Deployment Package
Authentication in Mobiles
Basic Profile + Security

Notes:

This document is the intellectual propriety of its author's organization. However, information contained in this document is free of use.

This material is furnished on an "as-is" basis. The author(s) make(s) no warranties of any kind, either expressed or implied, as to any matter including, but not limited to, warranty of fitness for purpose or merchantability, exclusivity, or results obtained from use of the material.

The processes described in this Deployment Package are not intended to preclude or discourage the use of additional processes that Very Small Entities may find useful.

Author	P. Maciel – CIMAT A.C. Jezreel Mejía – CIMAT A.C. (México)
Editors	P. Maciel – CIMAT A.C. Jezreel Mejía – CIMAT A.C. (México)
Creation date	25/06/21
Last update	25/06/21
Version	0.1

Version History

Date (yyyy-mm-dd)	Version	Description	Author
2021-06-25	0.1	Document Creation	Perla Maciel

Abbreviations/Acronyms

Abre./Acro.	Definitions
DP	Deployment Package - a set of artefacts developed to facilitate the implementation of a set of practices, of the selected framework, in a Very Small Entity.
VSE	Very Small Entity – an enterprise, organization, department or project having up to 25 people.
VSEs	Very Small Entities
TL	Technical Leader
AN	Analyst
DES	Designer
PR	Programmer
PM	Project Manager
SA	Security Advisor

Table of Contents

1. Technical Description

Purpose of this document

This Deployment Package (DP) supports the Basic Profile as defined in ISO/IEC TR 29110-5-1-2:2011 Management and Engineering Guide. The Basic Profile is one profile of the Generic profile group. The Generic profile group is composed of 4 profiles: Entry, Basic, Intermediate and Advanced. The Generic profile group is applicable to VSEs that do not develop critical software. The Generic profile group does not imply any specific application domain. The Basic Profile describes software development of a single application by a single project team with no special risk or situational factors.

A DP is a set of artefacts developed to facilitate the implementation of a set of practices in a Very Small Entity (VSE). A DP is not a process reference model (i.e. it is not prescriptive). The elements of a typical DP are: description of processes, activities, tasks, roles and products, template, checklist, example, reference and reference to standards and models, and tools.

The content of this document is entirely *informative*.

This document has been produced by Perla Maciel and Jezreel Mejía of CIMAT A.C. (México).

Why is the Authentication in Mobiles Important?

This consideration takes in mind proving the identity of a user to get access to information managed in a system. This process use the mechanism of associating an identifying credentials to a user that are compared to the system database to assure the identity of the user.

2. Definitions

In this section, the reader will find two sets of definitions. The first set defines the terms used in all Deployment Packages, i.e. generic terms. The second set of terms used in this Deployment package, i.e. specific terms.

Generic Terms

Process: set of interrelated or interacting activities which transform inputs into outputs [ISO/IEC 12207].

Activity: a set of cohesive tasks of a process [ISO/IEC 12207].

Task: required, recommended, or permissible action, intended to contribute to the achievement of one or more outcomes of a process [ISO/IEC 12207].

Sub-Task: When a task is complex, it is divided into sub-tasks.

Step: In a deployment package, a task is decomposed in a sequence of steps.

Role: a defined function to be performed by a project team member, such as testing, filing, inspecting, coding. [ISO/IEC 24765]

Product: piece of information or deliverable that can be produced (not mandatory) by one or several tasks. *(e. g. design document, source code)*.

Artefact: information, which is not listed in ISO/IEC 29110 Part 5, but can help a VSE during the execution of a project.

3. Relationships with ISO/IEC 29110

This deployment package covers the activities related to Definition of Requirements of the ISO/IEC 29110 Part 5-1-2 for Very Small Entities (VSEs) – Basic Profile [ISO/IEC29110] taking in consideration security practices in Mobiles.

In this section, the reader will find a list of Software Implementation (SI) process, activities, tasks and roles from Part 5 that are directly related to this topic. This topic is described in details in the next section.

