschein vereinbarten Zeitpunkt schriftlich die Änderung der im Erstellungsschein festgelegten Anforderungen an die Programme verlangen. Der Auftragnehmer hat die geänderten Leistungen auszuführen, soweit sie ihm im Rahmen seiner betrieblichen Leistungsfähigkeit nicht unzumutbar sind.” (The client can demand, in writing, changes to the requirements for the program specified in the specification up until acceptance or the date agreed in the specification. The contractor must provide the modified services if they are reasonable within the contractor’s capabilities.)

Obligation of the contractor to implement changes

In principle the contractor thus has the obligation to make changes to the application requested by the client public agency when these are reasonable. The BVB defines a period of 21 calendar days for the rejection of a request for a change that the contractor considers unreasonable.

If the change affects, e. g. the price or the schedule for completing the order, the BVB includes provisions that the contractual agreements are to be changed without delay taking into account the related additional or reduced effort. If,

[1] On this topic see the Internet link given in Activity 5.1.

however, there is no agreement between the client and the contractor, the work is continued as per the existing agreement unless the client terminates the agreement on the basis of § 649 BGB. In the latter case, the client must however pay the contractor for the services provided up until this time.

Change request procedure

It was already highlighted in Activity 4.11 that requests for changes are only to made in writing and in accordance with certain formalities. First the department affected checks whether the change is sensible and necessary, and formulates a statement on the request. This must, among other aspects, include an estimate in relation to the effects on the actual procedure affected and if necessary on other applications or processes. In addition, the anticipated costs are to be compared with the benefit that the change is to bring and the time frame for implementation is to be estimated.

Justification of requests for changes

Based on this technical statement, the E-Government Team checks the change request and makes a decision as to its acceptance or rejection. If necessary the budget department is also to be involved in this decision if additional funds must be provided for the change.

The above-mentioned period of 21 calendar days starts with the receipt of the written request for the change, within this period the contractor must make decision on whether the change in the services is reasonable. If the contractor accepts the change, the contractual agreements as well as the schedule of specifications that forms the basis for the agreements are to be changed as appropriate and by mutual agreement.

Period for acceptance or rejection of requests for change by the contractor

The last step is of course not necessary if the public agency is having the application developed in-house. However, even in this case the change request procedure in the form described above should be retained and under no circumstances are change requests to be made by a department “calling” the development team. The formalisation, that is to avoid saying the bureaucratisation of the procedure, is needed in this case to avoid cost and schedule overruns due to excessive requests for change.

Formalised procedure for channelling requests for change

5.4 Activity “Documentation”

Initiation responsibility: Head of the DP Development Team

Implementation responsibility: Programmers, Database specialists, IT specialists, Persons in charge of the technical procedure

In Activity 5.1 it was already highlighted that it must be possible to have any modifications, changes or additions to the software[1] that may become necessary later also implemented by other programmers (that is not the developers)[2]. However, this is only ensured if comprehensive documentation of all system components is available.

Tools for supporting the documentation of the source code

In practice, it has been found that in general the most efficient method is when the program documentation is prepared in parallel with the programming. The starting point for this task is detailed commenting of all program steps and modules in the source code. If the system developer keeps to specific formal rules during this process, significant parts of the complete program documentation can now be generated automatically from the commented source code using tools available on the market. This reduces the tedious routine workload on the programmer and helps the programmer to concentrate on the actual system development.

Along with its significance for subsequent ease of use, maintenance and further development of the application, documentation produced in parallel with development also has the advantage that errors and inconsistencies can be detected and rectified at an early stage.

Apart from a description of the program, the complete documentation of course also includes further elements that are briefly explained in the following[2].

Procedure concept

First the documentation must include an exact description of all procedure functions included in the e-government service and their DP-related implementation. Specifically this includes:

Description of all procedure functions

- Remit and role of the procedure
- Description of the solution path (outline technical concept)
- Description of the DP modelling of the functionality of the procedure (detailed technical concept)
- Representation of the interaction with other technical procedures

[1] This of course only refers to the programs individually developed for the new e-government service and not the standard software procured in addition in Activity 5.2 if necessary.

