

DIPLOMARBEIT

KTI – Kapsch Tablet Infrastructure



Ausgeführt im Schuljahr 2014/15 von

Sebastian Götze	5CHIF
Samuel Hammer	5CHIF
Michael Kaufmann	5CHIF
Konstanze Müller	5CHIF
Philip Steinhäuser	5CHIF

Betreuer / Betreuerin:

DI Harald Swoboda
DI Hannes Färberböck

Kurzfassung

Aufgabenstellung

Die Aufgabe von Kapsch an das Projektteam ist es, verschiedene Softwarelösungen zum Systemschutz von Android-Tablets zu testen. Dazu soll ein Untersuchungsbericht angefertigt werden. In diesem werden die Ergebnisse festgehalten und verglichen, um das beste System für die Kunden von Kapsch zu ermitteln. Ein Prototyp in Form eines Android-Tablets ist ebenfalls zu erstellen. Dieser soll mit dem besten, aus dem Untersuchungsbericht hervorgegangenen, Schutzsystem ausgestattet sein.

Realisierung

Die Firma Kapsch wünscht einen Untersuchungsbericht, aus dem hervorgeht, welcher Systemschutz für Android-Tablets am besten für spezifische Kunden geeignet ist. In diesem Bericht werden 2-3 verschiedene Systeme zur Absicherung eines Android-Tablets verglichen. Darin wird auf die Vor- und Nachteile des jeweiligen Systems eingegangen, sowie auf das Fehlen von Funktionen aufmerksam gemacht. Dies ermöglicht dem Auftraggeber das bestmögliche System für seine Kunden zu ermitteln.

Ergebnisse

Neben dem Untersuchungsbericht wird zusätzlich ein Tablet als Prototyp abgeliefert. Dieses Tablet ist mit dem Schutzsystem ausgestattet, welches aus dem Untersuchungsbericht als am empfehlenswertesten hervorgeht.

Am Ende des Projekts stehen 2 Endprodukte.

Der Untersuchungsbericht

Er vergleicht 2 bis 3 Softwarelösungen zur Absicherung von Tablets. Pro Lösung müssen ihre Vor- und Nachteile vorhanden sein. Sowie auch das Fehlen von Einstellungsmöglichkeiten.

Am Ende des Untersuchungsberichts hat ein Vergleich der Systeme zu stehen, aus dem herausgeht, welches, nach der Meinung des Projektteams, eingesetzt werden sollte. Dies muss mit Argumenten bekräftigt werden.

Der Prototyp

Ein Android-Tablet welches auf die Anforderungen eines Beispielunternehmens zugeschnitten ist. Dabei sollen nur bestimmte Aktionen möglich sein.

- Nutzung von 3 bestimmten Apps
- Kein Zugang zu den Einstellungen
- Keine Verbindung mit einem Computer möglich
- Kein Download von Apps, Fotos, Videos, etc.
- Kein Verlassen der gesicherten Umgebung

Abstract

Assignment of Tasks

Our project partner Kapsch provides enterprise grade solutions for major operating systems such as Microsoft Windows 8. Due to the Consumerization trend in industry platforms like Apple iOS and Android OS are on the rise in enterprise applications and leading system integrators like Kapsch are in need to understand those platforms, their applications and limitations. Based on experience and results from existing projects Kapsch has realized that for certain applications or scenarios the Apple iOS platform has its limitations and drawbacks. In particular industrial applications have a need for a rock-solid platform which enables a system integrator like Kapsch to operate a 24/7 application and service. So Kapsch is keen to expand their knowledge and experience towards the Android platform to be able to provide solution platform for industrial applications. Aim of this cooperation is to co-work on a set of different feasibility concepts for realizing such a platform based on Android.

Realisation

The company Kapsch wants an investigation report, which shows which system protection is suitable for Android tablets best for specific customers. In this report, 2-3 different systems to secure a Android tablets are compared. It takes into account the advantages and disadvantages of each system, as well as draw attention to the lack of features. This allows the customer the best possible system for its customers to determine.

Results

In addition to the investigation report a tablet is delivered as a prototype. This tablet is equipped with the protection system, which is apparent from the investigation report as most recommendable.

At the end of the project, there are 2 final results:

The Research Paper

It compares 2 to 3 software solutions for securing tablets. For each Solution, advantages and disadvantages must be present. As well as the lack of configuration options.

At the end of the investigation report, a comparison of the systems has to stand, out of the proceeds, which should be in accordance with the opinion of the project team used. This must be confirmed with arguments.

The Prototype

An Android tablet that is tailored to the requirements of a model company. It should be possible only certain actions.

- Use of 3 specific apps
- No access to the settings
- No connection to a computer possible
- No downloading of apps, photos, videos, etc...
- Leaving the secure environment not possible

Inhaltsverzeichnis

1 Einleitung	1
1.1 Motivation	1
1.2 Projektpartner	2
1.3 Ursprungsproblem	3
1.4 Vorgehensweise	3
1.4.1 Vorbereitung	3
1.4.2 Planung	3
1.4.3 Durchführung	4
1.4.4 Ergebnis	4
1.4.5 Organigramm	4
1.5 Arbeitsaufteilung	7
1.5.1 Sebastian Götze	7
1.5.2 Samuel Hammer	7
1.5.3 Michael Kaufmann	8
1.5.4 Konstanze Müller	8
1.5.5 Philip Steinhäuser (Projektleiter)	9
2 Studie	10
2.1 Android	10
2.1.1 Architektur	10
2.1.2 Security	11
2.1.2.1 System- und Kernel Security	11
2.1.3 Angriffsvektoren	12
2.1.3.1 Social Engineering	12
2.1.3.2 Drive-by Exploitation	12
2.1.3.3 Phishing	13
3 Varianten	14
3.1 Linux Manipulation	14
3.1.1 Allgemein	14
3.1.2 Schlussfolgerung	14
3.2 Mobile Device Management (MDM)	16
3.2.1 Allgemein	16
3.2.2 Mobile Device Management Standard	16
3.2.3 Informationen der getesteten Software	17
3.2.4 Installation	17
3.2.5 Features	18
3.2.5.1 Key-Features	18
3.2.5.2 Zusatzfeatures	19
3.3 MDM + Container	20
3.3.1 MobileIron Containertechnologie	20
3.3.2 Informationen der getesteten Software	20

3.3.3	Installation	20
3.3.4	End User Products	20
3.3.4.1	Docs@Work	20
3.3.4.2	Web@Work	21
3.3.4.3	Apps@Work	22
3.4	Samsung Knox	23
3.4.1	Informationen der getesteten Software	23
3.4.2	Samsung Knox Bestandteile	23
3.4.2.1	Platform Security	23
3.4.2.1.1	Kerntechnologien der Platform Security	24
3.4.2.2	Application Security	25
3.4.2.2.1	Application Containers	25
3.4.2.2.2	On-Device Data Encryption (ODE)	25
3.4.2.2.3	Virtual Private Network Support (VPN)	25
3.4.2.3	Management	26
3.4.2.3.1	Knox Lösungen	26
3.4.2.3.2	MDMs	27
4	Auswahl & Konzept	28
4.1	Endergebnis und Empfehlung	28
4.2	Evaluierung	28
4.3	Auswertung der Evaluierung	30
4.3.1	Allgemein	30
4.3.2	Inhaltseinstellungen	30
4.3.3	Statistik	30
4.3.4	Benutzer und Geräte	30
4.3.5	Managementseitige Anforderungen	30
4.3.6	Zusätzliche Anforderungen	30
4.4	Empfehlung	31
5	Lessons Learned	32
5.1	Sebastian Götze	32
5.2	Samuel Hammer	32
5.3	Michael Kaufmann	33
5.4	Konstanze Müller	33
5.5	Philip Steinhäuser	33
6	Danksagung	34
7	Begleitprotokoll	35
	Quellenverzeichnis	37
	Abbildungsverzeichnis	38
	Tabellenverzeichnis	39
8	Anhang	40

Einleitung

1.1 Motivation

Im zunehmenden Maße finden sich mobile Endgeräte (z.B. Tablet) als Marketing, Informations- und Imageelement, aber auch als Teil von Prozessen im Retail- als auch im Industrieumfeld wieder. Diese ersetzen oder erweitern bestehende Lösungen (z.b. Kiosk, Digital-Signage, Laptops, Industrie PCs) um interaktive Elemente und unterstützen so Unternehmen in ihren Geschäftsprozessen. Bei der eingesetzten Lösung ist es aus Sicht des Retail und Industrie-Kunden wichtig die Handhabung und das Erlebnis des Gerätes für den Anwender zu erhalten. Die Einsatzgebiete reichen dabei von Tablets in Produktionsprozessen zu Zwecken der Qualitätssicherung (eigene Applikationen mit Dokumentationsfunktionen) bis hin zu Multimedia-Terminals im stationären Handel. Um diese verschiedenen Szenarien realisieren zu können ist es notwendig eine stabile, sichere und wartbare Plattform zu konstruieren, die zum einen die Anwendung in unterschiedlichen Einsatzgebieten ermöglicht und zum anderen für den Betriebsführer leicht bereitzustellen und zu warten ist.

Um den hier angeführten allgemeinen Anforderungen ihrer Kunden gerecht zu werden, benötigt Kapsch die passendste Softwarelösung im Bereich Security für die hierfür in Betracht gezogenen Tablets. Um diese Softwarelösung zu ermitteln, wurde ein Projektteam des fünften Jahrgangs der HTBLVA Spengergasse, Fachbereich Informatik mit der Aufgabe der Eruierung dieser Lösung und der Erstellung eines Konzepts betraut. Das Projekt KTI – Kapsch Tablet Infrastructure wird von dem Projektmanager Philip Steinhäuser geleitet und besteht aus den Teammitgliedern Sebastian Götze, Samuel Hammer, Kaufmann Michael und Konstanze Müller. Das Projektteam steht in engem Kontakt mit dem Project-owner welcher mit den Mitarbeitern Bernhard Bruckner und Jürgen Krammer vertreten ist.

Kapsch unterstützt das Projekt mit diversen Hilfeleistungen sowie Technischer Support, Intellektuelle Hilfe und finanzielle Unterstützung beim Kauf des Tablets, welches nach Beendigung des Projekts wieder in den Besitz von Kapsch übergehen wird.

1.2 Projektpartner



Abbildung 1.1: Kapsch Logo

Die Kapsch Group setzt sich aus 3 Hauptunternehmen zusammen:

- Kapsch TrafficCom
- Kapsch CarrierCom
- Kapsch BusinessCom

Die Kapsch TrafficCom ist internationaler Anbieter von Technologien, Lösungen und Dienstleistungen für den Intelligent Transportation Systems (ITS) Markt.

Die Kapsch CarrierCom ist globaler Lösungspartner für Telco-Carrier und Communication-Provider sowie Railway-Operator.

Unser konkreter Projektpartner ist jedoch die Kapsch BusinessCom. Mit 1.400 Mitarbeitern, einem Umsatz von knapp 300 Millionen Euro und Niederlassungen in Österreich, Tschechien, Slowakei, Ungarn, Rumänien und Polen, positioniert sich das Unternehmen als einer der führenden ICT-Servicepartner in Zentral- und Osteuropa. Kapsch setzt auf Partnerschaften mit führenden Branchengrößen wie Apple, Cisco, Google, HP oder Microsoft, um seinen rund 17.000 Kunden den bestmöglichen Service gewährleisten zu können.

1.3 Ursprungsproblem

Die Aufgabe von Kapsch an das Projektteam ist es, verschiedene Softwarelösungen zum Systemschutz von Android-Tablets zu testen. Die vom Projektowner zum Test gewünschten Softwarelösungen sind.

- MDM
- MDM + Container
- Samsung Knox

Zusätzlich bestünde noch die Möglichkeit einer **Linux Manipulation**, welche jedoch für Kapsch aus rechtlichen Gründen nicht in Frage kommt, da diverse Garantieverletzungen auftreten würden und zusätzlich enorme Kosten anfallen würden aufgrund von Hohen Entwicklungskosten und langen Entwicklungszeiten bis das System einwandfrei funktionieren würde.

Zu diesen Schutzsystemmöglichkeiten soll ein Untersuchungsbericht angefertigt werden. In diesem werden die Ergebnisse festgehalten und verglichen, um das beste System für die Kunden von Kapsch zu ermitteln.

Auf Basis dieses Untersuchungsberichts soll dann die Passendste Softwarelösung ausgewählt werden, welche in ein Konzept eingebaut wird. Ein Teil dieses Konzepts beinhaltet die Implementierung der ausgewählten Softwarelösung auf ein Tablet welches dann als Prototyp deklariert wird.

Der Untersuchungsbericht in Kombination mit dem Prototyp ist das Ergebnis was über den Ausgang dieses Projekts entscheidet.

1.4 Vorgehensweise

1.4.1 Vorbereitung

- Vorbereitungsmeeting mit Kapsch
- Einlesen in Technologie
- Erste Recherchen

1.4.2 Planung

- Projektantrag
- Vorstudie
- PSP (Projektstrukturplan)
- OSP (Objektstrukturplan)
- Timetable bzw. Gantt-Chart
- Lastenheft
- Pflichtenheft
- Projekthandbuch
- Checklist Template
- Research Template
- Untersuchungsbericht
- Diplomarbeit

1.4.3 Durchführung

- Recherche
- Erstellung des Untersuchungsberichtes
- Auswahl der passenden Lösung
- Konzept erstellen
- Ausgewählte Lösung in Konzept einarbeiten
- Konzept teilweise umsetzen
 - Konfiguration der Lösung auf Tablet (Prototyp)

1.4.4 Ergebnis

1. Untersuchungsbericht
2. Prototyp

1.4.5 Organigramm

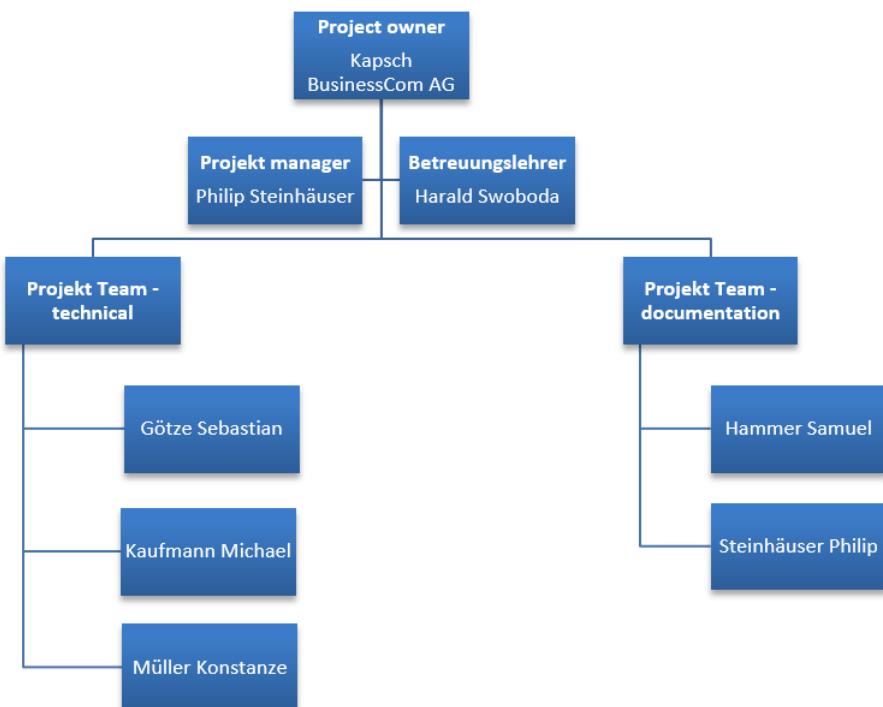


Abbildung 1.2: Projekt-Organigramm

Die Aufgaben des Projekts wurden ab diesem Zeitpunkt so verteilt das das Dokumentationsteam sämtliche Projektmanagementaufgaben und das Technik Team alle Research- und Testungsaufgaben übernimmt.

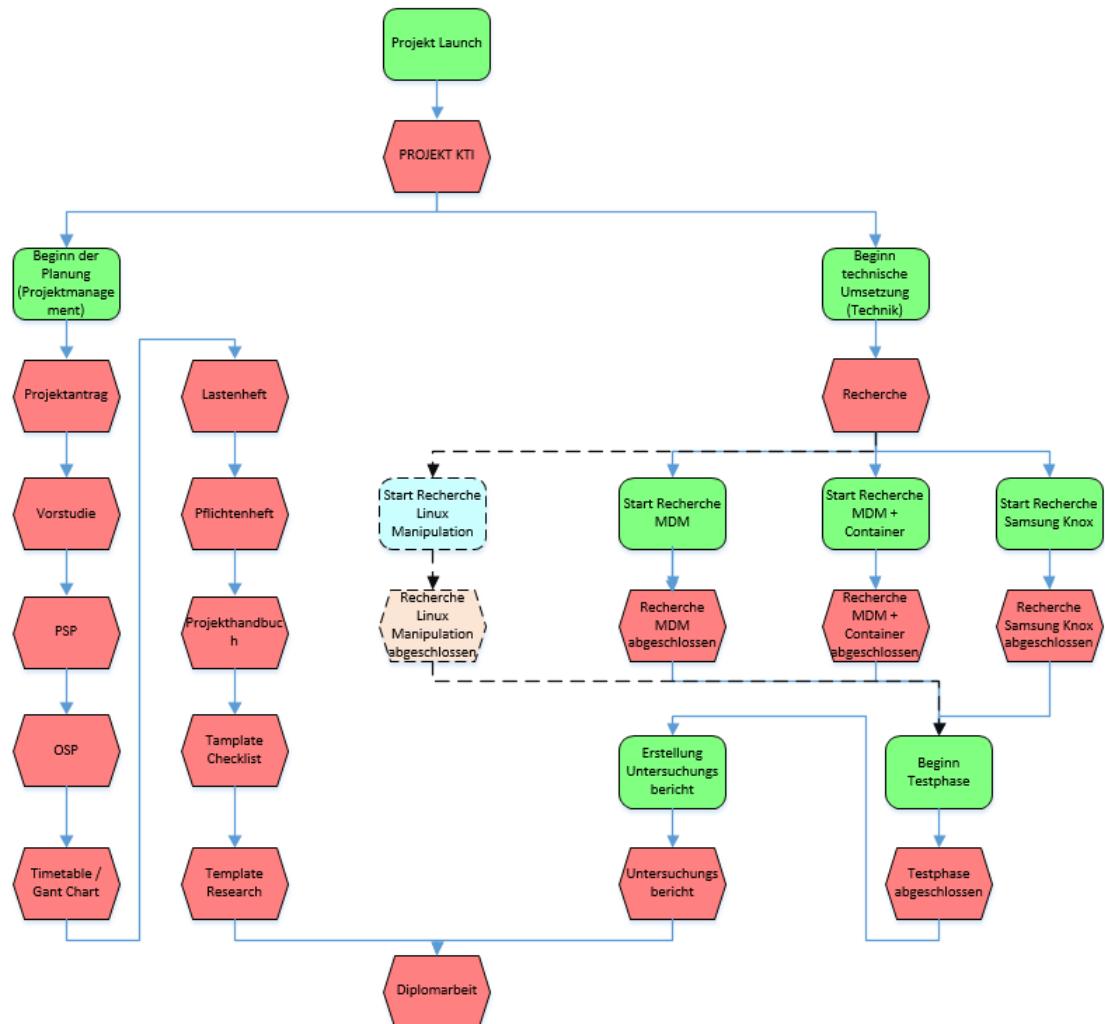


Abbildung 1.3: Aufgabenverteilung - Diagramm

Dieses Ablaufdiagramm zeigt auf der einen Seite die Reihenfolge der erstellten Dokumente, so wie beispielsweise, dass der OSP und der Timetable / Gant Chart auf dem PSP aufbaut, da dieser zuvor erstellt wurde und diese drei Dokumente konsistent sein müssen.