- **Process:**
  - **Software Implementation**
- **Activities:**
  - **SI.2 Software requirements analysis**
- **Tasks and Roles:**

Tasks	Roles
SI.2.2 Document or update the *Requirements Specification.* Identify and consult information sources (customer, users, previous systems, documents, etc.) in order to get new requirements. Analyse the identified requirements to determinate the scope and feasibility. Generate or update the *Requirements Specification.*	AN, CUS
SI.2.3 Verification and obtaining approval of the *Requirements Specification.* Verify the correctness and testability of the *Requirements Specification* and its consistency with the *Product Description*. Additionally, review that requirements are complete, unambiguous and not contradictory. The results found are documented in a *Verification Results* and corrections are made until the document is approved by AN. If significant changes were needed, initiate a *Change Request.*	AN, TL

4. Description of Processes, Activities, Tasks, Steps, Roles and Products

- **Process:**
  - **Software Implementation**
- **Activities:**
  - **SI.2 Software requirements analysis**
- **Tasks and Roles:**

Tasks	Roles
SI.2.2 Document or update the *Requirements Specification.* Identify and consult information sources (customer, users, previous systems, documents, etc.) in order to get new requirements. Analyse the identified requirements to determinate the scope and feasibility. Generate or update the *Requirements Specification.*	AN, CUS
SI.2.3 Verification and obtaining approval of the *Requirements Specification.* Verify the correctness and testability of the *Requirements Specification* and its consistency with the *Product Description*. Additionally, review that requirements are complete, unambiguous and not contradictory. The results found are documented in a *Verification Results* and corrections are made until the document is approved by AN. If significant changes were needed, initiate a *Change Request.*	AN, TL

This task is related with the following sub-tasks:

- Avoid Weak Patterns and Reinforce Authentication

Avoid Weak Patterns and Reinforce Authentication

Objectives:	Avoid weak patterns that make Insecure a Mobile Application and reinforce the authentication used
Roles:	Analyst
	Technical Leader
	Programmer
	Security Advisor
Artefacts:	Weak Authentication Patterns
Steps:	1. Define the kind of authentication needed for the Mobile Application

	2. Avoid the weak patterns for the two approaches
	3. Be aware of the Persistent authentication (Remember Me) functionality
	4. Avoid weak token/identifiers/passwords
Step Description:	***Step 1. Define the kind of authentication needed for the Mobile Application*** For a Mobile Application there are two ways to authenticate a user: client-side or server-side. Defining this approach make easier to know where to implement corrective actions to prevent authentication weakness. ***Step 2. Avoid the weak patterns for the two approaches*** Client-side bypass vulnerabilities can occur if the user authenticates locally. If the application stores data locally, the authentication can be bypassed on the device that is exited through a run or binary change operation. If you have strong business requirement for offline authentication If possible, make sure that all authentication requests are server-side. After successful authentication, application data is uploaded to the mobile device. This way, application data is only available after successful authentication. *TIP: The developers must assume that all client-side authorization and authentication controls can be bypassed by malicious users. Authorization and authentication controls need to be improved on the server side, if possible.* *TIP 2: When migrating a web application to a mobile application, the authentication requirements of the mobile application must match the authentication requirements of the web application component. Therefore, it is not possible to authenticate with fewer authentication factors than a web browser.* ***Step 3. Be aware of the Persistent authentication (Remember Me) functionality*** The Persistent authentication feature implemented in the mobile applications should never store the user's password on the device. Also, in mobile apps must be implemented as an option and should not be enabled by default. ***Step 4. Avoid weak token/identifiers/passwords*** Ideally, your mobile app should use a device-specific authentication token that the user can revoke. This will allow the app to minimize unauthorized access to stolen/lost devices. Do not use any spoof-able values for authenticating a user. This

	includes device identifiers or geo-location and if possible, do not allow users to provide 4-digit PIN code for their authentication password. *TIP: Due to requirements for offline use, mobile apps must perform local authentication or authorization checks in their mobile app code. In this case, the developer needs a local integrity check tool for the code to detect unauthorized code changes.*

Role Description

This is an alphabetical list of the roles, abbreviations and list of competencies as defined in Part 5.