[2] Reference here is also once again made to the paper cited in Activity 5.1 “Datenverarbeitung in der Bundesverwaltung”.

This part of the documentation has already largely been realised in Phase 4 by means of the preparation of the design specification and schedule of specifications (see Activity 4.12). If necessary, modifications must be made if changes to the previous concept have resulted, e. g., from Activity 5.3.

DP design and realisation

The second part of the documentation encompasses the exact description of the DP-related realisation of the applications. In particular, this includes:

- Complete program structure diagram
- Detailed description of the databases (tables, fields, formats, keys, indices etc.) as well as the access rights structure
- Listing of all files required (e. g. also auxiliary files for intermediate results among others.)
- All external data interfaces (layout of the records, data formats)
- Program documentation (including commented source code)
- Exact description of the hardware configuration (including the documentation on each of the hardware components)
- Documentation of the additional standard software used

Detailed DP concept

Documentation of the source code

This part of the documentation is also largely available with the detailed DP concept as the result of Activity 5.1 and, like the detailed technical concept, only needs to be adapted to the changes made in the meantime. Not included in the detailed DP concept however is the source code that is to be added to the documentation.

User documentation for technical staff

It was already highlighted in Activity 4.13 that the documentation should be divided up by target group. While the previous sections are directed particularly at IT specialists whose task is the maintenance and further development of the programs (system programmers), the documentation for the technical staff who support the application in day-to-day operation (administrators) must cover in particular the following points:

- Configuration of the hardware and software (e. g. firewall, databases)
- Data and files for operation
- User management (e. g. access rights)
- Instructions on backup

Technical documentation

User documentation for non-technical staff

The users in the public agencies should be provided with information on the following topics:

- Brief, understandable introduction to the system
- Functional overview
- Navigation in the application and use of the input forms
- Detailed instructions for special sub-tasks
- Help
- List of error messages, instructions on the rectification of errors
- Regulations and staff regulations on the procedure

Specialist documentation

User guide for external users of the e-government service

Unlike the other IT applications in use in the public agencies, the public in the role of the customer for an e-government service is also involved *actively* involved in the procedure. Accordingly, information on the operation of the application must be made available to the public. The nature and scope of such documentation depends of course largely on the e-government service in the specific case. Generally, it can therefore only be stated that the corresponding documents, in an appropriately modified form, should cover the same topics as the documents listed in the previous section for the specialists. Furthermore, the external user must receive information on the (minimum) requirements on his system environment, e. g.:

Customer-related documentation

- Hardware requirements (processor performance, main memory, spare hard disk space, etc.)
- The operating systems supported by the application
- Additional devices required (e. g. chip-card reader)
- Additional standard software required (e. g. web browser ... from version ...)
- ...

Minimum requirements for customer‘s system environment

The question of how this information can be made available to the external user in the most suitable form will be a topic of the activity “Public relations” as part of Phase 6. Useful guidance on this issue is also included in the module “Quality criteria for a public-user-friendly and secure website” in this manual.

Other components of the documentation

Along with the previously mentioned topics, complete documentation also includes:

Security concept, test results

- The security concept
- The test plan with the audit trails for the test results

These points will be discussed in detail in the three following activities.

5.5 Activity “Preparation of a test plan”

Initiation responsibility: Head of the E-Government Team

Implementation responsibility: E-Government Team, IT specialists, Technical manager

Review of compliance with the specifications in the schedule of specifications

The most important task after completion of the new e-government application from the point of view of the client is initially to check whether the product supplied corresponds on the one hand to the specifications defined in the schedule of specifications and on the other hand with the requirements placed in practice. The former is addressed in the test phase for the new application to be discussed in the following, the latter is addressed by the pilot operation with selected users planned for Phase 6.