Auf der Technik Seite wiederum sieht man das zuerst eine umfangreiche Recherche nötig ist bzw. war, bevor die Wahl der passendsten Softwarelösung getroffen werden kann / konnte, welche dann schlussendlich für das Konzept und in weiterer Folge für die Konfiguration des Prototyps verwendet werden konnte / wurde.

Wenn man sich unsere Vorgehensweise nun Schritt für Schritt auf die beiden Sub Teams aufgeteilt anschauen würde, würde folgendes Bild entstehen:

#	Dokumentation	Technik
1	<ul style="list-style-type: none"> • Erstellung des Projektantrages • Erstellung der Vorstudie • Erstellung PSP • Erstellung OSP 	-
2	<ul style="list-style-type: none"> • Erstellung des Gantt-Chart • Erstellung Lastenheft • Erstellung Pflichtenheft 	<ul style="list-style-type: none"> • Beginn der Recherchephase <ul style="list-style-type: none"> – Recherche Linux Manipulation – Recherche MDM – Recherche MDM+Container – Recherche Samsung Knox
3	<ul style="list-style-type: none"> • Erstellung Projekthandbuch • Erstellung Research Template • Erstellung Checklist Template 	<ul style="list-style-type: none"> • Festhalten der Rechercheergebnisse in Templates • Recherchephase abgeschlossen <ul style="list-style-type: none"> – Recherche Linux Manipulation abgeschlossen – Recherche MDM abgeschlossen – Recherche MDM+Container abgeschlossen – Recherche Samsung Knox abgeschlossen
4	<ul style="list-style-type: none"> • Erstellung Diplomarbeit 	<ul style="list-style-type: none"> • Beginn der Testphase • Erstellung des Untersuchungsberichts
5	<ul style="list-style-type: none"> • Fertigstellung Diplomarbeit • Fertigstellung Projekthandbuch 	<ul style="list-style-type: none"> • Abschluss der Testphase • Fertigstellung des Untersuchungsberichts

Tabelle 1.1: Projektablauf

1.5 Arbeitsaufteilung

1.5.1 Sebastian Götze

- Projektantrag
- Vorstudie
 - status quo
 - MDM
- Template Checklist
- Template Research
- Pflichtenheft
 - Unternehmensperspektive
 - Funktionale Beschreibung
- Projekthandbuch
 - Project Assignment
 - Plan of objects of consideration
 - Project Work-Package Specification
 - Project Rules
- Gantt-Chart
- Mobile Device Management (MDM)
 - Recherche
 - Testing
 - Untersuchungsbericht
 - Diplomarbeit
- Linux Manipulation
 - Recherche
 - Untersuchungsbericht
 - Diplomarbeit
- Prototyp

1.5.2 Samuel Hammer

- Projektantrag
- Vorstudie
 - Critical Appraisal
 - Android Technology Analysis
- Projekthandbuch
 - Project Assignment
 - Description of Pre- and Post-Project Phase
 - Project Work-Package Specification
 - Project Bar Chart
 - Minutes - Project Co-Ordination
- Pflichtenheft
 - Funktionen
 - Betriebsbedingungen
 - Härtung der mobilen Endgeräte
- Besprechungsprotokolle
- Präsentationen
 - Statuspräsentationen
 - * 09.12.2014
 - * 08.01.2015
 - * 19.02.2015
 - * 19.03.2015
 - * 16.04.2015
 - Abschlusspräsentation Englisch
 - Abschlusspräsentation Deutsch
- Gantt-Chart
- Diplomarbeit

1.5.3 Michael Kaufmann

- Projektantrag
- Vorstudie
 - Samsung Knox
- Projekthandbuch
 - Project assignment
 - Project Objectives
 - Project Work-Package Specification
- Pflichtenheft
 - System Architecture
 - * Hardware
 - * Software
- Gantt-Chart
- Samsung Knox
 - Recherche
 - Testing
 - Untersuchungsbericht
 - Diplomarbeit
 - Implementierung
- Prototyp

1.5.4 Konstanze Müller

- Projektantrag
- Vorstudie
 - MDM + Container
- Projekthandbuch
 - Project Assignment
 - Project environment Graphic
 - Project Environment Table
 - Project Work-Package Specification
 - Project Risk Analysis
 - Project Status Reports
- Pflichtenheft
 - Target Groups
 - Quality Criteria
- Besprechungsprotokolle
- Statusberichte
- Mobile Device Management (MDM) + Container
 - Recherche
 - Testing
 - Untersuchungsbericht
 - Diplomarbeit
- Prototyp

1.5.5 Philip Steinhäuser (Projektleiter)

- Projektantrag
- Vorstudie
 - Management Summary
 - Task
 - SWOT - Analyse
- Gantt-Chart
 - laufende Aktualisierung und Projektfortschrittskontrolle
- Kooperationsvereinbarung
- Meilensteinplan
- Meilensteintrendanalyse
- Objektstrukturplan - OSP
- Projektstrukturplan - PSP
- Pflichtenheft
 - Requirement Specification
 - Embedding in Organisation
 - Milestones
- Projekthandbuch
 - Project Assignment
 - Project Organisation Chart
 - Project Organisation
 - Work Breakdown Structure (WBS)
 - Project Work-Package Specification
 - Milestoneplan
 - Project Communication
 - Project Responsibility Matrix
 - Project Documentation
 - Project Co-Ordination
 - Project Status Report
 - Laufende Aktualisierung und Ausarbeitung des einzelnen Punkte des Projekt-handbuchs
- Statusberichte
- Abnahmeprotokoll (Acceptance Testing Protocol - ATP)
- Diplomarbeit

Studie

2.1 Android

Android ist, im Gegensatz zu anderen mobilen Betriebssystemen auf dem Markt, eine offene Plattform. Das kommt im Wesentlichen daher, dass Android auf einem open source System, dem Linux Kernel, aufbaut. Durch die freie Verfügbarkeit des Codes, gibt es im Android-Bereich eine große Developer Szene die laufend eigens modifizierte Betriebssysteme (ROMs) und Apps hervorbringt. Ein so hoher Grad an Offenheit birgt jedoch auch gewisse Sicherheitsrisiken und öffnet Angriffsvektoren. Um diese Gefahren besser zu verstehen sollte man sich eingehend mit der Android Architektur und dem darin enthaltenen Sicherheitskonzept befassen.

2.1.1 Architektur

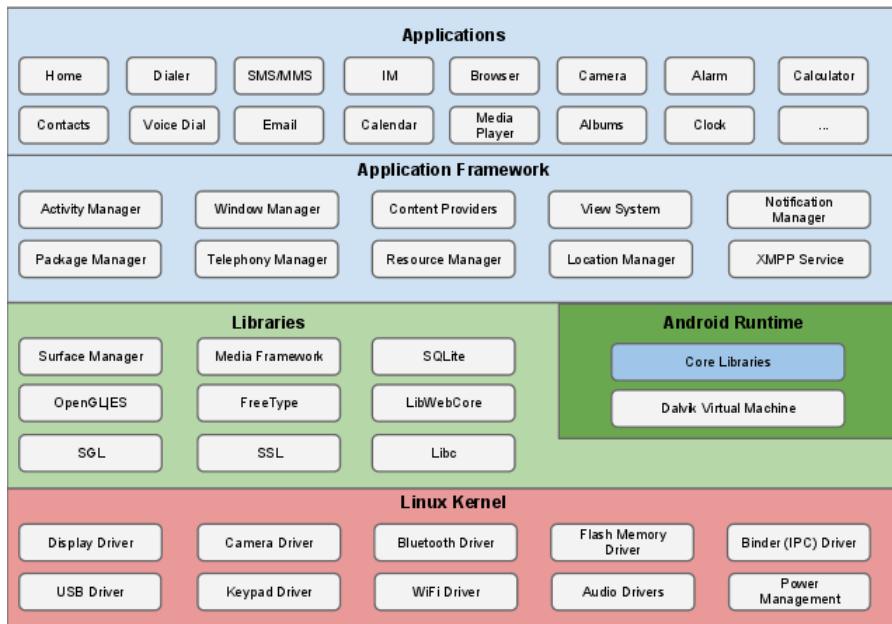


Abbildung 2.1: Android Systemarchitektur

Wie in Abbildung 1 zu sehen ist, bildet der Linux Kernel die unterste Schicht der Architektur. Auf ihm baut das gesamte Betriebssystem mitsamt aller Sicherheitskonzepte auf. Der Kernel stellt die Brücke zwischen Hard- und Software dar und enthält die Treiber für diverse andere Komponenten eines Smartphones wie beispielsweise Modem, GPS Empfänger, Kamera, etc...

In der darüber liegende Ebene sind die Android Libraries sowie die Runtime, also die „Dalvik Virtual Machine“ zu finden.

Android Apps sind meist in Java Programmiert und werden jeweils in einer eigenen Virtuellen Maschine ausgeführt, der DVM¹. Diese Systematik ermöglicht das logische, parallele, unabhängige Ausführen von verschiedenen Apps mit verschiedenen Benutzerrechten. Die Libraries steuern und kontrollieren im Wesentlichen die Funktionen des Kernels und sind für die zentralen Funktionalitäten auf niedriger Ebene zuständig.

Die nächst-höhere Ebene bildet das Application Framework. Hier befinden die Grundfunktionen von Android wie zum Beispiel Telefonie, Location Services, Window Manager, Notification Manager, etc... Auf dieser Ebene werden Entwicklern äußerst umfangreiche APIs für das Entwickeln von Benutzeranwendungen (Apps) zur Verfügung gestellt.

Die oberste Schicht in der Android-Systemarchitektur sind die Applikationen welche der Benutzer selbst installiert und auf den darunterliegenden Schichten aufbaut.

Dieser gesamte „Stack“ (=Stapel) wird in einem ROM zusammengefasst und auf ein Smartphone installiert. Durch die zuvor erwähnte Offenheit des Systems können so durch Modifizieren eines ROM Paketes stark angepasste Versionen von Android entwickelt und verwendet werden.

2.1.2 Security

Android wird als Betriebssystem mittlerweile auf ca 84% aller Smartphones weltweit eingesetzt.² Ein Betriebssystem mit einem so großen Marktanteil erfordert ein durchdachtes und ausgereiftes Sicherheitskonzept. Die drei grundlegenden Sicherheitsobjekte sind:

- Schützen von Benutzerdaten
- Schützen von Systemressourcen (inklusive Netzwerk)
- Bereitstellen von Applikationsisolation

Um diese Objekte zu erreichen, stehen eine Reihe von Security-Features zur Verfügung

- Robuste Security auf der OS Ebene durch den Linux Kernel
- Erforderliche Sandbox für alle Apps
- Sichere interprozess Kommunikation
- Application signing
- Application-defined and user-granted permissions

2.1.2.1 System- und Kernel Security

Der Linux Kernel stellt das Fundament von Android dar. Durch dessen langjährige Wartung und Weiterentwicklung hat dieser sich zu einem äußerst sicheren und weit verbreiteten Kernel entwickelt, welcher auch in Sicherheitsempfindlichen Umgebungen eingesetzt wird.

Der Linux Kernel stellt einige der wichtigsten Sicherheitsfunktionen zur Verfügung.

- Ein Benutzer-basiertes Rechtemodell
- Prozessisolation
- Mechanismus für sichere IPC³
- Die Möglichkeit potentiel gefährliche Teile des Kernels zu entfernen

¹Dalvik Virtual Machine

²Quelle: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp> Stand Q3 2014

³Interprozesskommunikation

Im Gegensatz zum normalen Linux, wo mehrere Anwendungen mit demselben Benutzer ausgeführt werden, weist Android jeder Applikation eine eigene User ID zu und führt diese mit ebendiesem User in einem separaten Prozess aus. Diese Vorgehensweise resultiert in einer „Kernel-level Application Sandbox“. Dadurch kann eine App von Haus aus mit keiner anderen App kommunizieren sofern dies nicht explizit erwünscht ist. Standardmäßig können Apps auch nicht auf Dinge wie Standort, die Telefonie-Funktion, etc. zugreifen sondern müssen erst um die Erteilung der entsprechenden Benutzerrechte ansuchen. Die Tatsache dass diese Sandbox-Systematik auf Jahrzehnte alter UNIX Technologie aufbaut, macht das System leicht überwachbar und transparent, jedoch gleichzeitig effizient.

Speicherfehler führen in vielen Betriebssystemen zu groben Sicherheitsrisiken und sind in der Lage die Sicherheit eines Gerätes komplett zu kompromittieren. Bei Android kann ein solcher Speicherfehler durch das Sandboxing aller Applikationen auf OS-Level keinen allzu großen Schaden anrichten, da eventueller Schadcode nur im Kontext einer bestimmten App, mit eingeschränkten Benutzerrechten ausgeführt werden kann.

Selbstverständlich ist auch die Applikations-Sandbox nicht zu 100% sicher, möchte man diese aber knacken, so muss man es schaffen den gesamten Linux Kernel zu komromittieren.

2.1.3 Angriffsvektoren

Angriffsvektoren sind Pfade um ein Security Konzept zu durchdringen und so Zugriff auf gesicherte Ebenen eines Systems zu erlangen. Gerade bei einem Mobilen Betriebssystem wie Android, ist es von besonderer Wichtigkeit diese Gefahrenquellen zu kennen und soweit wie möglich zu eliminieren da Geräte welche auf Android laufen einen Großteil der Zeit angeschaltet und mit dem Internet verbunden sind.

Ein weiterer Grund warum besonders mobile Endgeräte Ziel solcher Attacken sind ist die Tatsache dass Daten auf einem Smartphone oder Tablet meist aktuell sind. So werden beispielsweise neue E-Mails immer sofort via push mail heruntergeladen und gespeichert. Auch sogenannte TAN Codes, welche sich in den letzten Jahren als Autorisierung-Methode für online Überweisungen etabliert haben und via SMS übertragen werden, machen Smartphones zu einem brauchbaren Ziel für Hacker.

2.1.3.1 Social Engineering

Social Engineering ist eine Methode unautorisiert an Daten zu gelangen, ohne dabei Technische Änderungen am System vorzunehmen. Hierbei wird versucht den User mit Hilfe von Falschen Informationen dazu zu bringen das System zu verändern oder eine schädliche Datei herunter zu laden und dem Angreifer so selbst den Weg zu ebnen. Ein Beispiel für Social Engineering ist der sogenannte „ILOVEYOU Virus“. Dieser wurde via E-Mail verschickt und gab vor ein Liebesbrief zu sein, sodass der Benutzer ihn öffnet und selbst ausführt. Gegen das Prinzip des Social Engineerings gibt es kein brauchbares Gegenmittel, da es einzig und allein auf den Entscheidungen des Users aufbaut.

2.1.3.2 Drive-by Exploitation

Diese Angriffsmethode macht sich Bugs in vorhandener Software zu Nutzen um schadhafoten Code auf dem Device auszuführen. Mit Hilfe solcher Bugs gelingt es dem Angreifer Code über beispielsweise eine Website auf das Gerät des Opfers herunterzuladen und dann auf dem Zielsystem auszuführen. Drive-by Exploitation ist vor allem auf Desktop PC Systemen weit verbreitet. Durch diverse Sicherheitsmechanismen von Android wie das Sandboxing der Apps oder die DVM (Dalvik Virtual Machine), ist es auf solchen Geräten

wesentlich schwieriger schadhaften Code auszuführen und so das System nachhaltig zu beschädigen. Nichtsdestotrotz gibt es auch auf Android eine große Anzahl an Apps welche native Software-Bibliotheken verwenden und so angreifbar gegen Drive-by Exploitation sind.

2.1.3.3 Phishing

Phishing Attacken werden benutzt um sensible Informationen wie Login Daten abzugreifen. Der Angreifer gibt sich dabei als offizielle Instanz aus und bewegt den Nutzer dazu seine Daten preiszugeben. Dies geschieht beispielsweise über eine gefälschte Website, welche gleich aussieht wie die echte Website, nur dass die eingegebenen Daten direkt an den Angreifer weitergeleitet werden. Phishing beinhaltet einige Aspekte des Social Engineerings, daher gibt es auch hier keine wirklich gute Möglichkeit solche Attacken zu verhindern. Auf Mobilen Geräten ist die Situation allerdings ein wenig anders, da es für die meisten online Services eine eigene App gibt, welche nur über den offiziellen App Store heruntergeladen werden kann. Daher verwenden nur wenige User die Website und laufen damit nicht Gefahr Ziel einer Phishing Attacke zu werden.

Varianten

3.1 Linux Manipulation

3.1.1 Allgemein

Dieser Teil befasst sich mit der Veränderung des Grundsystems eines jeden Android Geräts. Dieses Grundsystem basiert auf dem Open Source Betriebssystem Linux und existiert dabei in einer für Mobilgeräte angepassten Form. In seiner Standardimplementation bietet es zwar einige Funktion zur Erhöhung der Gerätesicherheit, jedoch nicht genügend, um es in einem betrieblichen Umfeld einzusetzen. Da Linux ein Open Source System ist, darf der Source Code von jedem angesehen und nach eigenen Wünschen verändert werden. Und genau hier setzt die Methode der Linuxmanipulation an. In dem man den Source Code so verändert, dass bestimmte Teile des Betriebssystems unzugänglich gemacht oder verschlüsselt werden, ist es möglich den späteren Benutzer vor unabsichtlichen Änderungen am System zu bewahren, welche den reibungslosen Betrieb stören könnten. Das bedeutet, dass dadurch ein vollkommen an die Bedürfnisse des Kunden angepasstes Betriebssystem möglich wäre. Um sich einen besseren Eindruck davon zu verschaffen, wie diese Manipulationen letztendlich aussehen, lohnt es sich auf die diversen frei am Markt erhältlichen Derivate zu werfen. Bekannte Beispiele dafür wären:

- CyanogenMod
- Android AOSP
- Paranoid Android
- Dirty Unicorns
- etc.

Zwar sind diese nicht mit securitytechnischen Absichten entwickelt worden, aber sie zeigen trotzdem auf was mit einer Menge an Entwicklungsaufwand möglich ist. Da aber durch die tiefgreifenden Eingriffe in das System eines Android Gerätes auch die Garantieansprüche verloren gehen, ist das Projektteam zu einem schnellen Fazit zu kommen.

3.1.2 Schlussfolgerung

Die Methode der Linuxmanipulation ist für die Zwecke der Kapsch leider absolut nicht einsetzbar da sich bei ihr gewisse Konflikte bezüglich Garantieanspruch und Kosten ergeben. Zwar wären über diese Methode sämtliche Anforderungen an das Endprodukt erfüllbar, jedoch bedarf es dazu eines so großen Entwicklungsaufwands, dass dieser sich in einem dermaßen hohen Endkundenpreis niederschlagen würde, welcher von kaum einem Unternehmen zu bezahlen wäre. Denn nicht nur die Beschäftigung einer Vielzahl von Entwicklern über einen langen Zeitraum hinweg, sondern auch der anfallende Support für das Produkt würde sich selbst für ein großes Unternehmen wie Kapsch nicht rentieren. Ein weiteres großes Manko dieser Methode ist die verfallende Garantie für die Hardware. Egal mit

welchen Android Tablet diese Software ausgeliefert werden würde, sobald eine Veränderung der darauf vorinstallierten Software stattfindet, gehen sämtliche Garantieansprüche an den Hersteller verloren. Bei der geplanten Menge an ausgelieferten Geräten durch die Kapsch, wäre dies nicht vertretbar. Für einen industriellen Einsatzzweck ist die Variante daher absolut nicht geeignet und der damit einhergehende Aufwand würde sich nur durch einen extrem hohen Verkaufspreis ausgleichen lassen. Somit bleibt dem Projektteam nichts anderes zu sagen, als, dass diese Variante nicht passend für die Absichten der Kapsch ist.