	Role	Abbreviation	Competency
1.	Analyst	AN	Knowledge and experience eliciting, specifying and analysing the requirements. Knowledge in designing user interfaces and ergonomic criteria. Knowledge of the revision techniques and experience on the software development and maintenance. Knowledge of the editing techniques and experience on the software development and maintenance.
2.	Customer	CUS	Knowledge of the Customer processes, ability to explain the Customer requirements and experience in the application domain. The Customer (representative) must have the authority to approve the requirements and their changes. The Customer includes user representatives in order to ensure that the operational environment is addressed.
3.	Programmer	PR	Knowledge and/or experience in programming, integration and unit tests. Knowledge of the revision techniques and experience on the software development and maintenance. Knowledge of the editing techniques and experience on the software development and maintenance
4.	Project Manager	PM	Leadership capability with experience making decisions, planning, personnel management, delegation and supervision, finances and software development.
5.	Security Advisor	SA	Knowledge to advise the businesses to identify potential security weaknesses, create security policies, and reduce risks to their IT systems.

6.	Technical Leader	TL	Knowledge and experience in the software process domain.
7.	Work Team	WT	Knowledge and experience according to their roles on the project: TL, AN, DES, and/or PR. Knowledge on the standards used by the Client and/or by the VSE.

Product Description

This is an alphabetical list of the input, output and internal process products, its descriptions, possible states and the source of the product.

	Name	Description	Source
1.	Change Request	It may has the following characteristics: Identifies purpose of change Identifies request status (new, accepted, rejected) Identifies requester contact information Impacted system(s) Impact to operations of existing system(s) defined Impact to associated documentation defined Criticality of the request, date needed by The applicable statuses are: initiated, evaluated and accepted.	Software Implementation Customer Project Management

2.	Project Plan	It Includes the following elements which may have the characteristics as follows:
		- Product Description
		  ○ Purpose
		  ○ General Customer requirements
		- Scope description of what is included and what is not
		- Objectives of the project
		- Deliverables - list of products to be delivered to Customer
		- Tasks, including verification, validation and reviews with Customer and Work Team, to assure the quality of work products. Tasks may be represented as a Work Breakdown Structure (WBS). The task also includes the identification of security requirements, list of assets, threats, information security risk and incidents.
		- *Relationship and Dependence of the Tasks*
		- *Estimated Duration* of tasks
		- *Resources* o perform the project both in
			Project Management

		development and security (humans, materials, equipment and tools) including the required training, and the schedule when the resources are needed.
		- *Composition of Work Team*
		- *Competences pf personal Record*
		- *Role Matrix*
		- *Incidents Response of the project*
		- Schedule of the Project Tasks, the expected start and completion date, for each task.
		- Estimated Effort and Cost
		- Identification of Project Risks
		- Version Control Strategy
		  - Product repository tools or mechanism identified
		  - Location and access mechanisms for the repository specified
		  - Version identification and control defined
		  - Backup and recovery mechanisms defined
		  - Storage, handling and delivery (including archival and retrieval) mechanisms specified
		- Delivery Instructions
		  - Elements required for product release identified (i.e., hardware, software, documentation etc.)
		  - Delivery requirements
		  - Sequential ordering of tasks to be performed
		  - Applicable releases identified
		  - Identifies all delivered software components with version information
		  - Identifies any necessary backup and recovery procedures
		The applicable statuses are: verified, accepted, updated and reviewed.	

3.	Project Repository	It may have the following characteristics: - Stores project work products - Stores released deliverables products - Storage and retrieval capabilities - Ability to browse content - Listing of contents with description of attributes - Sharing and transfer of work products between affected groups - Effective controls over access - Maintain work products descriptions - Recovery of archive versions of work products - Ability to report work products status - Changes to work products are tracked to	Project Management

		Change Requests The applicable statuses are: recovered and updated.	
4.	Requirements Specification	It may have the following characteristics: - Introduction –general description of software and its use within the scope of the customer business; - Requirements description: - Functionality – established needs to be satisfied by the software when it is used in specific conditions. Functionality must be adequate, accurate and safe. - User interface – definition of those user interface characteristics that allow to understand and learn the software easily so the user be able to perform his/her tasks efficiently including the interface exemplar description; - External interfaces – definition of interfaces with other software or hardware; - Security – specification of the software executionlevel concerning the level of security implemented - Reliability – specification of the software execution level concerning the maturity, fault tolerance and recovery; - Efficiency – specification of the software execution level concerning the time and use of the resources; - Maintenance – description of the elements facilitating the understanding and execution of the future software modifications; - Portability – description of the software characteristics that allow its transfer from one place to other; - Design and construction limitations/constraints – needs imposed by the customer; - Interoperability – capability for two or	Software Implementation