The test phase itself is divided into two sections: while the objective of this activity is the preparation of a systematic test plan, Activity 5.6 that follows includes the performance of the test and subsequent acceptance of the new e-government application.

Framework conditions

In principle the procedure for test and acceptance of IT systems developed for the federal administration are regulated by the previously mentioned publication “Datenverarbeitung in der Bundesverwaltung” (see quotation in Activity 5.1). In section 3.4.5.1 it stated in this document:

Formal requirements on the test procedure

“IT-Verfahren sind vor ihrer Freigabe für den Betrieb, bei komplexen Verfahren auch schrittweise während der Erstellung der Programme, in allen Funktionen zu testen. Dabei ist den Berührungspunkten zu anderen Verfahren und der späteren organisatorischen Einbindung in den Betrieb besondere Beachtung zu schenken. Mit Hilfe der Tests muss insbesondere auch sichergestellt werden, dass jedes Programm nur die erforderlichen, in Vorgaben festzulegenden Funktionen erfüllt und keine unerwünschten Nebenwirkungen beinhaltet. Tests müssen unter Berücksichtigung aller Hard- und Software-Komponenten so durchgeführt werden, dass der Verfahrensbetrieb nicht beeinträchtigt wird.” (Before approval for operation, all the functions of IT procedures are to be tested, for complex procedures this can also be performed in steps during the preparation of the program. During this process particular attention is to be paid to the interfaces to other procedures and the subsequent organisational integration in operation. With the aid of the tests, it must also be ensured, in particular, that each program only provides the necessary functions defined in the requirements and does not produce any undesirable additional reactions. Tests must be performed taking into account all hardware and software components in such a manner that the operation of the procedure is not impaired.)

DIN ISO/IEC 12119

The efficient implementation of the objectives formulated in this way requires a systematic procedure during testing. This should be orientated on the binding quality requirements and test requirements for software products as are defined in the international standard DIN ISO/IEC 12119[1]. It cannot be an objective of this manual to describe the details of this test procedure. For detailed, practical information, reference is made here to the IT Baseline Protection Manual (GSHB) from the BSI, in particular the Safeguards S 2.82 to S 2.85. Although in this manual reference is made to the testing of standard software, the procedure can be transferred similarly to individual solutions such as the e-government applications considered here. Should a more detailed analysis for a special procedure be found to be necessary, here reference is made to the above-mentioned DIN ISO/IEC 12119 as well as the specialist literature based on this standard.

IT Baseline Protection Manual

In the following the key steps in a test procedure will be briefly outlined taking into account e-government-specific aspects.

Test plan

To ensure a systematic procedure during the performance of the tests, first a test plan that contains the following points is to be prepared:

Test content

- Definition of the test content: This is made based on the documentation on the application compiled in Activity 5.4. First the requirements that are to be reviewed are selected from the schedule of specifications. This results in a list of system components to be tested. Depending on the degree of complexity of the e-government application, during this process it is to be decided whether these components are to be tested individually or as a whole. It is also important that the exact version numbers of the parts of the program used in the test are documented. If, for specific reasons, certain components are to be excluded from the tests, this is also to be documented. Along with the actual DP components, the related documents (see Activity 5.4) are however also to be reviewed as to whether they are free or errors, complete and free of contradictions.

Test cases

- Test objectives: Test cases are to be built with the input data defined in advance and the outputs expected on correct execution (the latter is necessary to define objective criteria for success or failure of a test run). The test cases are so designed that they achieve the maximum possible coverage in relation to all conceivable constellations in operation. Specifically this means that:
  - Values from all data areas,
  - All possible program paths,
  - Every individual component,
  - Every statement

  are to be run through in examples[1].

Black box and white box tests

- Test methods: For every functional unit the procedure to be used for testing the unit is to be given. Here, e. g., functionality tests, black box or white box

[1] The international standard was derived from DIN 66285 and is identical.