3.2 Mobile Device Management (MDM)

3.2.1 Allgemein

Die folgenden Zeilen beschäftigen sich mit den Einsatz von Mobile Device Management Systemen als Betriebsplattform für potentielle Kunden der Firma Kapsch. Durchgeführt wurden alle Untersuchungen am bereits am Markt etablierten MDM-System MobileIron. Es wird beleuchtet was der MDM-Standard ist, welche Funktionen für den alltäglichen Gebrauch unumgänglich sind und wie diese im System realisiert sind. Ein besonders wichtiger Punkt hierbei ist auch das Aufzeigen von nicht vorhandenen Funktionen, die jedoch für das Unternehmen Kapsch von fundamentaler Wichtigkeit sind. Des Weiteren wird auf die Bedienbarkeit und die Komplexität der Installation eingegangen und inwiefern dies für den Projektauftraggeber und dessen Kunden relevant ist. Abschließend wird ein Statement abgegeben, ob bzw. wie es möglich ist diese Form von System für die von Kapsch gedachten Zwecke einzusetzen.

3.2.2 Mobile Device Management Standard

MDM bezeichnet einen von der Open Mobile Alliance (OMA) festgelegten industriellen Standard zur Verwaltung mobiler Endgeräte wie zum Beispiel Smartphones, Tablets oder Laptops. Es dient dazu die allgemeine Verwaltung einer Vielzahl von Geräten zu erleichtern und somit Zeit und Kosten zu sparen. Die mobile Hardware kann dabei vom Unternehmen zur Verfügung gestellt werden oder, sofern mit dem Mitarbeiter abgesprochen, von diesem selbst mitgebracht werden. „Bring Your Own Device“ (BYOD) nennt sich dieser Ansatz. Dieser Standard wird in der Software verschiedenster Hersteller implementiert, welche dann dieses Komplettsystem verschiedenen Unternehmen zur Verwaltung ihrer Geräte anbieten. Beispiele dafür sind:

- MobileIron
- Samsung EMM
- Cisco Meraki
- MaaS360
- AirWatch
- etc.

Bestandteile dieser Softwarelösung sind eine Serverkomponente und die verschiedenen mobilen Clients. Der Server dient dabei dazu die Konfigurationen und Statistiken für die Geräte zu halten und zu verwalten. Er sendet auch, auf dem MDM-Standard basierende, Management Kommandos an die Clients aus, wenn sich ein Parameter in deren Konfiguration verändert hat. War es am Anfang der MDM-Systeme noch notwendig das Gerät dazu physisch mit dem Server zu verbinden, geschieht dies heute vollautomatisch über Netzwerkverbindungen. Die implementierten Funktionen können zum Beispiel eine over-the-air (OTA) Verteilung von Applikationen, Daten oder Konfigurationen sein. So braucht ein Administrator nicht auf 100 Geräten das Wifi-Netzwerk einrichten, sondern kann per Knopfdruck diese Konfiguration auf alle im System registrierten Geräte verteilen. Auch in Punkto Sicherheit bieten MDM-Systeme einige Möglichkeiten und deshalb sind sie so interessant für die Zukunftspläne der Firma Kapsch. So bieten diese Systeme die Möglichkeit Passwortrichtlinien zu setzen oder sogar ganze Teile des Betriebssystems zu sperren, damit diese für den Benutzer nicht zugänglich sind. Dadurch soll die Anfälligkeit für Fehler im Berufsumfeld gesenkt und ein ordentlicher Arbeitsablauf genehmigt werden.

Name	MobileIron EMM
Hersteller	MobileIron, 415 East Middlefield Road, Mountain View, CA 94043
Version	7.5.0
Datum	27.01.2015
Preis	/
Website	http://www.mobileiron.com/
Dokumentation	https://support.mobileiron.com/eval/

Tabelle 3.1: MDM Übersicht

3.2.3 Informationen der getesteten Software

3.2.4 Installation

Die Installation von MobileIron gestaltet sich relativ einfach, wobei doch einige wichtige Dinge zu beachten sind. Nach dem Download einer Datei aus dem Online-Zugangsportal von MobileIron, kann über diese das Betriebssystem installiert werden. Diese gestaltet sich für erfahrene Nutzer sehr einfach, allerdings ist auf folgende Dinge acht zu geben:

Folgende Daten müssen bereit stehen bzw. eingerichtet werden:

- Lizenzerinformationen (Firma, Kontaktperson, E-Mail)
- IP-Adresse
- Externer Hostname (**Sehr wichtig, weil die mobilen Geräte den Server von außerhalb erreichen müssen**)
- Command Line Interface Passwort
- Administratorname und -passwort
- Mind. 1 physikalisches Interface
- Subnetzmasks
- Default Gateway
- Mind. 1 zu erreichender DNS-Server
- Wahlweise
 - SSH-Zugriff
 - Telnet Zugriff
 - NTP

Ist die Einrichtung erfolgt, kann man das System nach einem Neustart bereits einsetzen. Während dem Evaluierungsprozess sind dem Projektteam allerdings einige wichtige Details aufgefallen. Ein funktionierender externer Hostname ist von höchster Wichtigkeit, weil ohne ihn zwar die Einrichtung der Software auf den mobilen Endgeräten funktioniert, leider jedoch die Verbreitung von Konfigurationen versagt. Da jedoch 99 Prozent aller modernen Unternehmen über solche Möglichkeiten verfügen sollten, dürfte dies im realen Betrieb weniger problematisch ausfallen. Hervorzuheben ist hierbei die hervorragende Dokumentation, die MobileIron für den Installationsprozess zur Verfügung stellt. Auf deren Website findet sich eine Sammlung an Dokumenten, welche den Administrator am Anfang zwar überwältigen könnten, aber sich als eine schnell zu durchforstende Sammlung an bebilderten Skripten zur Einrichtung sämtlicher Funktionen herausstellen. Generell lässt sich die Webplattform, welche MobileIron hier zur Verfügung stellt, gut bedienen und bietet Informationen zu den verschiedenen Implementierungsszenarien und Komponenten des Systems. So findet man sich nach kurzer Zeit bereits relativ gut zurecht und weiß wo man suchen muss, um die benötigte Information zu finden.

3.2.5 Features

3.2.5.1 Key-Features

In diesem Teil werden die wichtigsten von MobileIron gebotenen Features beleuchtet und erklärt. Die folgende Liste stellt die wichtigsten Funktionen dar, welche während des Evaluierungsprozesses festgestellt werden konnten:

- MDM System
- Management von verschiedenen vielen Geräten
 - Jeder Gerätetyp ist möglich, egal ob Smartphone, Tablet oder Laptop
 - Lokalisierung sämtlicher eingebundener Geräte
 - Leicht aufzusetzen und zu verwenden
 - * Installation einer einzigen App ist notwendig, um das Gerät in die Plattform einzubinden
 - Verwendbar ab dem ersten Gerät
 - Mehrsprachig
 - Geringe Wartungskosten
- Konfiguration der eingebundenen Endgeräte
 - Vorkonfiguration von Email-Konten und sonstigem (Wifi, VPN, etc.)
 - * Der Angestellte muss dies nicht selbst erledigen
 - * Keine Chance einer Fehlkonfiguration
 - * Verteilbar auf hunderte Geräte innerhalb von Sekunden
- Statistiken
 - Verfügbare Statistiken
 - * Gerätetestatus
 - * Kompromitierungsstand
 - * Betriebssystem
 - * Betriebssystemversion
 - * Zugehörigkeit (gehört dem Unternehmen oder dem Angestellten)
 - * Netzbetreiber (3, A1, Telering, etc...)
 - * Registrierungszustand
 - Logging von Events

Device Actions	App -	Policy -	Space -	Status -
Register	Install	Activate	Add Space	Not started
Wipe	Uninstall	Modify	Remove Space	In progress
Lock	Set setting	Deactivate	Change Space Priorization	Completed successfully
Retire	Unset setting		Assign Space Admin	Failed
			Delete Space Admin	

Tabelle 3.2: MDM Event Logs

- Policies
 - Dienen zur Erhöhung der Sicherheit von registrierten Mobilgeräten
 - Blockieren von Systemteilen oder Einstellungen

- * Zum Beispiel: Der Benutzer kann das WLAN oder GPS nicht mehr ausschalten.
- * Diese Funktion wird benötigt, wenn die Chance besteht, dass der Anwender durch gewollte oder ungewollte Aktionen das Gerät in einen unbenutzbaren Zustand bringt.
- Passwortpflicht
 - * Der User ist dazu gezwungen, ein Passwort nach Unternehmensrichtlinien zum Sperren und Entsperren seines Gerätes zu setzen.
- Globaler Proxy
 - * Sämtlicher Netzwerkverkehr wird durch einen Proxy-Server des Unternehmens geleitet, welche dazu dient ungewollten Websiteinhalte zu filtern oder um Unternehmensdaten vor dem Verlassen des Firmennetzwerks zu schützen.
- Kiosk-Modus
 - * Das Gerät wird in einen Zustand versetzt, in dem nur mehr das Benutzen einer einzelnen Applikation möglich ist.
- Applikationen
 - * Erlauben von spezifischen Apps
 - * Verbieten –
 - * Erfordern –

3.2.5.2 Zusatzfeatures

MobileIron bietet die Möglichkeit seine Vielfalt an Features zu erweitern, indem man es an eine sogenannte „Standalone Sentry“ anbindet. Diese ermöglicht es dem Administrator des MDM-Systems die von MobileIron entwickelten Applikationen auf allen Geräten zu installieren. Diese Applikationen dienen dazu den E-Mail-Verkehr, das Webbrowsing, die Installation von Applikationen und den Zugang zu Unternehmensdokumenten zu sichern. Sie heißen:

- Apps@Work
- Docs@Work
- Web@Work

Nachdem sich das Projektteam in diese vertieft hatte, hat es erkannt, dass diese Applikationen eine sogenannte Containertechnologie einsetzen und somit nicht Bestandteil eines standardmäßigen MDM-Systems sind. Daher werden diese in einem anderen Teil des Evaluierungsprozesses behandelt, welcher „MDM+Container“ heißt. Eine weitere Zusatzfunktion, die MobileIron bietet, sind „ActiveSync Policies“. Diese stellen dabei einfach nur durchgehend upgedatete Policies dar, welche bei einer Verletzung sofort Alarm schlagen.

3.3 MDM + Container

3.3.1 MobileIron Containertechnologie

MobileIron Content Management (MCM) ist die Containertechnologie der gleichnamigen Firma MobileIron. Die Containertechnologie basiert auf dem MobileIron MDM System, welches in diesem Projekt separat analysiert wird.

Der Hauptnutzen dieser bestimmten Containertechnologie ist die Verwaltung von Benutzerressourcen. Ursprungsansatz der Entwicklung ist der fortschreitende Trend zur Arbeit auf mobilen Geräten wie Tablets. Das MCM System ist auf folgende Module aufgeteilt: Docs@Work, Apps@Work, Web@Work.

3.3.2 Informationen der getesteten Software

Name	MobileIron Mobile@Work
Hersteller	MobileIron, 415 East Middlefield Road, Mountain View, CA 94043
Version	
Datum	26.01.2015
Preis	/
Website	https://www.mobileiron.com/en/products/product-overview
Dokumentation	https://support.mobileiron.com/eval/

Tabelle 3.3: MDM + Container Übersicht

3.3.3 Installation

Die Konfiguration der Containertechnologie geschieht über ein Webinterface, welches man ebenso für das Standard MDM System von MobileIron verwendet. Dokumentation wie zum Beispiel ein Benutzerhandbuch über die Konfigurationsschritte von MobileIron direkt ist zwar vorhanden, wobei dieses erst größtenteils während unserer Projektzeit erschienen ist. Somit versuchte das Projektteam zunächst dies alleine zu bewältigen und scheiterten, da die gesamte Benutzeroberfläche nur gering selbsterklärend ist. Mittels dem Benutzerhandbuch gelang es uns nach mehreren Versuchen die Containertechnologien Web@Work und Apps@Work funktionstüchtig zu bekommen. Für mehr Aufwand sorgten Zertifikate und Konfigurationsdateien, welche man für die Konfiguration benötigt und nur sehr oberflächig in den bereitgestellten Dokumenten beschrieben worden sind. Weiters beschäftigte uns die nicht konstante und auffallend lange Synchronisationszeit zwischen dem Server und dem Client Device. Der Grund dafür war für uns nicht klar ersichtlich und hätte für eine genaue Analyse zu viel Zeit in Anspruch genommen. Jedoch deutet vieles darauf hin, dass die Ursache in unserem selbstaufgebauten Testnetzwerk entstanden ist, welches sehr klein und minimalistisch gehalten wurde.

3.3.4 End User Products

3.3.4.1 Docs@Work

Docs@Work stellt den Usern unternehmensinternen Content für deren tägliche Arbeit zur Verfügung. Dies basiert auf einem Cloud-Content-Management-System und wird am Gerät als eigene App dargestellt. In erster Linie gewährt die Containertechnologie einen sicheren Verbindungsaufbau zu „Content Repositories“. Unter „Content Repositories“ versteht man im Allgemeinen gewöhnliche unternehmensinterne Fileserver (SharePoint), welche

als Netzwerkressourcen zur Verfügung stehen. Besonders dabei ist, dass auch empfangene Daten via Email in dem gleichen Ausmaß wie normale „Content Repositories“ den Sicherheitsrichtlinien unterworfen sind. Dem User steht die Möglichkeit des Downloads dieser Dokumente offen. Diese Funktion, welche offiziell unter dem Namen „Secure Email Attachment“ geläufig ist, basiert auf MobileIron AppContent. Diese Technologie stellt eine konsistente, sichere Umgebung auf dem Android Gerät zur Verfügung. D.h. Unternehmensdaten welche auf dem Mitarbeiter-Gerät abrufbar sind, sind verschlüsselt.

Aus Administrator-Sicht gewährt Docs@Work neben der zentralen Verwaltung von Ressourcen für einzelne User ebenfalls die Möglichkeit bei nicht gewünschten Tätigkeiten eines Mitarbeiters seine Berechtigung einzuschränken. Administratoren können bestimmte Geräte von Mitarbeitern in Quarantäne verschieben bzw. diese ganz entfernen. Die betroffenen Geräte sind dann nicht mehr in der Lage sich mit dem Server zu verbinden.

Als gewissen Vorteil dieser Technologie kann man den Wegfall des üblichen zusätzlichen VPN auf den Geräten bezeichnen. Die Bedingung ist für den Anwender kompakter und schneller. Sofern der User Zugriff auf den „Content Repository“ hat, gilt die Regelung des Single-Sign-On. Dies gewährt einer nahezu einmaligen Anmeldung per einem User Account und garantiert Zugriff auf alle verknüpften Anwendungen ohne weitere Log-in Session.

Die oben genannte Secure Email Attachment Funktion ist nach Informationen von der offiziellen MobileIron Homepage bis zum heutigen Stand, nur mittels Email Applikationen namens Divide und Email+ funktionstüchtig.

Ebenso gibt es nur ausgewählte „Content Repositories“, welche Docs@Work unterstützen. Zu diesen zählen:

- Microsoft Sharepoint 2007/2010/2013
- CIFS Windows 2008 R2 SPI
- CIFS Samba CentOS 6.2
- Apache-based WebDAV content repositories
- IIS-based WebDAV content repositories

3.3.4.2 Web@Work

Web@Work garantiert Unternehmen einen sicheren, mitunter auch beschränkten Internetzugriff ihrer Mitarbeiter. Es basiert auf zwei Technologien namens AppTunnel und MobileIron Sentry, welche bei Nutzung von Web@Work auch konfiguriert werden müssen. Das Zusammenspiel dieser zwei Technologien gewährt eine Zugangsbeschränkung/Kontrolle sowie den verschlüsselten Datenaustausch. Die Administratoren sind in der Lage gewisse, in den Augen des Unternehmens wichtige Websites für die Mitarbeiter frei zu schalten und somit den Besuch dieser zu erlauben. Unter diesen Websites werden auch interne Unternehmens Websites verstanden. Daten wie der Zwischenspeicher des Browsers, Cookies, die Web History als auch Daten von anderen Websites werden alle verschlüsselt übertragen. Sofern ein Android-Gerät eines Mitarbeiters den Zugangsbestimmungen nicht mehr entspricht, werden all diese Daten aus Sicherheitszwecken gelöscht.

Laut der Dokumentation von MobileIron ist es möglich die User-/Geräte-Verwaltung dieses Systems mit dem Enterprise Directory des Unternehmens zu koppeln. Somit ist das Aktivieren und Zulassen von Websites basierend auf bestimmten Gruppen von Usern möglich.

Die Gegenmaßnahme von MobileIron zur Prävention von DLP (Data Loss Prevention) ist die Deaktivierung vom Erstellen von Screenshots des Users.

Beim Einsatz dieser Technologie steht der Vorteil bezüglich des VPN im Vordergrund. Um Mitarbeitern einen sicheren, abgeschirmten Zugriff auf Webressourcen zu gewähren war bisher eine mögliche Variante, eine VPN Verbindung einzurichten. Web@Work ersetzt das VPN und gewährt weitere Möglichkeiten wie bereits oben beschrieben.

3.3.4.3 Apps@Work

Diese Technologie stellt dem Benutzer des Devices die benötigten Apps zur Verfügung. Dabei kann der User selbst nicht entscheiden, welche Apps er installiert, sondern der IT Administrator. Somit wird das Verwenden des Devices für nicht unternehmensinterne Angelegenheiten verhindert. Der IT Administrator deklariert alle Apps, die laut Unternehmensführung genehmigt sind. Alle anderen Apps welche nicht angeführt sind, sind somit nicht für die Installation zulässig. Diese Technologie basiert auf AppConnect und AppTunnel.

Die erstangeführte Technologie AppConnect ist zuständig für die Implementierung eines Containers um die eigentliche App. Das Resultat daraus ist ein Schutz gegen „data-at-rest“ Daten des Devices bzw. Apps. Die vorhandenen Daten werden verschlüsselt und vor unberechtigten Zugriff geschützt. Auf dem jeweiligen Device sind alle App Container der Apps verbunden und kommunizieren miteinander. Dabei werden Informationen, wie zum Beispiel Richtlinien und Single sign-on Daten, ausgetauscht.

AppTunnel ist für die Sicherheit und den Schutz der data-in-motion Daten zuständig. Diese Technologie stellt sicher, dass die einzelnen Container um die Apps vom restlichen System abgeschirmt sind und keine Verbindung von außen über das Android Basissystem auf die Container stattfinden kann. Die Verbindung wird einzig und alleine zu autorisierten Apps, Usern und Devices aufgebaut. Die „certificate-based session authentication“ verhindert man-in-the-middle Attacken.

Bei der Benutzung von Apps@Work unterscheidet man zwischen 3 verschiedenen Möglichkeiten Apps in das System einzubinden und dem User bereitzustellen.

Die wohl geläufigste Art Apps auf ein Android Device zu implementieren ist das Herunterladen mittels dem Google Play Store.