		more systems or software components be able to change information each other and use it. - Reusability – feature of any product/sub-product, or a part of it, so that it can be used by several users as an end product, in the own software development, or in the execution of other software products. - Legal and regulative – needs imposed by laws, regulations, etc. Each requirement is identified, unique and it is verifiable or can be assessed. The applicable statuses are: verified, validated and baselined.	
5.	Software User Documentation	It may have the following characteristics: - User procedures for performing specified tasks using the Software - Installation and de-installation procedures - Brief description of the intended use of the Software (the concept of operations) - The supplied and required resources - Needed operational environment - Availability of problem reporting and assistance - Procedures to access and exit the Software - Lists and explains software commands and system-provided messages to the user - As appropriate for the identified risk, it includes warnings, cautions, and notes, with corrections - It includes troubleshooting and error correction procedures. It is written in terms understandable by users. The applicable statuses are: preliminary, verified and baselined.	Software Implementation
6.	Verification Results	It may include the record of: - Participants - Date - Place - Duration - Verification check-list - Passed items of verification - Failed items of verification - Pending items of verification - Defects identified during verification	Project Management Software Implementation
7.	Validation Results	Documents the validation execution, It may include the record of:	Software Implementation

		- Participants - Date - Place - Duration - Validation check-list - Passed items of validation - Failed items of validation - Pending items of validation - Defects identified during validation	

Artefact Description

This is an alphabetical list of the artefacts that could be produced to facilitate the documentation of a project. The artefacts are not required by Part 5, they are optional.

	Name	Description
1.	Weak Authentication Patterns	Describes the weak authentication Patterns that must be avoided in the development of a mobile application.

5. References

Key	Reference
[Code Complete]	Steve McConnell, Code Complete, Second Edition, Redmond, Washington, Microsoft Press, 2004**.**
[Art of Software testing]	Glenford J. Myers, The Art of Software Testing, Second Edition, 2004.
[Practitioner's Guide]	Lee Copeland, A Practitioner's Guide to Software Test Design, 2004
[Defect Prevention]	Marc McDonald, The Practical Guide To Defect Prevention, 2008
[Introduction to Software Testing]	Paul Ammann & Jeff Offutt, Introduction to Software testing, 2008
[Testing Computer Software]	Cem Kaner, Testing Computer Software
[Pratical Software Testing]	Ilene Burnstein, Practical Software Testing, 2002
[SE Support Activities for VSE]	Vincent Ribaud, Software Engineering Support Activities for Very Small Entities, 2010
[Application of ISES in VSE]	Claude Y. Laporte, The application of International Software Engineering Standards in Very Small Enterprises, 2008
[A SE Lifecycle Standard for VSEs]	Claude Y. Laporte, A Software Engineering Lifecycle Standard for Very Small Enterprises, 2008
[Misuse Code Coverage]	Brian Marick, How to Misuse Code Coverage, 1999
[IEEE 1012-2004]	IEEE 1012-2004 IEEE Standard for Software Verification and Validation, IEEE Computer Society
[ISO/IEC 12207]	ISO/IEC 12207:2008 Systems and software engineering – Software life cycle processes.
[ISO/IEC TR 29110-5-1-2]	ISO/IEC TR 29110-5-1-2:2011, Software Engineering—Lifecycle Profiles for Very Small Entities (VSEs) – Part 5-1-2: Management and Engineering Guide – Generic Profile Group - Basic Profile
[ISO/IEC 24765]	ISO/IEC 24765:2010 Systems and software engineering vocabulary
[ISO/IEC 20926]	ISO/IEC 20926:2003 Software engineering -- IFPUG 4.1 Unadjusted functional size measurement method -- Counting practices manual
[ISO/IEC 29881:2008]	ISO/IEC 29881:2008 Information technology--Software and systems engineering--FiSMA 1.1 functional size measurement method,
[IEEE 1233-1998]	IEEE Guide for Developing System Requirements Specifications