[1] In relation to further specification of test records see also the section “Test preparation” in Activity 5.6.

tests can be used. The first of these are simply for demonstrating that the required specifications are met. If the functionality is are only checked based on the output results for all possible test inputs without consideration of the processing in the program, the term black box test is used. This method, in which the program is so to speak a “black box” with unknown content, is always needed if the source code is not available, as is the case for instance in general with standard software.

Testing the source code

Conversely, if the source code is available, then not just the result itself but also how it is produced due to the flow in the program can also be checked. The “black” box mentioned previously then becomes so to speak “transparent” (white box). For an individually developed e-government application, the source code should in general be available. However, as white box tests can be very time-consuming (in the majority of cases they also go beyond the requirements of the IT Baseline Protection), they should only be used for technically important components or components that are critical for security-related reasons. Also additional white box analyses are recommendable when a black box test delivers erroneous results. Procedures such as a formal verification of correctness, symbolic program execution, failure analysis, etc. are available as analysis methods.

- Tools for test support: Along with the test methods, the test plan should also define suitable test tools for each of the components to be tested. Examples of these are:

  Static analysis

  - Tools for program structure review (static analysis): these are used for finding formal errors, such as the use of data fields without prior value assignment, failure to use data fields or parts of the program, etc. Many of these functions are already integrated in modern compilers.

  Dynamic analysis

  - Tools for program flow review (dynamic analysis): these permit a search for errors by following the flow of the program step-by-step (“debugging”).

  Test data generator

  - Test data generator: is used for the preparation of representative test records with the highest possible coverage (cf. section “Test objectives”).

Test environment separated from productive operation

- Test environment: in the section quoted above from the publication “Datenverarbeitung in der Bundesverwaltung” it is stated that the operation must not be impaired by the tests. For this reason, test operation must take place in a dedicated test environment completely isolated from production operation. On the one hand this should as far as possible model a complete image of the subsequent operational environment, on the other hand it must not be so complex that it ceases to be cost-effective. Correspondingly, this test environment must be exactly defined in the test plan. Here it is to be stated which resources (computer resources, IT infrastructure) must be available and to what extent.

- Personnel and responsibilities: the testing of complex developments like that generally represented by e-government applications can necessarily only be performed by a group of persons responsible for testing with appropriate division of labour. This is to be exactly specified in the test plan according to the specialist knowledge necessary for the individual test tasks. In general it involves:

Persons responsible for testing

  - Programmers,
  - Other IT specialists (e. g. network experts),
  - IT planners,
  - Representatives from the specialist department.

Separation of test execution and examination of the results

On the distribution of the tasks within this group, it is to be ensured that the functions "test execution" and "inspection of the results" are clearly separated in relation to personnel (two person rule). Under this framework condition, it is to be defined in the test plan who is responsible for which task for every item of test content.

- Test flow and time planning: to have the tests run purposefully and in a co-ordinated manner, the sequence for the test steps as well as their scheduling are to be specified as exactly as possible. Specifically, the following should be defined in the test plan:

  Schedule

  - Sequence of the tests,
  - Dates and amount of time required for the individual test steps,
  - Dates and amount of time required for material to be made available (e. g. additional servers to be installed) and human resources.

  Resource planning

  Of course, it cannot be assumed that everything will always go "smoothly" particularly in a test phase. If errors are found in a test step (actually the objective of the testing), the analysis of the errors can take more time than provided in the schedule outlined above. For this reason the schedule should not be ambitious, it should certainly contain some time for contingencies. Particularly the material resources should not be "skimped" in the planning: if a bottleneck in the schedule can be bridged for instance by the installation of an additional server, as a rule the costs are considerably lower than if a project delay or additional human resources are approved.

- Definition of decision criteria: the test plan must define criteria as to how errors that occur are to be evaluated. A possible categorisation is provided e. g. by the following scheme (see IT Baseline Protection Manual, Safeguard S 2.81):

  Categorisation of tolerable and intolerable errors

  - Error category 0: requirements are not met; there are intolerable errors that cannot be rectified.
  - Error category 1: requirements are met, however there are reservations (restricted functionality); minor errors have been found that either occur very rarely in productive operation or that occur with effects that are negligible.
  - Error category 2: requirements are met in full; errors that may have arisen can either be rectified or have no effect whatsoever on productive operation.