Der theoretische Ablauf ist folgender: Der IT Administrator tätigt einen Vorschlag für eine ausgewählte App des Google Play Store und hat somit eine Freigabe für den Download dieser App. In unserem Anwendungsfall trifft diese Lösung nicht zu, da der Google Play Store einzig und allein bei Verwendung eines Google Accounts auf dem Device funktioniert. Die Verwendung eines Google Accounts ist in unserem Fall jedoch nicht zielführend, da weitere Sicherheitsprobleme auftreten würden. Daraus ergibt sich die zweite Variante zum Management der auf den Devices installierten Apps anzuwenden. Die In-house Apps basieren hauptsächlich auf selbst entwickelten Applikationen, welche zum Beispiel das Unternehmen selbst in Auftrag gegeben hat. Für die Freischaltung zum Download dieser durch die User benötigt man die APK der App, welcher der IT Administrator zunächst selbst hochladen muss. Weiters kann man mittels Benutzung von In-house Apps ebenfalls zum Teil Apps installieren, welche im Google Play Store verfügbar sind. Voraussetzung dafür ist der rechtmäßige Besitz der APK Files der Apps.

3.4 Samsung Knox

Dieser Abschnitt beschäftigt sich mit dem Einsatz von Samsung Knox als System für potentielle Kunden von der Firma Kapsch. Hier findet sich sowohl Informationen zu Samsung Knox im Allgemeinen, als auch darüber, welche der von der Firma Kapsch erwünschten Features mit Samsung Knox realisierbar sind und welche nicht. Außerdem wird noch auf die Bedienbarkeit und die Installation eingegangen.

3.4.1 Informationen der getesteten Software

Name	Samsung Knox
Hersteller	Samsung
Version	2.1
Datum	08.07.2014
Preis	<ul style="list-style-type: none"> • Express(E): Free but limited to 250 seats • Premium(P): USD \$1 MSRP per device/month • Workspace(W): USD \$3.60 MSRP per device/month
Website	https://www.samsungknox.com/de
Dokumentation	

Tabelle 3.4: Samsung Knox Übersicht

Samsung Knox ist eine Sammlung Business-orientierter Sicherheitsfunktionen für Android-Geräte von Samsung. Das System, welches auf SE for Android basiert, ist speziell auf die Bedürfnisse von Unternehmen in Hinsicht auf die Sicherheit der Endgeräte ausgerichtet.

3.4.2 Samsung Knox Bestandteile

Um wirklich erklären zu können was Samsung Knox ist, muss man es zuerst einmal in seine 3 Hauptbestandteile unterteilen.

3.4.2.1 Platform Security

In jedem Samsung Gerät ist ein physischer Hardware Chip eingebaut, welcher automatisch einen höheren Schutz für Samsung Geräte bieten soll. Mittels diesem Chip ist es Samsung möglich, bis in die tiefste Software-Schicht eines auf Android basierenden Geräts einzugreifen.

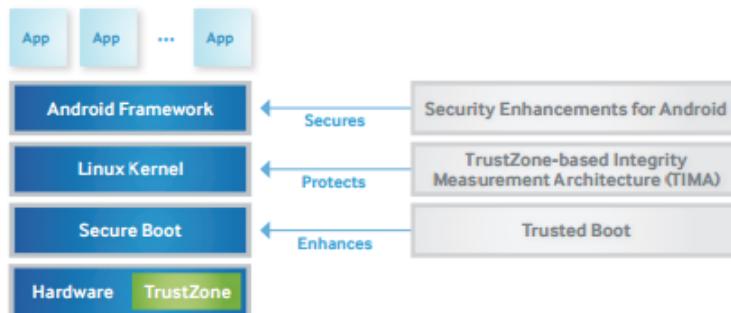


Abbildung 3.1: Samsung Knox Platform Security

In der Grafik links sieht man den groben Aufbau eines Android Systems. In der Grafik rechts sieht man die Technologien, mit deren Hilfe es Samsung möglich ist in die Schichten einzugreifen.

3.4.2.1.1 Kerntechnologien der Platform Security

- **SE for Android**

- **Funktionsweise**

SE for Android basiert auf SELinux-Technologie und definiert die Zugriffskontrolle auf Linux-Ebene. Mandatory Access Control (MAC) und Discretionary Access Control (DAC) überwachen und verwalten welche Dateien und Apps auf das System des Geräts zugreifen können.

- **Sicherheit**

SE for Android erzwingt MAC, wobei Apps nur genau die Rechte zugewiesen erhalten, die sie für den Zugriff auf das System des Geräts benötigen. Sollte ein böswilliger Benutzer oder eine bösartige App also Zugriff auf das Gerät erhalten, dann betrifft der Schaden immer nur einen bestimmten Bereich, während die restlichen Bereiche des Geräts geschützt bleiben.

Wenn SE for Android ausgelöst wird, sendet es eine Präventionsinformationsmeldung an den Benutzer des Geräts, die in der Informationsleiste erscheint. Die Präventionsinformationsmeldung gibt an, welche Anwendung versucht hat, auf Daten des Geräts zuzugreifen, und sie wird die Deinstallation der betreffenden Anwendung empfehlen. Beachten Sie dabei jedoch, dass SE for Android auch von autorisierten Anwendungen ausgelöst werden kann. Dies kann vorkommen, wenn die App einen von dem in der SE for Android-Richtlinie vorgegebenen Pfad verwendet. Wenn dies der Fall ist, sollten Sie die Datei der Sicherheitsrichtlinie entsprechend aktualisieren.

- **ARM TrustZone-Based Integrity Measurement Architecture (TIMA)**

- **Funktionsweise**

TIMA schützt Ihr Gerät auf zwei verschiedene Weisen. Erstens prüft sie in regelmäßigen Abständen, ob der Kernel des Geräts geändert wurde, indem sie den gegenwärtigen Zustand mit dem Original-Kernel vergleicht. Zweitens authentifiziert TIMA Kernel-Module, wenn diese geladen werden, sodass Geräte nie ungeschützt sind.

- **Sicherheit**

Die TIMA TrustZone ist ein manipulationssicherer Sektor des ARM-Prozessors in Ihrem Gerät. Sie authentifiziert und verifiziert den Linux-Kernel über regelmäßige Messungen. Wenn TIMA for Android ausgelöst wird, sendet es eine Ermittlungsinformationsmeldung an den Benutzer, die in der Informationsleiste erscheint. In der Meldung werden Sie normalerweise zum Neustart des Geräts aufgefordert.

- **Secure Boot und Trusted Boot**

- **Funktionsweise**

Secure Boot verhindert, dass unbefugte Bootloader und Kernels in das Gerät geladen werden. Dies bedeutet, dass Ihr Gerät nicht manipuliert wurde und der KNOX-Container geladen werden kann.

Trusted Boot vergleicht den Bootloader und den Kernel des Betriebssystems mit den originalen Werksversionen. Dies wird dadurch erreicht, dass die Originaldaten des Geräts aufgezeichnet werden und das Gerät permanent beim Systemstart mit diesen gegengeprüft wird, um sicherzustellen, dass sich diese Daten nicht geändert haben.

- Sicherheit

Es gibt vier Bootloader auf dem Gerät. Jeder Bootloader prüft die Gültigkeit des vorhergehenden Bootloaders oder Kernels. Trusted Boot sichert diese Bootloader. Die Funktion ist in der ARM TrustZone, einem manipulationssicheren Sektor des ARM-Prozessors, eingebettet. Trusted Boot verwendet kryptografische Schlüssel, um sicherzustellen, dass die Messungen am Gerät dem Original entsprechen. Diese Schlüssel werden erst von Trusted Boot freigegeben, wenn SE for Android bestätigt, dass genehmigte Firmware auf dem Gerät ausgeführt wird.

3.4.2.2 Application Security

3.4.2.2.1 Application Containers Container sind quasi ein Android im Android und bezeichnet einen gesicherten, separaten Bereich (Container) auf dem Gerät. Dieser Bereich hat einen eigenen Homescreen, eigene Anwendungen und eigene Daten. Diese Funktion ist damit vergleichbar mit einer Art Dual-Boot-Variante. Anwendungen außerhalb des Containers haben dabei keinerlei Zugriff auf die Daten oder Prozesse innerhalb des Containers. Anwendungen innerhalb des Containers haben grundsätzlich keinen Zugriff auf Daten außerhalb des Containers.

Mittels Richtlinienkonfiguration kann der IT-Verwalter des Gerätes einen read-only (nur lesen) Zugriff für bestimmte Anwendungen im Container auf Daten außerhalb des Containers einrichten. Umgekehrt besteht diese Möglichkeit allerdings nicht. Daten innerhalb des Containers werden dabei mit einem Verschlüsselungsalgorithmus und einem AES-256 Bit Schlüssel geschützt. Ein Zugriff ist erst nach Eingabe eines Passwortes möglich.

3.4.2.2.2 On-Device Data Encryption (ODE) ODE ist ein im Knox enthaltenes Feature zur Verschlüsselung von Daten. Dabei kann sowohl der sichere Container als auch der normale Bereich, sowie interner als auch externer Speicher verschlüsselt werden. Verwendet wird ein AES Algorithmus mit einem 256 Bit starken Schlüssel. Die Verschlüsselungsfunktion kann vom User selbst unter den Einstellungen oder vom IT Administrator durch eine Richtlinie aktiviert werden.

3.4.2.2.3 Virtual Private Network Support (VPN) VPN-Verbindungen verwendet man um sicherzustellen, dass Daten bei der Übertragung geschützt sind und um den Netzwerkverkehr nicht mit den Daten von persönlichen Apps zu belasten.

- **Funktionsweise**

KNOX Workspace bietet 3 VPN-Optionen für den Schutz Ihrer Daten bei der Übertragung. Ein geräteweites VPN kann von Benutzern konfiguriert werden, sofern sie den entsprechenden Servernamen und die dazugehörigen Informationen zur Hand haben. VPN pro App oder ein containerweites VPN kann von IT-Administratoren über die MDM-Konsole eingerichtet werden. Auf der MDM-Konsole kann man bis zu 5 verschiedene VPN-Profile einrichten und diese einzelnen Apps zuweisen, um VPN pro App zu implementieren.

- **Sicherheit**

VPN-Verbindungen sind die sicherste Methode, um die Daten bei der Übertragung zu schützen. Der KNOX VPN-Client kann ein bestehendes VPN-Gateway für den Schutz von Daten verwenden. KNOX VPN ist über Ihre MDM-Konsole im FIPS-Modus konfigurierbar. Zu den Sicherheitsfunktionen gehören NSA-Suite-B-Algorithmen, Unterstützung für X.509 mit Zertifikatsprüfung auf OCSP-Basis und 256-Bit-AES-Verschlüsselung. Wenn Ihr Unternehmen SmartCards verwendet, können diese mit VPN-Anmeldedaten konfiguriert werden.

3.4.2.3 Management

Mit MDM (Mobile Device Management) bietet Samsung Knox der IT Abteilung des Unternehmens die Möglichkeit eine integrierte Lösung zur Verwaltung und Administration von Samsung-Geräten vorzunehmen, ohne dabei auf Drittanbieter zurückgreifen zu müssen. Der Nachteil dabei ist die ausschließliche Verwendung mit Samsung-Geräten, welche über Samsung Knox verfügen.

Samsung Knox gibt es in 2 Varianten, wobei beide Varianten ein MDM als Basis benötigen um zu funktionieren.

3.4.2.3.1 Knox Lösungen

- **Knox Express**

Diese Lösung ist für kleine bis mittelständische Unternehmen gedacht. Sie ist gratis, jedoch limitiert auf 250 Geräte.

Falls die 250 Plätze nicht mehr reichen, kann man Knox Express problemlos auf Knox Premium updaten.

Das ist die Variante, die die Projektgruppe empfehlen würde, da sie gratis ist und da laut Angaben von der Firma Kapsch es eher unwahrscheinlich ist, dass ein Kunde mehr als 250 Geräte hat und falls doch ist es ja problemlos auf Premium erweiterbar.

- **Knox Premium**

Diese Komplettlösung eignet sich ideal für Unternehmen, in denen Sicherheit oberste Priorität hat.

Knox Premium kostet pro Gerät pro Monat 1\$ und kann um zusätzliche Add-ons erweitert werden.

Weiters bietet Knox Premium einige zusätzliche Features für Android und eine bessere IOS Integration. In Hinblick auf die von Kapsch erwünschten Funktionalitäten bringen diese zusätzlichen Features jedoch keine Vorteile.

- **Add-Ons**

- **Knox Workspace**

Knox Workspace ist eine Erweiterung im Bereich Container.

Dieses Add-on bietet eine einfachere Konfiguration, zwei Container pro Gerät, zusätzliche Apps und verbesserte Sicherheitsfunktionen.

Die Hauptfeatures, die Workspace mit sich bringt, sind:

- * Erweiterte Containerverwaltung mit sicheren Richtlinien
- * Datenverschlüsselung bei jedem Entsperren des Containers
- * Pro-App-VPN für sichere und schnelle Verbindung
- * Support für zwei separate Container für maximale Produktivität und für Trennung von arbeitsbezogenen und privaten Daten

All diese Einstellungen, die mit Knox Workspace dazu kommen, gelten nur innerhalb des Containers. Da es jedoch keine Einstellung gibt um zu verhindern, dass der Benutzer den Container verlässt und somit alle Workspace-Einstellungen umgeht, ist dieses Add-on in unserem Fall ungeeignet.

Jedoch soll in Zukunft eine Art Container-only-mode implementiert werden. Sobald dieser vorhanden ist, wäre Knox Worspace auf jeden Fall ein Addon das in Betracht gezogen werden sollte.

- **Knox IAM**

Knox IAM ist eine SSO-Lösung(Single Sign-On-Lösung).

Das heißt mit dem IAM Add-on hat man nur mehr einen einzigen Account, mit dem man sich in allen Apps anmelden kann.

3.4.2.3.2 MDMs Knox bietet zur Verwaltung der Geräte ein eigenes MDM, das EMM, hat aber auch genügend Partner, deren MDMs Samsung Knox Funktionen implementiert haben.

Samsung Knox funktioniert also mit jedem MDM aus der nachfolgenden Liste:

- **Liste an Kompatiblen MDMs**

- Samsung EMM
- MobileIron
- Absolute Software
- AirWatch
- CA Technologies
- Centrify
- Citrix
- FancyIron
- MaaS360
- NQ Mobile
- Samsung SDS
- SAP
- SOTI

- **Samsung EMM**

Samsung KNOX EMM ist eine cloudbasierte Verwaltungslösung für Unternehmen. IT-Administratoren können damit Benutzer, Apps und plattformübergreifende Geräte über eine cloudbasierte Konsole verwalten. Außerdem bietet KNOX EMM Single Sign-On (SSO) und eine starke Authentifizierung für eine benutzerfreundliche und sichere Arbeitsumgebung für Mitarbeiter.

Es ist einfach zu installieren und sehr übersichtlich gestaltet. Falls in der Firma noch keine andere MDM Lösung installiert sein sollte, ist EMM zu empfehlen.

Auswahl & Konzept

4.1 Endergebnis und Empfehlung

Nach Abschluss des Evaluierungsprozesses ist das Projektteam an diesem Punkt in der Lage eine Empfehlung an das Unternehmen Kapsch auszusprechen. In diesem Teil wird nun erläutert, welches evaluierte Konzept mit den Anforderungen des Auftraggebers übereinstimmt und wodurch dies zu Stande gekommen ist. Nachdem die Linuxmanipulation aus Garantietechnischen Gründen bereits als ausgeschieden gilt, befasst sich der folgende Abschnitt nur mehr mit den Konzepten MDM, MDM + Container und Samsung Knox.

4.2 Evaluierung

Um feststellen zu können welche der 3 Lösungen am besten die Anforderungen des Auftraggebers erfüllen kann, hat das Projektteam eine Nutzwertanalyse erstellt, mit derer Hilfe das Projektteam die Funktionen der Systeme direkt miteinander vergleichen konnten.

Nutzwertanalyse ERP Lösung		max. Punkte	Gewicht	MDM	Bewertung	MDM+Container	Bewertung	Samsung Knox	Bewertung
Allgemein		8							
Nutzbar ohne Public-Cloud-Services		3	10	30	10	30	10	30	
Keiner speziellen Person zugewiesen		5	10	50	10	50	10	50	
Nutzwert Allgemein		80		80		80		80	
Inhaltseinstellung		44							
Installieren von Apps kann verboten werden		6	7	42	7	42	10	60	
Deinstallieren von Apps kann verboten werden		6	3	18	3	18	0	0	
Kein benutzerseitiger Zugang zu Systemeinstellungen		8	10	80	10	80	10	80	
Standard Apps können ausgeblendet werden		2	2	4	2	4	2	4	
Browser Inhalt kann gefiltert werden		4	10	40	10	40	10	40	
Remote-Zugriff ist möglich		8	10	80	10	80	10	80	
Apps können per Remote verwaltet werden		5	6	30	8	40	10	50	
Websites können per Remote verwaltet werden		2	0	0	0	0	0	0	
Inhalt (Dokumente, Videos, Bilder) können per Remote verwalten können		1	0	0	8	8	0	0	
Hintergrund und Speerbildschirm können per Remote verwalten können		2	5	10	5	10	5	10	
Nutzwert Inhaltseinstellungen		440		304		322		324	
Statistik		7							
Up/Down-Time einsehbar		3	10	30	10	30	10	30	
Statistiken sind pro Device und Standort auslesbar		3	10	30	10	30	10	30	
Statistiken über Aktivitätszeit, App aufrufe und Websites		1	4	4	4	4	10	10	
Nutzwert Statistik		70		64		64		70	
Benutzer und Geräte		10							
Geräte können zu einem spezifischen Status zurückgesetzt werden		5	3	15	3	15	3	15	
Benutzerdaten (history, Spielstände, Lesezeichen) können gelöscht werden		3	3	9	4	12	3	9	
Automatischen zurücksetzen der Geräte nach einer Zeiteinheit		2	0	0	0	0	0	0	
Nutzwert Benutzer und Geräte		100		24		27		24	
Managementseitige Anforderungen		20							
Geräteverwaltung von mehreren Standorten möglich		3	10	30	10	30	10	30	
Geräte (inkl. deren Status) können angezeigt werden		4	10	40	10	40	10	40	
Fehlerberichte sind pro Gerät und pro Standort verfügbar		4	7	28	7	28	9	36	
Geräte können zu einem spezifischen Status per Remote zurückgesetzt werden		5	3	15	3	15	3	15	
Statistiken können per Remote ausgelesen werden		3	10	30	10	30	10	30	
Inhalt (Dokumente, Videos, Bilder) können per Remote geändert können		1	0	0	8	8	0	0	
Nutzwert Managementseitige Anforderungen		200		143		151		151	
Zusätzliche Anforderungen		11							
Gerät ohne Kiosk-Mode verwendbar		5	10	50	10	50	10	50	
Hardwareunabhängig		1	10	10	10	10	10	10	
Geringer Wartungsaufwand		2	10	20	10	20	10	20	
mehrsprachigen Unterstützung		2	10	20	10	20	10	20	
Einsetzbar ab dem ersten Gerät		1	0	0	0	0	10	10	
Nutzwert Zusätzliche Anforderungen		110		100		100		110	
Nutzwert Gesamt		1000		715		744		759	
Rang Nutzwert				3		2		1	
Kosten		10		6		4		10	
Rang Kosten				2		3		1	
Kosten/Leistungsverhältnis - Kosten je Pkt.				119,17		186,00		75,90	
Rang Kosten/Leistungsverhältnis				2		3		1	

4.3 Auswertung der Evaluierung

Um eine aussagekräftiges Ergebnis zu erhalten, wurden die Anforderungen des Gesamtsystems in verschiedene Unterpunkte unterteilt, welche auf ihre Funktionstüchtigkeit hin untersucht wurden.

Die Lösungen wurden dann je nach Erfüllungsgrad der einzelnen Punkte bewertet, was eine objektive Vergleichbarkeit schaffen sollte.

4.3.1 Allgemein

Da alle 3 Systeme sowohl ohne public-cloud-service nutzbar waren als auch keiner speziellen Person zugewiesen werden müssen, haben die alle 3 Lösungen volle Punkte erreicht und sind somit was diesen Punkt angeht gleich auf.

4.3.2 Inhaltseinstellungen

In diesem Bereich hat sich herausgestellt ist Samsung Knox kapp aber doch noch vor der MDM+Container Lösung das am besten geeignete System. Denn auch wenn es mit MDM+Containern möglich ist, die Deinstallation von betriebsinternen Apps zu verbieten und man auch Inhalte der Geräte wie zum Beispiel Bilder, Dokumente, etc per remote verwalten kann, ist Samsung die hier die bessere Alternative, da es eine bei weitem bessere Verwaltung von Apps per Remote bietet.

Eine Anforderung die jedoch keines der drei Systeme erfüllen konnte war das Verwalten von Websites per remote.

4.3.3 Statistik

Auch den Unterpunkt Statistiken kann Samsung Knox wieder für sich entscheiden. Diesmal jedoch mit einem Größeren Vorsprung. Zurückzuführen ist das darauf, dass Samsung Knox die Möglichkeit bietet, sich jede beliebige Statistik die man für sein Unternehmen haben will, einfach mittels SQL Code selbst zu erzeugen(siehe Punkt 8.3.1 Statistiken).

4.3.4 Benutzer und Geräte

In diesem Bereich kann die MDM+Container Lösung punkten. Denn im Gegensatz zu den anderen zwei Systemen, kann man mit dieser Lösung am Gerät gespeicherte Lesezeichen löschen.

Auch bei diesem Unterpunkt gibt es wieder eine Anforderung die mit keinem unserer getesteten Systeme umsetzbar war. Und zwar war es nicht möglich die Geräte so zu konfigurieren, dass sie sich nach einer gewissen Zeit von alleine auf einen definierten Stand zurücksetzen.

4.3.5 Managementseitige Anforderungen

Bei der Erfüllung der Managementseitigen Anforderungen sind Samsung Knox und die MDM+Container-Lösung gleich auf. Denn den Vorsprung den Knox gewinnt durch besseres Handling der Fehlerberichte, kann die MDM+Container-Lösung dadurch wegmachen, das sie im Gegensatz zu Samsung Knox eine Möglichkeit bietet, Geräteinhalt per remote zu ändern.

4.3.6 Zusätzliche Anforderungen

Da Samsung Knox das einzige System ist, welches bereits ab dem ersten Gerät einsetzbar ist und sonst alle Punkte von allen Systemen erfüllt werden ist auch hier Knox der klare Sieger.

4.4 Empfehlung

Nach eingehender Analyse ist das Projektteam zu dem Schluss gekommen, dass für die geplanten Projekte der Firma die Betriebsplattform Samsung Knox am ehesten geeignet ist. Es implementiert die meisten der benötigten Features, aber lässt dennoch einige fundamentale Punkte aus. Deshalb ist es hier für das Projektteam auch nicht möglich eine hundert prozentige Empfehlung zu geben. Den Informationen der Dokumentation von Samsung Knox zur Folge werden einige benötigte Features in kommenden Versionen eingebaut. Allerdings ist nicht absehbar ob und wann diese erscheinen. Für die beiden anderen Systeme kann deshalb keine Empfehlung ausgesprochen werden, weil sie weniger und besonders im Einsatz mit Nicht-Samsung-Geräten signifikant weniger Funktionen bieten.

Lessons Learned

5.1 Sebastian Götze

Durch das Diplomprojekt Kapsch Tablet Infrastructure habe ich gelernt, dass es besonders bei der Recherche wichtig ist, sich nicht immer auf den Inhalt von Dokumenten und Präsentationen zu verlassen, weil dieser oft nicht die Realität wiederspiegeln. Besonders beim System Samsung Knox hat sich dies deutlich herauskristallisiert, da in Samsungs Unterlagen zwar steht, dass es einen Modus gibt, in dem der Nutzer sich ausschließlich in einem abgesicherten Container befindet, aber dieser in der Praxis nur bedingt existiert. Der Modus war vorhanden, allerdings war es für den Nutzer jederzeit möglich aus diesem herauszuspringen. Ein weiterer wichtiger Lernaspekt für mich war es, dass man bei der Installation von unbekannten Systemen vor Beginn die Installationsanleitung lesen sollte und nicht sich danach auf Fehlersuche zu begeben. Besonders bei der Einrichtung der zusätzlichen Containerapplikationen zu MobileIron wäre dies von Vorteil gewesen, da es hier einige wichtige Details zu beachten gab, ohne die der Prozess nicht abgeschlossen werden konnte.

5.2 Samuel Hammer

Während das Projekts KTI habe ich in erster Linie vieles bezüglich Projektmanagement gelernt. Durch die oftmals eng gesetzten Zeitintervalle, war es äußerst wichtig die Koordination zwischen den zwei Subteams bzw. den einzelnen Teammitgliedern zu perfektionieren. Ein teilweise großes Problem war auch, dass sich Projektbezogene Abgaben und Termine immer wieder mit Tests, Schularbeiten und anderen unterrichtsbezogenen verpflichtungen überschnitten. Hier war es von großer Wichtigkeit, sich mit den zuständigen Lehrern bzw. dem Projektpartner abzusprechen. Einiges gelernt habe ich natürlich auch in Hinblick auf die wichtigen Dokumente, welche es im Laufe des Projektes zu erstellen und aktualisieren gab. Alles in Allem war das Projekt "Kapsch Tablet Infrastructure" eine wertvolle Erfahrung, aus der ich viel für meine zukünftige berufliche Laufbahn mitnehmen werden.

5.3 Michael Kaufmann

Einer der wichtigsten Punkte die ich durch die Arbeit am Diplomprojekt in Erfahrung bringen konnte ist, wie wichtig gut funktionierende Kommunikation innerhalb des Projekt Teams ist. Um den Arbeitsaufwand bestmöglich zu verteilen und um, so produktiv wie möglich, sein zu können haben wir Sub-Teams gebildet und unser Projekt in technischen und dokumentationsbezogenen Teil geteilt. Das Problem hierbei war nur das die Sub-Teams teilweise abhängig von einander waren und es hin und wieder vorkam das die Arbeit, auf die das eine Team warten musste um weiter machen zu können, bereits erledigt war, aber diese nichts davon wussten. Dies führte logischerweise zu Verzögerung des gesamten Projekts. Außerdem habe ich gelernt dass man der Dokumentation und den Whitepapers einer Fertiglösung nicht immer blind vertrauen kann. Denn wie zum Beispiel im Falle Samsung knox sich herausgestellt hat, werden oft Funktionen versprochen, welche tatsächlich nicht existieren. Dies führt nicht nur zu einer Verzögerung des Aufwandes was das testen der Lösung betrifft, sondern auch dazu das die Lösung vor dem Testen besser eingeschätzt wurde als sie tatsächlich war.

5.4 Konstanze Müller

Das diesjährige Diplomprojekt mit unserem Partner der Kapsch Business Com gewährte mir in erster Linie einen Einblick eines kompletten Projektdurchlaufes in der Praxis. Theoretische Inhalte für einen reibungslosen Projektverlauf, welche wir in den Jahren davor im Unterricht gelehrt bekommen haben, konnten wir als Projektteam in diesem Abschlussjahr erfolgreich umsetzen. Auch wenn unser Projektleiter Philip Steinhäuser in diesem Bereich die Oberhand hatte, unterstützte ich ihn so gut wie möglich und war somit auch mit formalen Abwicklungskriterien vertraut. Aus Projekttechnischer Sicht lernte ich, dass man den Aufwand für die Evaluierung von einem System schwer im Voraus abschätzen kann. Des Öfteren war es der Fall, dass Anforderungen von der Checkliste nicht auf Anhieb funktionierten. Selbstverständlich hört man nicht gleich auf zu testen, sondern probiert weiteres. Das Schwierige meiner Meinung war nun den richtigen Zeitpunkt zu finden, um mit einer hohen Wahrscheinlichkeit sagen zu können, dass diese Anforderung vom ausgewählten System nicht unterstützt wird. In Summe bedeutet dies, dass man in Projekten wie diesem sich selbst eine Grenze stecken muss um auch ein schlussendliches Ergebnis vorlegen zu können.

5.5 Philip Steinhäuser

Für mich als Projektleiter hat dieses Projekt ein Menge an Erfahrungen nicht nur im Managementbereich sondern auch im Zwischenmenschlichen Umgang bereitgehalten, da ich in einigen Situationen dazu gezwungen war ein Machtwort zu sprechen wenn es Streitigkeiten innerhalb des Projektteams gegeben hat. Zusätzlich hab ich einiges zum Thema Zeitplanung und Terminkoordination gelernt, was mir auf meinem weiteren Weg welcher mich hoffentlich in die Wirtschaft bzw. einen wirtschaftlichen Beruf führen wird helfen wird. Zu guter Letzt bleibt mir nicht mehr zu sagen als das ich dieses Projekt als ein für mich sehr Erfahrungsreiches Ereignis verbuche, welches mir in den Bereichen der Teamführung bzw. Projektleitung oder Projektcontrolling sowie Projektmanagement als auch Krisenmanagement eine bzw. mehrere Lehrstunden erteilt hat. Dennoch möchte nichts was wir an dem Projekt gemacht haben sowie die Zusammenarbeit mit diesem großartigen Projektteam missen.

Danksagung

Wir möchten uns in erster Linie bei unseren Betreuungslehrern, Herrn DI Harald Swoboda und Herrn DI Hannes Färberböck für die großartige Unterstützung während des gesamten Projektverlaufs bedanken.

Großer Dank gilt auch Herrn DI (FH) Mag. Bernhard Bruckner und Herrn Jürgen Krammer, die uns im Namen der Kapsch BusinessCom bei Problemen immer mit Rat und Tat zur Seite standen.

Herzlichen Dank auch an unsere Deutsch Professorin, Frau Dr. Margaretha Antonescu, die sich freundlicherweise dazu bereit erklärte, diese Diplomarbeit zu korrigieren.

Begleitprotokoll

Name des Schülers: Sebastian Götze

Name des Schülers: Samuel Hammer

Name des Schülers: Michael Kaufmann

Name der Schülerin: Konstanze Müller

Name des Schülers: Philip Steinhäuser

Thema der Arbeit: KTI - Kapsch Tablet Infrastructure

Name der Betreuungsperson: DI Harald Swoboda

Datum	Bearbeiter/in	Tätigkeit	Anmerkung
26.01.2015	Philip Steinhäuser	<ul style="list-style-type: none"> • Kurzfassung in Deutsch • Kurzfassung in Englisch 	
26.01.2015	Samuel Hammer	Beginn der Android-Studie	Erste Teile der Diplomarbeit in LaTeX integriert.
27.01.2015	Philip Steinhäuser	Einleitung erstellt	
27.01.2015	Samuel Hammer	einleitung in LaTeX Dokument integriert	
29.01.2015	Philip Steinhäuser	Ergebnis aktualisiert	
29.01.2015	Samuel Hammer	Weitere Teile in LaTeX eingebunden	
19.02.2015	Samuel Hammer	Aktualisierung der Studie	
04.03.2015	Philip Steinhäuser	Aktualisierung der Struktur der Diplomarbeit (Inhaltsverzeichnis)	
27.03.2015	Samuel Hammer	Aktualisierung der Struktur der Diplomarbeit (Inhaltsverzeichnis)	
02.04.2015	Samuel Hammer	<ul style="list-style-type: none"> • Anlegen der Punkte Varianten und Auswahl & Konzept • Aktualisierung des Quellenverzeichnisses 	
16.04.2015	Philip Steinhäuser	Übertragung des Inhaltes des Untersuchungsberichts in den Punkt Varianten.	
16.04.2015	Samuel Hammer	Übertragung des Inhaltes des Untersuchungsberichts in den Punkt Auswahl & Konzept	
30.04.2015	Samuel Hammer	Aktualisierung der Struktur der Diplomarbeit (Inhaltsverzeichnis)	
30.04.2015	Philip Steinhäuser	Vervollständigung der Diplomarbeit <ul style="list-style-type: none"> • Text zu Projektpartner verfasst • Arbeitsaufteilung dokumentiert • Lessons Learned verfasst • Danksagung hinzugefügt • Quellen-, Tabellen- und Abbildungsverzeichnis aktualisiert 	
30.04.2015	Samuel Hammer	Vervollständigung der Diplomarbeit <ul style="list-style-type: none"> • Text zu Projektpartner verfasst • Arbeitsaufteilung dokumentiert • Lessons Learned verfasst • Danksagung hinzugefügt • Quellen-, Tabellen- und Abbildungsverzeichnis aktualisiert 	

Tabelle 7.1: Begleitprotokoll

Quellenverzeichnis

Abbildungsverzeichnis

1.1	Kapsch Logo	2
1.2	Projekt-Organigramm	4
1.3	Aufgabenverteilung - Diagramm	5
2.1	Android Systemarchitektur	10
3.1	Samsung Knox Platform Security	23

Tabellenverzeichnis

1.1	Projektablauf	6
3.1	MDM Übersicht	17
3.2	MDM Event Logs	18
3.3	MDM + Container Übersicht	20
3.4	Samsung Knox Übersicht	23
7.1	Begleitprotokoll	36

Anhang

1. Kooperationsvereinbarung
2. Pflichtenheft
3. Projekthandbuch
4. Untersuchungsbericht
5. Abnahmeprotokoll

KOOPERATIONSVEREINBARUNG

zwischen

1. Kapsch BusinessCom AG, Wienerbergstraße 53, 1120 Wien

vertreten durch

 [Name des/der Vertreters/Vertreterin]

(in der Folge „**die Projektpartnerin**“)

und

2. Götze Sebastian, Hammer Samuel, Kaufmann Michael, Müller Konstanze, Steinhäuser Philip

..... [Namen der SchülerInnen]

(in der Folge „**das Projektteam**“)

PRÄAMBEL

Das Projektteam und die Projektpartnerin beabsichtigen gemäß der Verordnung über die abschließenden Prüfungen in den berufsbildenden mittleren und höheren Schulen, BGBL II, Nr. 70/2000 vom 24.2.2000, die Planung und Durchführung eines Diplomprojektes.

Durch die Zusammenarbeit soll insbesondere den Mitgliedern des Projektteams die Möglichkeit eingeräumt werden, im Rahmen ihrer schulischen Ausbildung bei der Durchführung eines Projektes an die Verhältnisse im technischen Berufsleben herangeführt zu werden, um dabei die in der Schule erworbenen theoretischen Kenntnisse und Fähigkeiten in der Praxis anzuwenden bzw. zu erweitern. Hingewiesen wird in diesem Zusammenhang auf den unentgeltlichen Charakter dieser Vereinbarung.

Die am Projekt teilnehmenden SchülerInnen werden in der Folge als Projektteam bezeichnet. Die Inhalte dieser Vereinbarung gelten für jedes einzelne Projektmitglied gleichermaßen.

§ 1 Gegenstand

Gegenstand ist die Erstellung von Arbeitsergebnissen zum Thema des Diplomprojektes. Das Thema des Diplomprojektes ist der

Projektbeschreibung und dem Pflichtenheft zu entnehmen, welches der Kooperationsvereinbarung beiliegt.

Das Diplomprojekt hat die Erstellung folgender Arbeitsergebnisse zum Ziel: **[genau beschreiben]**

Die Projektpartnerin wird darauf hingewiesen, dass es sich um ein Projekt im Zusammenhang mit der schulischen Ausbildung handelt und daher jede Haftung des Projektteams bzw. der Projektpartnerin hinsichtlich der Unentgeltlichkeit des Vertrages ausgeschlossen ist.

§ 2 Laufzeit

Die vorliegende Kooperation tritt am in Kraft und wird bis zum Ende der Reife- und Diplomprüfung an der HTL Spengergasse abgeschlossen.

§ 3 Rechte und Pflichten des Projektteams

Die Mitglieder des Projektteams haben das Recht, die Räumlichkeiten der Projektpartnerin samt Infrastruktur und EDV-Infrastruktur im für die Projektabwicklung erforderlichen Ausmaß nach vorheriger schriftlicher Genehmigung durch die Projektpartnerin mitzubenutzen.

Das Projektteam verpflichtet sich, die im Gegenstand genannten Arbeiten sorgfältig und unter möglichster Schonung der Interessen der Projektpartnerin durchzuführen.

Das Projektteam unterliegt der Betriebsordnung der Projektpartnerin.

Das Projektteam verpflichtet sich zur Geheimhaltung aller ihm zur Kenntnis gelangenden Geschäfts- und Betriebsgeheimnisse (siehe auch § 6 unten).

§ 4 Rechte und Pflichten der Projektpartnerin

Die Projektpartnerin verpflichtet sich, dem Projektteam beratend zur Verfügung zu stehen und alles zu unterlassen, was der Vollendung des Projekts entgegensteht.

Die Projektpartnerin verpflichtet sich, dem Projektteam folgende Hilfsmittel zur Verfügung zu stellen:

.....
.....

§ 5 Urheberrechte, Erfindungen

Das Projektteam verpflichtet sich gegenüber der Projektpartnerin hinsichtlich des gegenständlichen Diplomprojekts alle übertragbaren Teile des Urheberrechtes und sämtliche damit verbundenen Verwertungsrechte ausschließlich an die Projektpartnerin zu übertragen.

Die Projektpartnerin wird demnach berechtigt, das vom Projektteam erstellte Diplomprojekt und die darin enthaltenen Erkenntnisse territorial und zeitlich unbegrenzt in jeder beliebigen Art zu nutzen. Das Projektteam hat der Projektpartnerin auch einen allfälligen Source-Code in zu vereinbender Form zu übertragen.

Die Übertragung der Nutzungs- und Verwertungsrechte durch das Projektteam an die Projektpartnerin erfolgt ausschließlich, das heißt auch das Projektteam selbst verpflichtet sich, die Ergebnisse ihrer Arbeit, insbesondere auch Software und Source-Code, weder ganz noch teilweise für sich selbst zu nutzen oder an Dritte, abgesehen von § 7 unten, weiterzugeben.

Sollte das Projektteam im Rahmen dieser Kooperationsvereinbarung eine Erfindung machen, die nach dem Gebrauchsmustergesetz bzw. dem Patentgesetz schutzwürdig ist, gilt diese Erfindung als Diensterfindung im Sinne des PatG und die §§ 6-19 PatG (in der geltenden Fassung) entsprechend. Das Projektteam verpflichtet sich, die Projektpartnerin von einer im Rahmen der Kooperationsvereinbarung gemachten Erfindung unverzüglich in Kenntnis zu setzen. Die Projektpartnerin hat daraufhin das Recht, binnen vier Wochen ab dieser Bekanntgabe zu erklären, dass sie das Patentrecht für sich beansprucht. In diesem Fall steht dem Projektteam eine entsprechende Vergütung nach den einschlägigen Bestimmungen des PatG (in der geltenden Fassung) zu.

§ 6 Geheimhaltung

Das Projektteam ist zur Verschwiegenheit gegenüber Dritten verpflichtet, in Bezug auf sämtliche Informationen und Unterlagen, die es im Rahmen der Erstellung des Projektes durch die Projektpartnerin erhält bzw. die ihm zur Kenntnis gelangen und vor allem zur Wahrung sämtlicher Betriebs- und Geschäftsgeheimnisse der Projektpartnerin und der einschlägigen Datenschutzvorschriften. Diese Verpflichtung besteht auch nach Beendigung dieser Vereinbarung fort.