Furthermore, criteria are to be defined as to when the testing is to be interrupted (for instance to make improvements in individual components) and which activities are to be repeated when testing is recommenced.

Criteria for interrupting testing

5.6 Activity “Test execution and acceptance”

Initiation responsibility: Head of the E-Government Team

Implementation responsibility: E-Government Team, IT specialists, Technical managers

Based on the test plan prepared in the last activity, the actual test phase for the new e-government application now follows. When all “significant[1]” errors found during the testing have been rectified by the contractor, the client public agency can then provide acceptance.

Test preparation

Prior to the start of the test phase, additional resources for the evaluation are to be made available as per the test plan:

- Preparation of checklists: checklists are the most suitable method of systematically and clearly documenting the test results.

Checklists

- Generation of the test data: as per the schedule requirements defined in the test plan, an appropriate number of test records is to be generated. For each of these records the result to be expected if the application functions correctly is to be documented at the same time. To achieve the level of coverage sought, records must be generated from the three following categories:

Test data

  - Standard cases: this is data as normally arises in practice. This data are used for reviewing the correct functionality of the application. As from experience programs often have problems processing inputs at the limits of the permissible value range, the test records should also include such “limit values” as well as the “normal values”.

Normal values and limit values for test data

  - Error cases: with these cases it is to be determined whether the application is capable of intercepting erroneous inputs (e. g. a name instead of a date of birth) and to respond with a meaningful error message.

Intercepting erroneous inputs

  - Exception cases: with these inputs the program must react differently than for standard cases (e. g. leap year rule: accept date of birth 29.02.2000, but do not accept 29.02.1900).

Correct processing of special cases

  The generation of test data can be performed in various ways:

  - “Synthetic” data: these data are generated “by hand” or with the aid of test data generators as per the categories described above. They have the advantage that by means of careful selection the tester can rapidly find errors already suspected or errors that are frequently found based on experience.

Test with fictive data

[1] As per BVB-Erstellung (§ 10 para. 4) acceptance is not allowed to be denied due to “insignificant deviations” from the specification.

  - Real data: the use of data that have arisen in earlier business processes for test purpose is only permissible if the data are first rendered anonymous. The advantage is that on the one hand it is not necessary to generate synthetic test data and that the cases tested are actually real. In this way completely unexpected errors are often found, as "real life" is always more creative than the programmer. However, to achieve the level of coverage sought, real records must always be supplemented with synthetically generated "limit cases".

Test with realistic data

- Preparation of the test environment: the test environment described in the test plan must be setup and the applications to be tested installed. During this process the configuration of each component is to be documented, so that the test environment in this form can be reproduced at any time.

Installing the test environment

Test execution

The test is performed as per the test plan, here every single step and the related result are to be documented in such a manner that the test can be repeated in the same manner at any time and checked. For functional tests, in particular the following is to be examined:

Functional check

- Presence of the function
- Correctness of the function: is the execution error-free?
- Suitability of the function: is the task actually completely performed?
- Freedom from contradictions: are there discrepancies between the documentation and the program?

Along with these functions directly orientated on the task, the following properties of the application, which are no less important for productive operation, should also be tested:

Testing ergonomic and run-time properties

- Performance: can a process be performed in a period of time tolerable for the user?
- Reliability: can the last working state prior to a system crash (e. g. as a consequence of a power failure) be reproduced afterwards? In the data stock still consistent?
- Ease of use: is the frontend ergonomically designed?
- Maintainability: is the application easy to administrate?
- Documentation: is the documentation (as per the requirements from Activity 5.4) complete, correct and error-free? Is it also unambiguous, understandable and clearly laid out?