Sollte die Weitergabe der oben beschriebenen Daten, Informationen oder Unterlagen an Dritte, notwendig sein, so darf diese Weitergabe,

abgesehen von § 7 unten, erst erfolgen, nachdem die Projektpartnerin dieser ausdrücklich schriftlich zugestimmt und Einsicht in allfällige relevante schriftliche Unterlagen erhalten hat. Das Projektteam haftet auch für die Einhaltung der Geheimhaltungsverpflichtung durch diese Personen.

Das Projektteam verpflichtet sich bei Beendigung dieses Vertrages, aus welchen Gründen auch immer, alle ihm zur Verfügung gestellten und noch in seinem Besitz befindlichen Unterlagen an die Projektpartnerin auszufolgen.

Bei Verletzung der Geheimhaltungsverpflichtung ist die Projektpartnerin unter anderem berechtigt, Schadenersatz zu verlangen.

§ 7 Einsicht und Präsentation

Da die Tätigkeit des Projektteams auch Inhalt bzw. Grundlage der an der HTL Spengergasse zu erstellenden Diplomarbeit ist, berechtigt die Projektpartnerin die zuständigen Organe des Bundes zur Einsicht und Kontrolle, um die in der Verordnung über die abschließenden Prüfungen an den berufsbildenden mittleren und höheren Schulen genannten Aufgaben zu erfüllen. Das Projektteam ist auch berechtigt, Ergebnisse der Diplomarbeit bei der mündlichen Reifeprüfung zu präsentieren. Die zuständigen Organe des Bundes sind ihrerseits wiederum gegenüber jedermann zur Geheimhaltung über sämtliche ihnen dabei zur Kenntnis gelangenden Geschäfts- und Betriebsgeheimnisse der Projektpartnerin verpflichtet.

§ 8 Vorzeitige Beendigung

Dieser Vertrag kann jederzeit mit sofortiger Wirkung beendet werden, wenn eine der Parteien gegen eine Bestimmung dieses Vertrages, insbesondere die Geheimhaltungspflicht verstößt.

§ 9 Vertraulichkeit

Die HTL Spengergasse verpflichtet sich während der Dauer dieser Vereinbarung und für 10 Jahre nach Beendigung derselben, sämtliche Informationen und Unterlagen von der Projektpartnerin sowie das vertragsgegenständliche Diplomprojekt (kurz „vertrauliche Informationen“) vertraulich zu behandeln und sie Dritten (abgesehen von § 7 oben) ohne Zustimmung von der Projektpartnerin weder bekannt zu geben noch zugänglich zu machen. Ferner verpflichtet sich die HTL Spengergasse diese Geheimhaltungsverpflichtung all denjenigen Mitarbeitern schriftlich aufzuerlegen, die während und nach dem

Diplomprojekt gemäß dieser Vereinbarung einen wie immer gearteten Zugang zu dem Projekt haben oder haben könnten (abgesehen von § 7 oben).

Für den Fall, dass vertrauliche Informationen Inhalt von weiteren schulischen Projekten bzw. in weiterer Folge von Studienarbeiten usw. werden sollen, die zuständigen Organen/Behörden des Bundes vorzulegen sind, ist die Projektpartnerin von solch einem Vorhaben vorab schriftlich zu informieren. Auf Verlangen von der Projektpartnerin wird die jeweilige Schule, FH oder Universität die Sperre dieser Schul- bzw. Studienarbeiten usw. für mindestens 2 Jahre beantragen.

Die Projektpartnerin verpflichtet sich ebenfalls für die oben genannte Dauer all jene Informationen die sie von der HTL Spengergasse erhält geheim zu halten.

Von dieser Vertraulichkeitsverpflichtung ausgenommen sind jedoch jene Informationen, die (i) ohne Verschulden der empfangenden Vertragspartei öffentlich bekannt werden oder vor Übermittlung an die empfangende Vertragspartei bereits öffentlich bekannt waren; (ii) vor Übermittlung an die empfangende Vertragspartei der empfangenden Vertragspartei nachweislich bekannt waren; (iii) durch die empfangende Vertragspartei von einem Dritten ohne Einschränkung oder Verletzung dieser Vereinbarung erworben wurden; (iv) von der empfangenden Vertragspartei ohne Kenntnis der geheim zu haltenden Informationen der anderen Vertragspartei selbstständig entwickelt wurden oder (v) von der herausgebenden Vertragspartei zur Veröffentlichung schriftlich freigegeben wurden.

§ 10 Veröffentlichungen

Veröffentlichungen des Diplomprojekts bzw. von den einzelnen Arbeitsergebnissen werden grundsätzlich nur mit vorheriger schriftlicher Zustimmung von der Projektpartnerin vorgenommen. Sollte die Projektpartnerin jedoch innerhalb von drei Monaten ab Einlangen des schriftlichen Ersuchens um Veröffentlichung oder Mitteilung der geplanten Veröffentlichung ihre Zustimmung nicht ausdrücklich schriftlich verweigern oder sollte sie keine Mitteilung übermitteln, so gilt die Zustimmung als erteilt. Die Projektpartnerin wird die Zustimmung nur dann verweigern, wenn Firmeninteressen durch eine Veröffentlichung beeinträchtigt würden.

§11 Sonstiges

Die Projektpartnerin und das Projektteam halten fest, dass durch diese Vereinbarung weder ein Arbeitsverhältnis, welcher Art auch immer, begründet wird noch eine solche Begründung beabsichtigt ist.

Die Mitglieder des Projektteams nehmen zur Kenntnis, dass sie für eventuelle Versicherungen oder eine gegebenenfalls notwendige Anmeldung zur Sozialversicherung selbst verantwortlich sind.

Sofern ein Projektmitglied während der Projektlaufzeit aus dem Projekt ausscheidet, wird das Projekt, wenn möglich, durch die übrigen Projektmitglieder fortgeführt und beendet.

Sollten einzelne Bestimmungen dieser Vereinbarung ungültig sein oder werden, so berührt das nicht die Gültigkeit der übrigen Bestimmungen. Anstelle einer ungültigen Bestimmung soll eine dem Sinn und wirtschaftlichen Zweck dieser Vereinbarung entsprechende gültige Bestimmung in Kraft treten.

Vor Abschluss dieser Vereinbarung getroffene Nebenabreden zwischen den Parteien, die den Inhalt dieser Vereinbarung betreffen, verlieren mit Abschluss dieser Vereinbarung ihre Gültigkeit.

Es wird einvernehmlich festgestellt, dass keine schriftlichen oder mündlichen Nebenabreden zwischen den Parteien bestehen. Änderungen und Ergänzungen dieser Vereinbarung bedürfen zu ihrer Wirksamkeit der Schriftform.

Auf diese Vereinbarung ist Österreichisches Recht anwendbar. Als Gerichtsstand für Streitigkeiten aus diesem Vertragsverhältnis wird die ausschließliche Zuständigkeit des sachlich zuständigen Gerichtes in Wien vereinbart.

Wien, am


Projektpartnerin
Ing. Mag. Christian Wenner

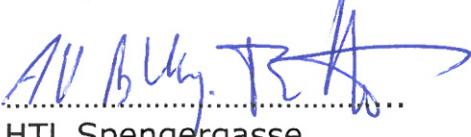

Dipl. Ing. Gerhard Schrott
Prokurst

Wien, am 21.10.2014


SGötz S. Hanke Kuhn Philipp Spengler
Projektteam (alle Mitglieder)

Die HTL Spengergasse bekundet, den gegenständlichen Vertrag zu kennen und verpflichtet sich ihrerseits zur Einhaltung der auf sie anwendbaren Bestimmungen dieser Vereinbarung.

Wien, am


HTL Spengergasse

905417

Höhere Bundes-Lehr- u. Versuchsanstalt
für TextilIndustrie Wien V
1050 Wien, Spengergasse 20

Functional Specification Document

KTI – Kapsch Tablet
Infrastructure

Sebastian Götze, Samuel Hammer, Michael
Kaufmann, Konstanze Müller, Philip
Steinhäuser

Content

1.	Requirement Specification	3
2.	Embedding in Organization	4
3.	Target Groups.....	5
4.	Service Requirements.....	6
4.1.	Hardening of mobile Enddevices.....	6
4.2.	Employees' perspective (eg.: production supervisor, service technicians)	7
4.3.	Company' perspective	7
4.3.1.	IT-Outsourcing companys.....	8
4.4.	Functional description.....	9
4.4.1.	Users.....	9
4.4.2.	IT-Leitung.....	9
4.4.3.	Kapsch.....	9
4.5.	Quality Criteria	10
4.6.	System architecture.....	11
4.6.1.	Hardware	11
4.6.2.	Software	13
4.7.	Functions	14
4.8.	Test cases	15
4.9.	Milestones	15
4.10.	Addition	16
4.11.	Glossary	16

History

Person	Date	Note
Steinhäuser, Müller	15.12.2014	Creating Functional Specification Document
Hammer, Götze, Kaufmann	16.12.2014	Making first notes for preperation
Hammer, Götze, Müller, Steinhäuser	20.12.2014	Finishing first version

1. Requirement Specification

Project objectives

Main objectives

- Identifying indicators, which need to be fulfilled by an Android based industrial solution
- Identifying, highlighting and describing different realization concepts
- Evaluation of selected concepts for prototyping
- Outline the results of the prototype concept

Non-objectives

- Developing a self-programmed solution
- Solve problems of a selected standard solution
- Prototype is a ready-to-sell product

2. Embedding in Organization

The product which Kapsch will generate from the result of our project, is going to be used by various groups of employees who will be increasingly active in the field. Our project aims at ensuring that the final product can be used by employees of Kapsch's as well as those customers and thus is an important part of the infrastructure.

The final product can be individually calibrated to the needs of customers, which means that the product for each customer receives its own expression.

Furthermore, it is not possible that customers or users are able to embed personal devices in infrastructure. This is only reserved to the devices, we and further more Kapsch calibrate and provide.



3. Target Groups

The target groups of our project are the future customers of Kapsch, which have decided to use tablets in their business process. It is to mention, that our end product is not the finale system solution for the costumers of Kapsch. The Kapsch BusinessCom AG will use our project as a pre-study of possible solution varieties for developing the real project.

Kapsch costumers are (efficient) companies of different industries. Logistics, building industries, industry in general and also the retailer sector are some examples. Employees of the building industries can use a technic device like a tablet for their specific workflow to reduce the organisation time of their paper. They are not forced to execute the elaborate process of scanning every important construction plan or some other specific document for sending it to the partners via email. Through this project solution employees would save a lot of time. Employees would have every document as a digital file. The whole process of sharing with other people will be faster than before. Saving time is the motive for companies in general to implement a tablet infrastructure with a specific software solution for their employees.

Companies of the retailer sector are interested in such software solutions to increase their own costumers offer. They anticipate a better costumer's satisfaction and thereby a better position on the market. For example McDonalds provide their customers tablets for surfing, using apps and also for playing games on the tablet in some of their restaurants.

4. Service Requirements

As from today's perspective, the use of mobile devices in the corporate sector has grown significantly, it is necessary to provide the user with a safe and easy platform for the implementation of business processes.

4.1. Hardening of mobile Enddevices

Such a platform requires a very safe hardening of the mobile end devices. The devices have to be both physically and software-side secured, to prevent data loss or abuse.

- **Physically**
 - It should not be possible to trigger certain hidden features of the mobile end device using a keyboard shortcut.
 - It should not be possible to reset the mobile end device using a key combination and so bypass the safety precautions.
 - If needed, the mobile end device must be protected from physical damage. (Falling-damage, etc.)
- **Software-side**
 - The mobile terminal should be protected against malicious and inadvertent changes by the user.
 - Users are not allowed to install their own apps and make adjustments on their own.
 - The user has only apps and setting options are available that have been determined in advance by the IT management.

4.2. Employees' perspective (eg.: production supervisor, service technicians)

If we now consider our product from employees' perspective, we see so far many advantages in availability than the 24 hours 7 days a week, our product is available.

By this we mean that the final product of Kapsch, is designed without an external cause for continuous operation and is to be used as such.

One limitation that affects the user is that it is not permitted by Kapsch or the person responsible for the product in the respective company, that the system should be changed consciously or unconsciously.

By Subconscious or unconscious change the system we mean:

- Conscious Change: the user has deliberately changed parts of the system.
- Unconscious change: the user should be due to the fault of IT management and system administrations of the company have more freedom than he should have.

Despite all of that users should not be hindered or restricted at work through these limitations.

The aim of our project or product is to facilitate the work of the user and not difficult it.

4.3. Company' perspective

For the evaluated software solution, there are various corporate perspectives from which the final product can be considered.

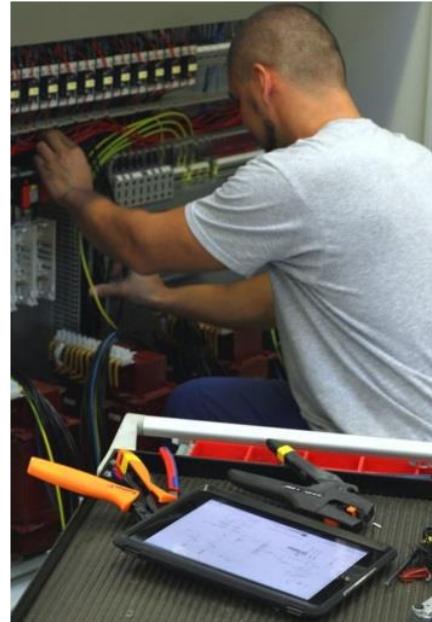
For the IT management some factors have a special importance, as these make their life a lot easier and therefore some more costs can be saved.

One of those factors is the remote maintenance. Current solutions had a lack of functionality in resetting devices from the company headquarter, which will be a major point of the evaluation process.

Until now, wrong configured devices needed to be sent back to IT—Management by post. This leaded to enormous maintenance costs and a huge time loss. Another important aspect is the remote management. Through this functionality of the evaluated solution the IT-Department is able to alter the devices content. This results in the possibility of the installation or deletion of applications or the possibility of a factory-reset, which deletes the device data unrestorable.

4.3.1. IT-Outsourcing companies

From the view of an IT-Outsourcing company, like Kapsch, the final solution will offer the possibility to use it for different types of customers and to adapt it individually. So one of the most important factors is the scalability of the concept and the possibility of using it in different environments. It must be usable for a construction company as well as for a trading or service company. The solution should be adaptable for each customer individually. For example, in a construction company it is necessary that specific applications are usable all the time. The rest of the system should be secured in a way that the employee is not able, accidentally or not, to put the device into an unusable state. The time of implementing a system is another important factor in the view of a providing company. This time should be kept as low as possible. As mentioned above flexibility should be given in a form that the solution is usable with 2 devices as good as with 200 devices.



4.4. Functional description

4.4.1. Users

For endusers it is a key fact that they can use the tablet specifically for his tasks. Neither more nor less. The standard android user experience should not be changed. The user is:

- Able to open and use ,only the from the company specified, applications
- Not able to alter the device settings
- Not able to install or delete any applications
- Able to reset the device to a pre-defined state
- Able to delete all of his usage data before leaving the device, like his browser history. (This is extremely important in the stationary trade.)

4.4.2. IT-Leitung

The IT management of a company must have access to different functions of the evaluated solution to ensure smooth operation and to gather important information about the use of the equipment. The central IT of the costumer should

- Multiple sites can manage devices centrally
- can read the status of the individual devices and locations
- Error messages can be read and analyzed
- Single / All devices, when needed, can be reseted to a defined state
- Statistics and reports on the individual Tablets / locations read
- install and delete apps on the Enddevices
- Guidelines for the use of the browser set the device settings

4.4.3. Kapsch

Due to the fact that Kapsch will distribute the solution to many different customers, it must also fulfill certain functions and standards so that Kapsch can wait customers' devices and manage them easily. Thus, the business solution must provide the following options for the client.

Kapsch can

- Adjust the platform for various clients individually
- The solution, according to the number of required equipment, scale to any size

4.5.Quality Criteria

- **Cost-effective**
- Nowadays all companies, which are using common software on their tablets for business processes have a big problem. Every time when the tablet of an employee does not work right as it should, the tablet has to be sent to Kapsch to fix it. In the most cases the employee itself is responsible for this. The result of this time-consuming process is that the employee is restricted of working on. So every tablet, which has to be sent to Kapsch make indirect costs for the company.
By the new modified software, the employees' wouldn't be able to crash the tablet software. So no tablet has to be fixed by Kapsch. For the companies it means that there are no indirect costs for repairing the employee's tablets.
- **Business-standard security**
- **Flexibility**
 - The end software product of Kapsch can be adaptable for every customer and their necessities. Kapsch can control which specific system configurations have to be activated and which are not needed by the customer. This specific system configuration can be for example activating companies necessary Apps. This results in one specific, individual software solution for every customer.

4.6. System architecture

4.6.1. Hardware

The Hardware component of the solution will be a standard Android Tablet. There is no requirement for a certain processor or amount of RAM, because the solution should be usable on any tablet available on the market. The only restriction is, that it should be capable of smoothly running the operating system. With choosing Android 4.2 as Operating System, it is necessary that the device fulfills the following requirements, set by Google. Therefore we refer on the following document:

<https://static.googleusercontent.com/media/source.android.com/de//compatibility/4.2/android-4.2-cdd.pdf>

4.6.1.1. Screen

- Screen size of at least 426 dp x 320 dp
- Screen aspect ratio of between 4:3 and 16:9
- A minimum of 120 dpi
- Devices MUST support dynamic orientation by applications to either portrait or landscape screen orientation
- Device implementations MUST support both OpenGL ES 1.0 and 2.0
- Screen Types
 - Fixed-pixel display implementations: the screen is a single panel that supports only a single pixel width and height. Typically the screen is physically integrated with the device. Examples include mobile phones, tablets, and so on.
 - Variable-pixel display implementations: the device implementation either has no embedded screen and includes a video output port such as VGA, HDMI or a wireless port for display, or has an embedded screen that can change pixel dimensions. Examples include televisions, set-top boxes, and so on.

Devices MUST support displays capable of rendering 16-bit color graphics

4.6.1.2. Input Devices

- MUST provide at least one soft keyboard implementation (regardless of whether a hard keyboard is present)
- The Home, Menu and Back functions are essential to the Android navigation paradigm. Device implementations MUST make these functions available to the user at all times when running applications
- Device implementations SHOULD have a pointer input system of some kind (either mouse-like, or touch).

4.6.1.3. Sensors

- Device implementations SHOULD include a 3-axis accelerometer.
- Device implementations SHOULD include a 3-axis magnetometer (i.e. compass.)
- Device implementations SHOULD include a GPS receiver.
- Device implementations SHOULD include a gyroscope (i.e. angular change sensor.)

4.6.1.4. Data Connectivity

- Device implementations **MUST** include support for one or more forms of data networking.
- Android 4.2 **MAY** be used on devices that do not include telephony hardware
- Android 4.2 device implementations **SHOULD** include support for one or more forms of 802.11 (b/g/a/n, etc.)
- Device implementations **SHOULD** include support for Wifi direct (Wifi peer-to-peer)
- Device implementations **SHOULD** include a Bluetooth transceiver.
- Device implementations **SHOULD** include a transceiver and related hardware for Near-Field Communications (NFC).

4.6.1.5. Cameras

- Device implementations **SHOULD** include a rear-facing camera, and **MAY** include a front-facing camera
- **Rear-facing Camera**
 - **MUST** have a resolution of at least 2 megapixels
- **Front-facing Camera**
 - **MUST** have a resolution of at least VGA (that is, 640x480 pixels)
 - **MUST** horizontally reflect (i.e. mirror) the stream displayed by an app in a Camera Preview.

4.6.1.6. Memory and Storage

- Device implementations **MUST** have at least 340MB of memory available to the kernel and userspace. The 340MB **MUST** be in addition to any memory dedicated to hardware components such as radio, video, and so on that is not under the kernel's control.
- Device implementations **MUST** offer shared storage for applications. The shared storage provided **MUST** be at least 1GB in size.

4.6.1.7. USB

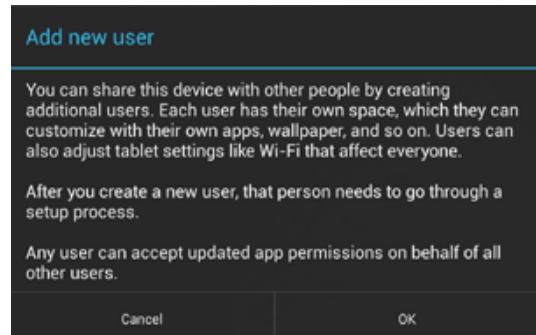
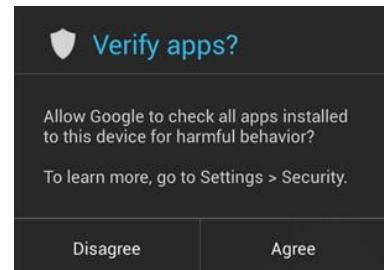
- Device implementations **SHOULD** include a USB client port, and **SHOULD** include a USB host port.
- **Client-port**
 - the port **MUST** be connectable to a USB host with a standard USB-A port
 - the port **SHOULD** use the micro USB form factor on the device side
- **Host-port**
 - it **MAY** use a non-standard port form factor, but if so **MUST** ship with a cable or cables adapting the port to standard USB-A

4.6.2. Software

4.6.2.1. Operating System (OS)

We decided on using Android 4.2+ as our operating system, because of the major security improvements made in this version of Android. It is also economically more sensible, due to the fact, that older versions need more maintenance and won't be supported anymore soon. The following features were the reason of not choosing Android 4.0+ as OS, like suggested by Kapsch:

- A new built-in scanning service, which checks installed app for a harmful behaviour. Whenever sideloading an App, the device sends information about the programm anonymously to Googles Servers, which then analyse the information and compare it with the info, stored in Googles Databases.
- An improved app-permission screen. It shows up, anytime you install an app outside of the Google Play store and gives you information about the requested permissions.
- A background feature, informing you every time an application tries to send a text message that could cost money. Typically when a message is sent to a fee-collecting short code, which is a number that automatically bills your carrier, when receiving a message.
- Multiuser support for Android tablets. This enables the device administrator to configure multiple user, each of whom maintains his own separate home screen, apps, wallpapers and general settings.



4.6.2.2. Mobile Device Management (MDM)

Another important part of the evaluated solution is the mobile device management system, which gives the IT-management the possibility to collect status information about the registered devices.

Mobile Management



Another feature is the management of applications, which can be grouped into specific user groups, like management, accounting, technical or anything else. It is not limited to a single type of end device, which means that tablets, smartphones, laptops, etc. can be managed parallel. It consists out of a server and a client component. The server typically stores all information and configurations and spreads them across the different devices. It also collects statistics

about the usage of each device. A client is a device, like the ones mentioned above, with a special application installed, that enables the connection to the MDM-server.

4.7.Functions

/L100/ Handling and workflow of the mobile end devices are retained for the user. This means that if knowledge with Android Tablets is present, the user itself does not need to get used to a new user interface.

/L200/ The user is able to call a set of predefined sites in the internal web browser and interact with them, if authorized by the IT management. This means subsequently that it is not allowed to call their own websites.

/L300/ The user can reset the Android tablet on the spot to a predefined state if necessary.

/L400/ The user is able to reset the mobile end device to a predefined state when leaving his work, to erase his tracks of use.

/L500/ The user has a predefined set of apps or applications. The apps are determined by the IT department, so the user has no control. That means, subsequently, that users are not allowed to install their own apps from the Internet or from Google Play Store.

/L600/ The device resets itself to a predefined state after a predefined time span of inactivity, to prevent data abuse.

4.8.Test cases

See the attachment: [Checklist Template 20141119 v0_5](#)

4.9.Milestones

MILESTONEPLAN				
WBS-Code	Milestone	Plan date	Revised date	Actual date
1.1.5	Project-Planning finished	06.11.2014	20.12.2014	20.12.2014
1.2.5	Preparation of templates, specifications and research work finished	19.12.2014	23.12.2014	20.12.2014
1.3.4	Creating research paper completed	30.01.2015	-	20.12.2014
1.3.6	Configuring prototype finished	20.02.2015	-	20.12.2014
1.5	Result presentation	15.05.2015	-	20.12.2014



Project handbook

KTI – Kapsch Tablet Infrastructure

001

Version 3.1
Project manager: Philip Steinhäuser
Date: 06/05/2015

Content

1 Project plans.....	5
1.1 Project Assignment	5
1.2 Project Objectives (objectives, non-objectives).....	6
1.3 Description of Pre- and Post Project Phase	7
1.4 Project Environment Analysis	8
1.5 Relationship to Other Projects and the Organisations´s Strategy	9
1.6 Project Organisation Chart	10
1.7 Plan of Objects of Consideration of the Project.....	11
1.8 Work Breakdown Structure (WBS)	12
1.9 Project Work-Package Specification	13
1.10 Project Responsibility Matrix	21
1.11 Milestoneplan	22
1.12 Project Bar Chart.....	23
1.15 Project Communication	24
1.16 Project „Rules“	25
1.17 Project Risk Analysis.....	26
1.18 Project Documentation.....	27
2 Project Co-ordination.....	28
2.1 Approval of Work-packages	28
2.2 Minutes – Project Co-ordination	29
3 Project Controlling	37
3.1 Project Status Report	37
4 Project Close Down	38
4.1 Project Close Down Report.....	38

Document versions

Versionno.	Date	Change	Author
1.0	29.9.2014	Start	Götze
1.1	24.10.2014	Project plans updated	Götze
1.2	30.10.2014	Project management handbook updated	Steinhäuser, Müller
1.3	31.10.2014	Project bar chart	Steinhäuser, Hammer
1.4	03.11.2014	Project Work-Package Specification	Steinhäuser
2.0	24.03.2015	Work packages and Project Bar Chart updated	Steinhäuser
3.0	10.04.2015	Status reports and minutes of meeting added	Steinhäuser, Hammer
3.1	06.05.2015	Project Close down updated	Steinhäuser

Distribution list

1 Project plans

1.1 Project Assignment

<p>KTI – Kapsch Tablet Infrastructure 001</p> <h1 style="text-align: center;">PROJECT-ASSIGNMENT</h1>							
<p>Project start event:</p> <ul style="list-style-type: none"> • Kick-off <p>project close down event in terms of content:</p> <ul style="list-style-type: none"> • Research paper completed <p>Formal project close down event:</p> <ul style="list-style-type: none"> • 1.4.2 Handover of the final results to the partner 	<p>Project start date:</p> <ul style="list-style-type: none"> • 13.10.2015 <p>Project close down dates:</p> <ul style="list-style-type: none"> • 15.05.2015 						
<p>Project objectives:</p> <ul style="list-style-type: none"> • Identifying indicators, which need to be fulfilled by an Android based industrial solution • Identifying, highlighting and describing different realization concepts • Evaluation of selected concepts for prototyping • Outline the results of the prototype concept 	<p>Non-objectives:</p> <ul style="list-style-type: none"> • Developing a self-programmed solution • Solve problems of a selected standard solution • Prototype is a ready-to-sell product 						
<p>Main tasks (Project phases):</p> <ul style="list-style-type: none"> • Project-Planning • Preparation on templates, specifications and research work • Study on feasibility concepts • Evaluation of selected concepts for prototyping • Result presentation 	<p>Project resources and costs*:</p> <table border="1"> <thead> <tr> <th>resource/type of cost</th> <th>unit</th> <th>Costs (€)</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	resource/type of cost	unit	Costs (€)			
resource/type of cost	unit	Costs (€)					
<p>Project owner:</p> <ul style="list-style-type: none"> • Kapsch BusinessCom AG 	<p>Project manager:</p> <ul style="list-style-type: none"> • Philip Steinhäuser 						
<p>Project team members:</p> <ul style="list-style-type: none"> • Sebastian Götze • Samuel Hammer • Michael Kaufmann • Konstanze Müller 							

* Possible categories of total Project budget:

Category A: up to 0.3 million Euro

Category B: up to 1 million Euro

Category B: up to 1 million Euro
Category C: up to 10 million Euro

Category D: more than 10 million Euro

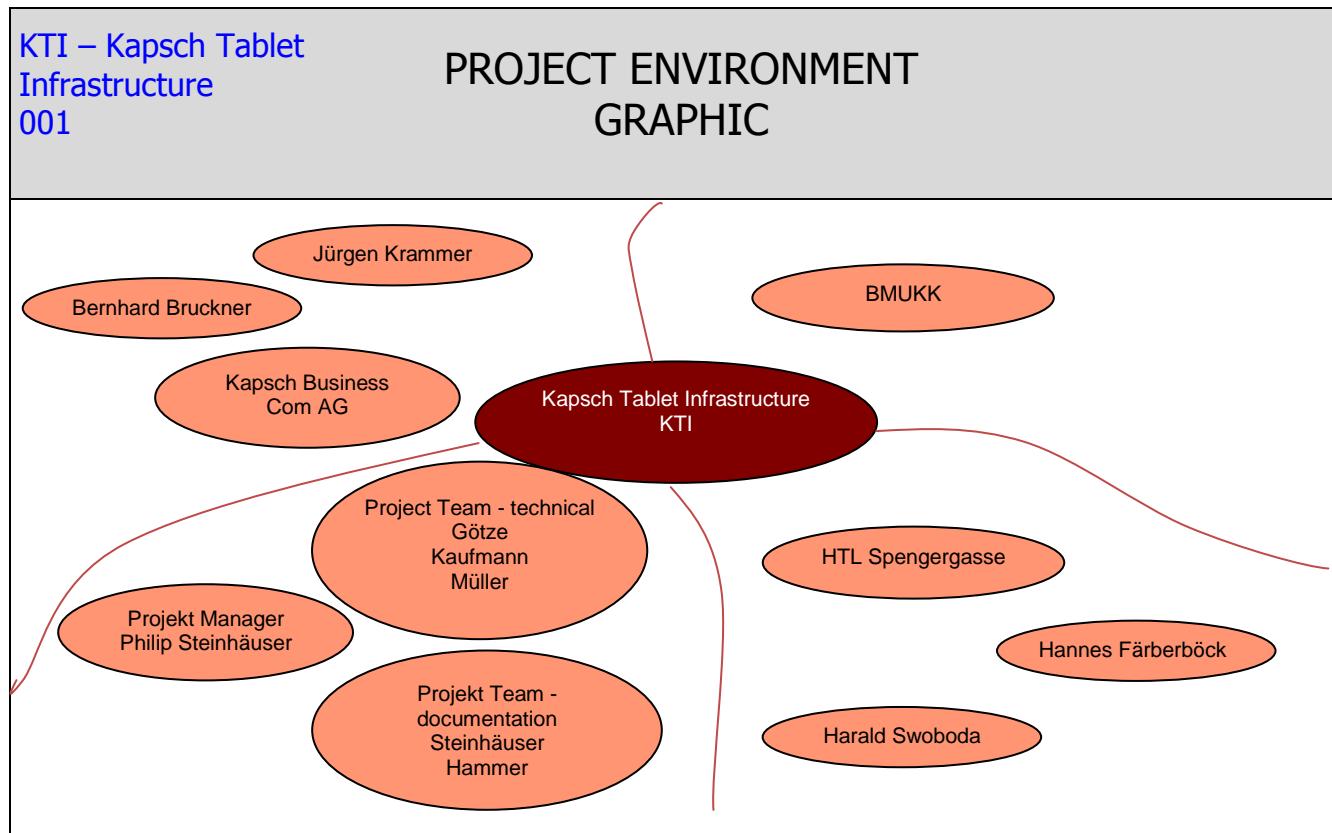
1.2 Project Objectives (objectives, non-objectives)

KTI – Kapsch Tablet Infrastructure 001		
PROJECT OBJECTIVES		
Type of objective	Project objectives	Adjusted project objectives as of...
objectives: <ul style="list-style-type: none"> • Main objectives • Additional objectives 	<ul style="list-style-type: none"> • Identifying indicators, which need to be fulfilled by an Android based industrial solution • Identifying, highlighting and describing different realization concepts • Evaluation of selected concepts for prototyping • Outline the results of the prototype concept • No additional objectives • 	<ul style="list-style-type: none"> •
Non-objectives	<ul style="list-style-type: none"> • Developing a self-programmed solution • Solve problems of a selected standard solution • Prototype is a ready-to-sell product • • 	<ul style="list-style-type: none"> •

1.3 Description of Pre- and Post Project Phase

<p><Project name></p> <p><Project no.></p>	<p>DESCRIPTION OF PRE- AND POST- PROJECT PHASE</p>
<p>1) Pre-project phase</p> <p><i>What triggered the project?</i></p> <p>Our project partner Kapsch provides enterprise grade solutions for major operating systems such as Microsoft Windows 8. Due to the Consumerization trend in industry platforms like Apple iOS and Android OS are on the rise in enterprise applications and leading system integrators like Kapsch are in need to understand those platforms, their applications and limitations. Based on experience and results from existing projects Kapsch has realized that for certain applications or scenarios the Apple iOS platform has its limitations and drawbacks. In particular industrial applications have a need for a rock-solid platform which enables a system integrator like Kapsch to operate a 24/7 application and service. So Kapsch is keen to expand their knowledge and experience towards the Android platform to be able to provide solution platform for industrial applications.</p> <p>Aim of this cooperation is to co-work on a set of different feasibility concepts for realizing such a platform based on Android.</p> <p><i>Relevant documents for the project („Minutes“, ... ONLY documents and no content necessary)</i></p> <ul style="list-style-type: none"> • Requirement specification from Kapsch • Meeting minutes from the pre-project phase <p><i>Experience from similar projects</i></p> <p>/</p>	
<p>2) Post-project phase</p> <p><i>What will happen after the project has ended? (follow-up activities, further projects, ...)?</i></p> <ul style="list-style-type: none"> • Following studies based on the results of the project by Kapsch • Detailed study on the selected concept for developing a ready-to-sell product by Kapsch 	

1.4 Project Environment Analysis



KTI – Kapsch Tablet Infrastructure 001

PROJECT ENVIRONMENT TABLE

Environment	Relationship (potential/conflict)	Measures	Who / when WBS Code
Kapsch	Project Owner	Project owner meeting	
HTL Spengergasse	School / marking authority	Project status report	
BMUKK	Juritically authority	Presentation of the result	1.4.4
Project Team	workforce	notes	

1.5 Relationship to Other Projects and the Organisations's Strategy

KTI – Kapsch Tablet
 Infrastructure
001

RELATIONSHIP TO OTHER PROJECTS

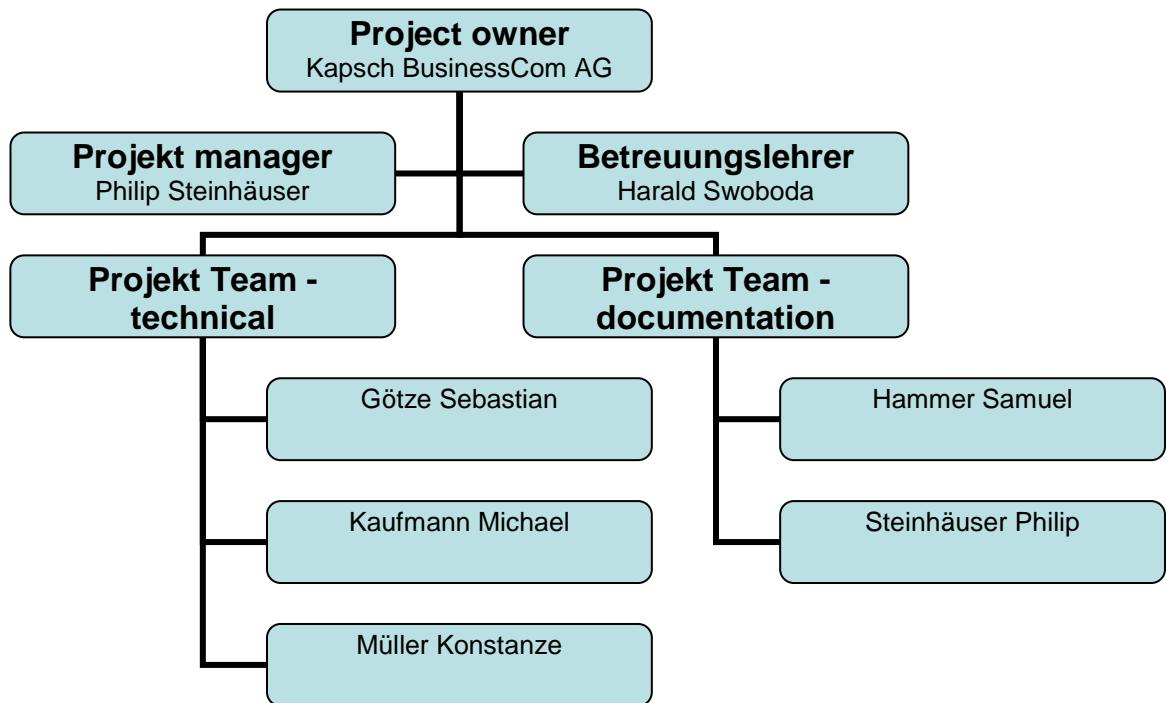
Programs/ Projects/	Relationship (potential/conflict)	Sanctions	Who / when WBS Code
/	/	/	/

KTI – Kapsch Tablet
 Infrastructure
001

CONNECTION TO THE ORGANISATION'S STRATEGY

Strategy	Description of connection/relationship
Company cooperation	The HTL Spengergasse wants to strengthen their connections to austrian industries. To give the students a more hands on experience on project work.