Along with the functional properties of the application, further test criteria are:

Testing of formal properties

- Compatibility
- Interoperability
- Compliance with standards
- Compliance with (public agency) internal rules and legal regulations
- Software quality: has the source code been prepared in accordance with the requirements in Activity 5.1 in relation to uniformity, standardisation, commenting, etc.?

Security-specific tests

Particular attention should be paid during testing of the new e-government application as to whether the security-specific requirements defined in Activity 4.7, part B, have actually been implemented. For each of the required security functions, the following are to be checked:

Testing of the security functions

- Presence of the function: are, e. g., all accesses to person related data logged?
- Effectiveness and correctness of the security function: can, e. g., individual data fields be manipulated during authorised (logged) access without this being recorded in the log?
- Strength of the security mechanisms: can unauthorised persons, e. g. due to weak password encryption, obtain access rights to the data?
- Protection against bypassing and enforcement of the security mechanisms: is it possible to log on to the database with a different frontend bypassing the application and thus obtain access to the data? (see e. g. the scenario described in section B3 of activity 4.7)

Further detailed explanations of testing security functions are given in safeguard S 2.83 of the IT Baseline Protection Manual.

Penetration tests

Due to their high level of exposure as well the generally high level of confidentiality of the data exchanged, for e-government services there is higher risk of system break in from the outside than for conventional application mostly only operated internally within the public agency (e. g. hacker attack). Here the special category of security-specific tests with the objective of uncovering weak spots in the application in relation to this hazard will be discussed briefly.

Analysis for possible weak spots

Penetration tests should be performed right at the end of the test phase when all other tests have been completed. During these tests there is specifically the risk that the test environment may be seriously damaged. To avoid serious consequences and in particular to prevent the loss of the test results obtained previously, a data backup should in all circumstances be performed prior to performing penetration tests.

During penetration tests, weak spots in the application are specifically sought with the aid of hacker tools. It is clear that during this process, the application needs to be fairly “roughly “ handled. After all during their attacks hackers do not as a rule have any consideration for whether their tampering may cause serious damage to the system attacked (often destruction is the actual purpose of the attack). The precautionary measures described previously are therefore certainly not superfluous, even if the application is “only being tested”.

Evaluation criteria for security against attacks from the exterior

The objective of penetration tests is to test the actual strength of the security mechanisms in the application and to categorise them accordingly. Here the expertise that a potential attacker requires is used to categorise the results as well as the effort the attacker must make to overcome the security precautions. Thus, for instance, the security level is categorised as “high” when a trained IT expert even with expensive special equipment and the aid of insiders needs several months to break into the system (see IT Baseline Protection Manual, Safeguard S 2.83).

Conversely, if penetration tests come to the result that an attacker with some basic IT knowledge and perhaps a few hacker tools freely available on the Internet is able to bypass the security mechanisms in the e-government application in a few minutes, then, euphemistically expressed, there is considerable need to completely revise the entire security concept.

Test evaluation

Target versus actual comparison

The tests are evaluated based on the decision criteria defined in the test plan. For this purpose as a rule a target versus actual comparison between the results to be expected on correct program execution and the results actual provided during the test is first performed for the test records. If during this “black box process” discrepancies are found, then the first issue is to localise potential errors. For this purpose a more exact analysis of the program flow is then generally required (white box test). The error is allocated a category as per the classification scheme mentioned above, the action to be taken is then decided depending on the category. If the error is not serious and can be rectified, the results of the error analysis are to be documented and the tests continued. On the other hand, if errors occur that are so serious that the function of the entire application is placed in question, it may be necessary to interrupt the tests.

Refined analysis

Test documentation

Test documentation as part of the complete documentation of the application

The course and results of the test phase must be documented so that on the one hand if deficiencies are found, claims for improvement can be justified to the contractor and on the other hand to demonstrate that the necessary due care has been taken on the introduction of a new IT procedure. The test documentation is included as an integral part of the overall documentation (see Activity 5.4) for the application. It can be helpful during the analysis of any errors that occur subsequently during operation.

The test documentation should comprise the following points:

Components of the test documentation

- The entire test plan as per the list given above,
- A type of “logbook “ on the course of the test, including all checklists,

- The results of the evaluation of the test including the analyses of the system behaviour, the system messages, if necessary the error categorisation as well as technical evaluations of the test results by the departments involved (IT and specialist departments).

Acceptance

In principle it must first be noted that only those errors can be found as the result of the test that actually manifest themselves *as deviations from the specification*. Even if the application has passed all tests without any complaints, it is *not proved* that the software is actually completely error-free. Passing the tests only indicates that the contractor has completed the development work for the new e-government application as required by the contract.

100 % freedom from errors of software cannot be proved

If errors have been found during the tests, these are to advised to the contractor in writing without delay (§ 11 No. 4 BVB-Erstellung) and must be rectified by the contractor. As per the decision criteria defined in the test plan, the improved application must then be re-tested either as a whole or in part. If necessary this procedure may even need to be run through in iterations until all errors that cannot be regarded as negligible are rectified. The resulting delay can be penalised by the client with a contract penalty (§ 11 No. 6 BVB-Erstellung). If the attempts at improvements are on the other hand unsuccessful, the client can withdraw from the contract.

Improvement by contractor

If there are no deviations or only insignificant deviations[1] from the requirements in the specification, the client provides acceptance (§ 11 No. 5 BVB-Erstellung). The development of the new e-government service is then complete.

[1] Minor errors must also be rectified by the contractor in the context of the guarantee even after acceptance.

5.7 Activity "Adaptation of the IT security concept"

Initiation responsibility: E-Government Team leader

Implementation responsibility: IT security officer

The IT security concept has already been prepared in the planning phase for the new e-government service (see Activity 4.10). This was necessary because the security requirements defined in this concept have a significant effect on the realisation of the application. For instance the design of the database architecture is based on the role and access rights concept included in the security concept.

Security-relevant findings during the realisation and test phase

However, as a rule during the realisation and test phase, new aspects are found or changes must be made to the original design of the application that make a corresponding adjustment to the IT security concept necessary.

Product-related customisation

The security requirements placed on the hardware and software during the planning phase were formulated in relation to product *types*. They were used as selection criteria for the procurement of hardware and software in Activity 5.2. Once a decision has been made here for specific products, the product-specific security settings for this hardware and software can, and must, be precisely defined in the security concept.

Examples:

Security-specific configuration of the hardware and software

- Hardware: specification of the configuration of a firewall based on the security functions implemented in the product procured (e. g. configuration of filtering by network ports if the products provides this feature),
- Software: configuration of an encryption plug-in in the e-mail browser (e. g. setting for 2048-bit RSA encryption if the product provides this feature).

Other adaptations

A further need to modify the security concept prepared in Phase 4 can arise during the realisation of the application or in the test phase:

Security-specific adaptation due to technical changes

- Adaptation to suit a change request: the extensions to the functionality of the application required by a specialist department (see Activity 5.3) can make it necessary to extend the security concept.

Adaptation due to weak spots found during the test phase

- Changes due to the findings in the test phase: during a penetration test a serious weak spot in the application was found that must be rectified by appropriate security safeguards; this situation must be documented in the security concept.

5.8 Activity "Briefing of all those concerned"

Initiation responsibility: Public agency management

Implementation responsibility: Public agency management, Public relations, Staff representatives

Expression of appreciation to staff involved in development

Informing all involved that the new e-government service has been successfully completed should certainly be pleasant task after all the busy months of planning and realisation. As the development of the application would not have been possible without the dependable collaboration of the specialist departments with the IT specialists, the persons in charge of planning and organisation, the senior management of the public agency should acknowledge this fact. As a quasi "spearhead" for the project, the E-Government Team deserves special recognition.

Preparations for the pilot phase

In collaboration with the staff representatives, the preparations for the introduction of the application and the initial operation and the pilot phase must be made. Here the issue is to define the staff groups involved and to inform these groups. In addition, the staff who will in future work with the application must be informed about training schedules as well as any dates already planned.

Informing the public

Target group the public

However, the success of the new e-government service in practice depends predominantly on how it is perceived by its potential customers and the level of acceptance of the service among the potential customers. Here there is a major challenge for the public relations area in the public agency to make the new service known as widely as possible to public by means of a suitable policy of information in the media and to convey a positive impression of the new service. The procedure should be orientated on the related target group:

- If the users of the e-government service will predominantly be individual members of the public, then a targeted campaign in the print media is recommendable, particularly in the specialist press, as well as on radio and television. However here it does not appear useful to use "product advertising" in the conventional sense with more or less expressive advertisements and television spots (apart from the fact that the majority of public agencies would not have the funds for this type of advertising). Instead press releases, appearances at trade shows and the like should be used to arise the interest of the media in the new e-government service. A wide section of the public can then be addressed via the resulting reporting. The focus must always be the provision of factual information to the public. Confidence in the security of the new e-government service can only be won by serious arguments. The public must be convinced of the advantages that the use of the new service will provide them personally.

Target group business

- If the new e-government service forms an interface between administration and business (e. g. an electronic platform for processing procurement by the administration), then unlike for the case described previously, the user group is relatively limited. Correspondingly, more direct communication paths can be used here for the purpose of providing information on the new service. One possibility is to address the companies directly by e-mail or using conventional mail. In addition, trade shows or information events organised by the public agency can be used for making contacts.

Irrespective of the potential user group that is to be addressed: an important objective during the preparation for the subsequent introduction phase is to awake the interest of the public to take part in the pilot projects.

7 Checklists

The following checklist can be used to ascertain whether all the essential results of the present phase are available. It can also be used where the above activities have been carried out in a different order or in a different form.

7.5 Checklist for Phase 5

Outcome(s)	**Who?**	**When?**	**Done?**
Detailed DP concept has been checked and accepted			
Hardware and additional standard software for the new application have been procured			
All technically justified changes have been included			
The documentation has been prepared			
The test plan has been prepared			
Checklists for the tests have been prepared and the test data have been generated			
Functionality and penetration tests have been performed and test documentation prepared			
The new e-government service has been accepted			
The IT security concept has been adapted to suit the products			

8 Author Profile

Dr. Herbert Blum, BSI

Herbert Blum studied physics and electrical engineering at Saarbrücken University. After graduating, he moved to the University of Mainz where, in 1992, he was awarded a doctorate in nuclear physics. The many calculations he was required to perform as part of his thesis resulted in a concentration on IT-related issues. He subsequently spent several years working on large-scale IT projects in industry and the public services with a special focus on the development of client-server database applications. As a project manager, one of his achievements was to implement an electronic procurement system for Mainz University. In 1998, Herbert Blum took up a position in the Bundesamt für Sicherheit in der Informationstechnik where he was initially responsible for providing IT security training. As of September 2001, he has been contributing to the development of the E-Government Manual in the field of "application concepts and consultancy".

Dateiname:	3_Phase_5_en.doc
Verzeichnis:	E:\internet
Vorlage:	R:_AKTEN\BSI_EGOV_634-00-11\Dokumentenvorlage\egov.dot
Titel:	E-Government-Handbuch
Thema:	Phasenplan E-Government
Autor:	BSI
Stichwörter:	
Kommentar:	
Erstelldatum:	28.05.2004 09:22
Änderung Nummer:	44
Letztes Speicherdatum:	09.06.2004 12:00
Zuletzt gespeichert von:	GerhardDieter
Letztes Druckdatum:	09.06.2004 12:00
Nach letztem vollständigen Druck	
Anzahl Seiten:	32
Anzahl Wörter:	9.303 (ca.)
Anzahl Zeichen:	53.030 (ca.)