1.6 Project Organisation Chart

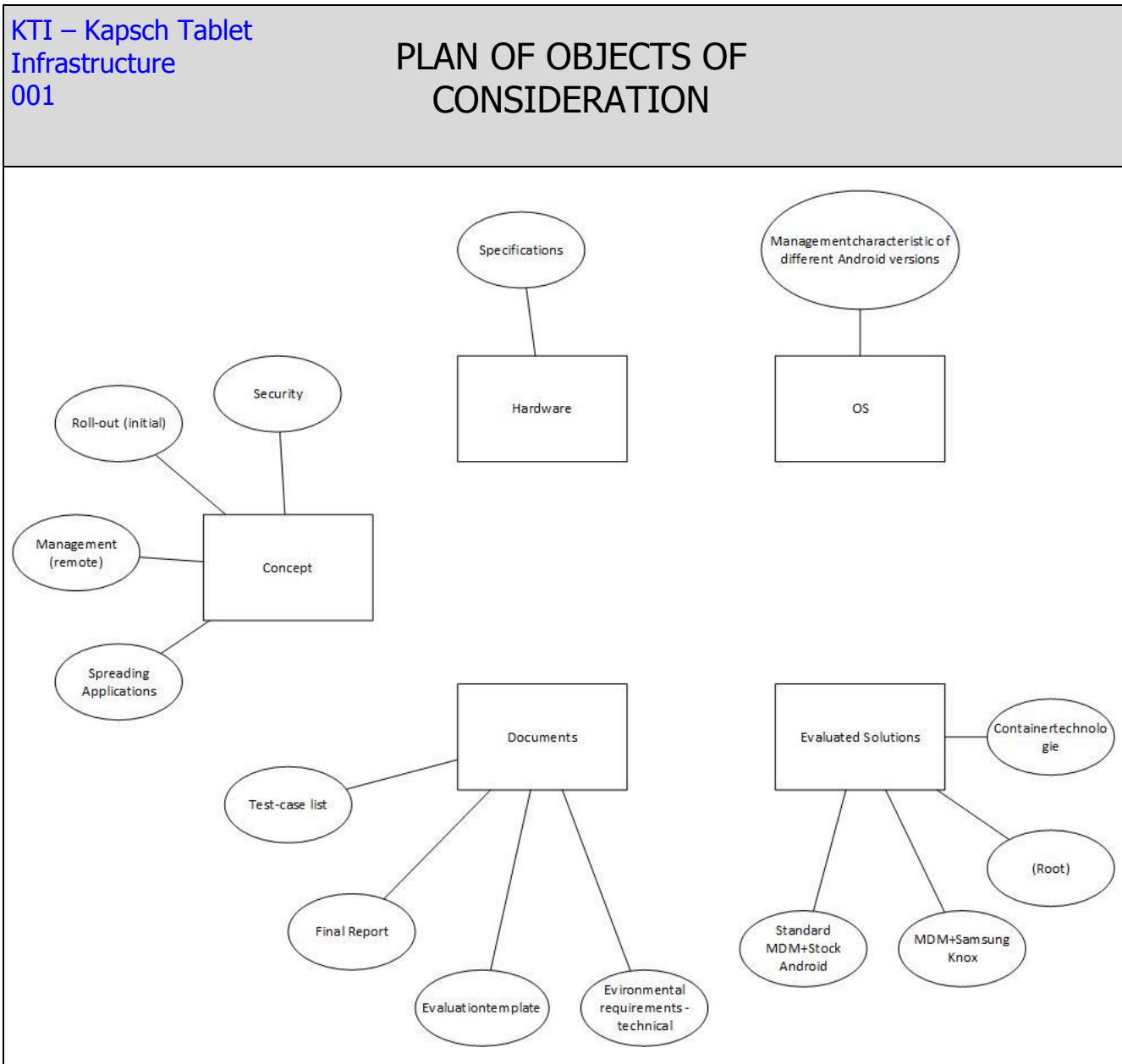


KTI – Kapsch Tablet
Infrastructure
001

PROJECT- ORGANISATION

Role in Project	Field of duties/Skills	Name
Project owner		Kapsch BusinessCom AG
Project manager		Philip Steinhäuser
Project team members		Sebastian Götze Samuel Hammer Michael Kaufmann Konstanze Müller Philip Steinhäuser
Project members		

1.7 Plan of Objects of Consideration of the Project



1.8 Work Breakdown Structure (WBS)

1. KTI

1.1. Project-Planning

- 1.1.1. Assignment of work packages and responsibility
- 1.1.2. Creating Project relevant Documents
- 1.1.3. Creating Timetable
- 1.1.4. Enumerating work packages
- 1.1.5. Project-Planning finished**

1.2. Preparation of templates, specifications and research work

- 1.2.1. creating of documentation templates
- 1.2.2. selection of specifications
- 1.2.3. research of varieties
- 1.2.4. configuring the tablet
- 1.2.5. Preparation on templates, specifications and research work finished**

1.3. Study on feasibility concepts

- 1.3.1. Testing the selected varieties
- 1.3.2. Creating the final concept
- 1.3.3. Creating Research paper
- 1.3.4. Creating research paper completed**
- 1.3.5. Configuring prototype
- 1.3.6. Configuring prototype finished**

1.4. Evaluation of selected concepts for prototyping

- 1.4.1. Perform final quality check
- 1.4.2. Handover of the final results to the partner
- 1.4.3. Acceptance test by teachers
- 1.4.4. Acceptance test by BMUKK

1.5. Result presentation

1.9 Project Work-Package Specification

<p>KTI – Kapsch Tablet Infrastructure 001</p>	<h3>PROJECT WORK-PACKAGE SPECIFICATION</h3>
<p>1.1.1, Assignment of work packages and responsibility</p>	<p>WP Content (<i>What shall be done?</i>)</p> <ul style="list-style-type: none"> The work packages should be defined by the following points <ul style="list-style-type: none"> WP Content Non-WP Content WP Result Progress Measurement
	<p>Non-WP Content (<i>What shall not be done? optional</i>)</p> <ul style="list-style-type: none"> The work packages shouldn't be defined another way
	<p>WP Result (<i>What is achieved after WP was finished?</i>)</p> <ul style="list-style-type: none"> All packages are clearly defined and well understandable
	<p>Progress Measurement (<i>How is progress measured?</i>)</p> <ul style="list-style-type: none"> Fill in all points of the PROJECT WORK-PACKAGE SPECIFICATION

<p>KTI – Kapsch Tablet Infrastructure 001</p>	<h3>PROJECT WORK-PACKAGE SPECIFICATION</h3>
<p>1.1.2, Creating Project relevant Documents</p>	<p>WP Content (<i>What shall be done?</i>)</p> <ul style="list-style-type: none"> Project management handbook Research Paper
	<p>Non-WP Content (<i>What shall not be done? optional</i>)</p> <ul style="list-style-type: none"> List of Tests Document of the hardware and android specification
	<p>WP Result (<i>What is achieved after WP was finished?</i>)</p> <ul style="list-style-type: none"> Project plans: Milestone plan, Project Environment Analysis, Project Organisation Chart, Work Breakdown Structure (WBS), Project Risk Analysis,... Project start
	<p>Progress Measurement (<i>How is progress measured?</i>)</p> <ul style="list-style-type: none"> Completion level of a single document

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.1.3,
Creating Timetable

WP Content (*What shall be done?*)

- Based on WBS
- Create the timetable in MS Project

Non-WP Content (*What shall not be done? optional*)

- Creating of the WBS

WP Result (*What is achieved after WP was finished?*)

- Time table – gives a overview on the working hours, which will be needed to finish the project in time.

Progress Measurement (*How is progress measured?*)

- The MS Project file will be created.
This document will be updated frequently.

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.1.4,
Enumerating work
packages

WP Content (*What shall be done?*)

- Based on WBS
- Every project member should be responsible at least one work package.
- These work packages will be separated between the two topic related teams.

Non-WP Content (*What shall not be done? optional*)

- Do the work of the work packages
- One person is responsible for all work packages.

WP Result (*What is achieved after WP was finished?*)

- responsibility matrix

Progress Measurement (*How is progress measured?*)

- after the discussion about the responsibility for certain packages, the results will be held in a responsibility matrix

PROJECT WORK-PACKAGE SPECIFICATION

1.2.1, Creating documentation templates	WP Content (<i>What shall be done?</i>) <ul style="list-style-type: none"> • creating empty templates for <ul style="list-style-type: none"> ◦ evaluation template ◦ activity report ◦ minutes of meeting
	Non-WP Content (<i>What shall not be done? optional</i>) <ul style="list-style-type: none"> • template for research paper
	WP Result (<i>What is achieved after WP was finished?</i>) <ul style="list-style-type: none"> • is a tool for enforcing a standard layout and look and feel across multiple pages or within content regions.
	Progress Measurement (<i>How is progress measured?</i>) <ul style="list-style-type: none"> • designing the standard layout for each of the documents

PROJECT WORK-PACKAGE SPECIFICATION

1.2.2, Selection of Specification	WP Content (<i>What shall be done?</i>) <ul style="list-style-type: none"> • contains a description of the hard- and software which will be used for testing and evaluating the different varieties
	Non-WP Content (<i>What shall not be done? optional</i>) <ul style="list-style-type: none"> • how we are using the hard- and software
	WP Result (<i>What is achieved after WP was finished?</i>) <ul style="list-style-type: none"> • Document of specification
	Progress Measurement (<i>How is progress measured?</i>) <ul style="list-style-type: none"> • Analyse the hard – and software requirements based on the final concept

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.2.3,
Research
varieties

of

WP Content (*What shall be done?*)

- The evaluation of the different software varieties which were defined in the Pre-Study

Non-WP Content (*What shall not be done? optional*)

- Configuring the evaluated variety on the tablet
- Creating a comparison of the varieties

WP Result (*What is achieved after WP was finished?*)

- Filled research templates

Progress Measurement (*How is progress measured?*)

- Filling in the determined information in the research templates

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.2.4,
Configuring
tablet

the

WP Content (*What shall be done?*)

- Configuring the tablet so that it could be used for the daily work

Non-WP Content (*What shall not be done? optional*)

- Configuring the evaluated variety on the tablet
- Testing the evaluated variety

WP Result (*What is achieved after WP was finished?*)

- Tablet which is usable for the daily use

Progress Measurement (*How is progress measured?*)

- You can follow the progress of this work-package by looking on our tablet

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.3.1,
Testing the selected
varieties

WP Content (*What shall be done?*)

- The evaluated software variety should be tested on the tablet to see if the features really work like expected.

Non-WP Content (*What shall not be done? optional*)

- If some features don't work like expected, find a solution to fix that.

WP Result (*What is achieved after WP was finished?*)

- A document, which provides an overview of what is working like expected and what isn't.

Progress Measurement (*How is progress measured?*)

-

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.3.2,
Creating the final
concept

WP Content (*What shall be done?*)

- Creating a concept which gives an overview about the following:
 - A security Concept
 - How the roll-out should be done
 - How the devices are manageable (per remote or have it to be local)
 - How to spread Applications

Non-WP Content (*What shall not be done? optional*)

- Creating the document described above for every software variety defined in the Pre-Study

WP Result (*What is achieved after WP was finished?*)

- A well structured document about the concept.

Progress Measurement (*How is progress measured?*)

-

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.3.3,
Creating Research
paper

WP Content (*What shall be done?*)

- Creating a document which contains a summary of the tested software solutions and our research work.

Non-WP Content (*What shall not be done? optional*)

- A complete documentation of our testing procedure

WP Result (*What is achieved after WP was finished?*)

- The paper is finished and sums up the entire research work of the project

Progress Measurement (*How is progress measured?*)

- How much of the research and testing work is already documented

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.3.5,
Configuring
prototype

WP Content (*What shall be done?*)

- Configuring the evaluated variety on the tablet

Non-WP Content (*What shall not be done? optional*)

- Testing the evaluated variety

WP Result (*What is achieved after WP was finished?*)

- A prototype/tablet configured with the evaluated variety

Progress Measurement (*How is progress measured?*)

- Configure the evaluated variety on the tablet

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.4.1,
Perform final quality
check

WP Content (*What shall be done?*)

- Project team should check, if the created documents and the prototype fulfil all of their requirements

Non-WP Content (*What shall not be done? optional*)

- Create new documents or make huge changes

WP Result (*What is achieved after WP was finished?*)

- The document creation is finished and matches the requirements of the project owner

Progress Measurement (*How is progress measured?*)

- The amount of reviewed documents

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.4.2,
Handover of the final results to the partner

WP Content (*What shall be done?*)

- The documents and prototype are handed over to Kapsch to declare the end of the project

Non-WP Content (*What shall not be done? optional*)

- Handover of the final result to the teachers or the BMUKK

WP Result (*What is achieved after WP was finished?*)

- Cooperation between project partner and team ends

Progress Measurement (*How is progress measured?*)

- Hand-over meeting performed or not?

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.4.3,
Acceptance test by teachers

WP Content (*What shall be done?*)

- The output of the the project should be graded by the responsible teachers

Non-WP Content (*What shall not be done? optional*)

- Alter anything
- Handover of the final result to the project partner (Kapsch BusinessCom AG)

WP Result (*What is achieved after WP was finished?*)

- The project is graded by the project teams teachers

Progress Measurement (*How is progress measured?*)

- Amount of reviewed project output

KTI – Kapsch Tablet
Infrastructure
001

PROJECT WORK-PACKAGE SPECIFICATION

1.4.4,
Acceptance test by
BMUKK

WP Content (*What shall be done?*)

- The Bundesministerium für Unterricht und Kultur grades our handed in project output.

Non-WP Content (*What shall not be done? optional*)

- Alter anything after handing it in

WP Result (*What is achieved after WP was finished?*)

- Project successful finished

Progress Measurement (*How is progress measured?*)

- Amount of reviewed project output

1.10 Project Responsibility Matrix

WBS-Code		WP-Title		PROJECT-RESPONSIBILITY-MATRIX							
		Roles & Environment		Kapsch BusinessCom AG	Steinhäuser Philip	Götze Sebastian	Hammer Samuel	Kaufmann Michael	Müller Konstanze	Harald Swoboda	Jürgen Krammer
1.1	Project-Planning	I	R	C		C		C	C	I	I
1.1.1	Assignment of work packages and responsibility	I	C	C	R		C		C	I	I
1.1.2	Creating project relevant documents	I	R	C	C		C		C	I	I
1.1.3	Creating Timetable	I	I	I	R		I		I	I	I
1.1.4	Enumerating work packages	I	R	C	C		C		C	I	I
1.2	Preparation of templates, specifications and research work	I	C	R	C		C		C	I	I
1.2.1	Creating of documentation templates	I	R	C	C		C		C	I	I
1.2.2	Selection of specifications	I	I	C	I		R		C	I	I
1.2.3	Research of varieties	I	I	C	I		C		R	I	I
1.2.4	Configuring the tablet	I	I	R	I		C		C	I	I
1.3	Study on feasibility concepts	I	I	C	I		R		C	I	I
1.3.1	Testing the selected varieties	I	I	C	I		R		C	I	I
1.3.2	Creating the final concept	I	I	C	I		C		R	I	I
1.3.3	Creating research paper	I	I	C	I		R		C	I	I
1.3.5	Configuring prototype	I	I	R	I		C		C	I	I
1.4	Evaluation of selected concepts for prototyping	C	R	C	C		C		C	I	C
1.4.1	Perform final quality check	C	R	C	C		C		C	I	C
1.4.2	Handover of the final results to the partner	C	R	C	C		C		C	I	C
1.4.3	Acceptance test by the teachers	C	R	C	C		C		C	I	C
1.4.4	Acceptance test by the BMUKK	C	R	C	C		C		C	I	C

Functions

R Responsible

C Contribution

I has to be informed

1.11 Milestoneplan

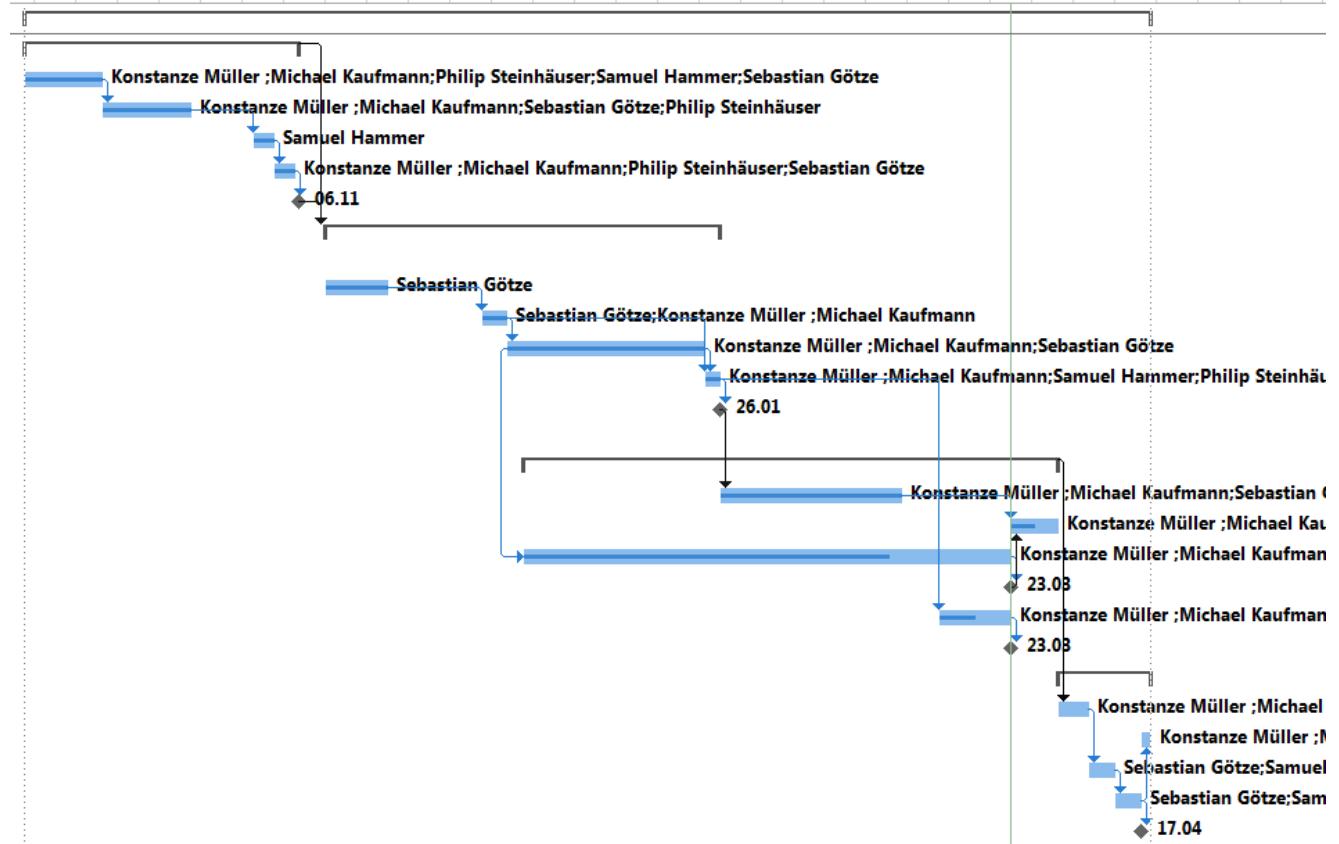
KTI – Kapsch Tablet Infrastructure 001		MILESTONEPLAN		
WBS-Code	Milestone	Plan date	Revised date	Actual date
1.1.5	Project-Planning finished	06.11.2014	20.12.2014	04.12.2014
1.2.5	Preparation of templates, specifications and research work finished	19.12.2014	23.12.2014	04.12.2014
1.3.4	Creating research paper completed	30.01.2015	1. 23.02.2015 2. 23.03.2015	23.03.2015
1.3.6	Configuring prototype finished	20.02.2015	23.03.2015	23.03.2015
1.5	Result presentation	15.05.2015		17.04.2015

*In order of plan dates.

1.12 Project Bar Chart

▲ KT1	217 dys	Mon 15.09.14	Sun 19.04.15
▲ Project planning	53 dys	Mon 15.09.14	Thu 06.11.14
Project planning	15 dys	Mon 15.09.14	Mon 29.09.14
Assignment of work packages and responsibility	17 dys	Tue 30.09.14	Thu 16.10.14 3
creating timetable	4 dys	Wed 29.10.14	Sat 01.11.14 4
Enumerating work packages	4 dys	Sun 02.11.14	Wed 05.11.14 5
Project planning finished	0 dys	Thu 06.11.14	Thu 06.11.14 6FS+1 dy
▲ Preparation on templates, specifications and research work	76 dys	Wed 12.11.14	Mon 26.01.15 2;7
Creating of documentation templates	12 dys	Wed 12.11.14	Sun 23.11.14
Selection of specifications	5 dys	Fri 12.12.14	Tue 16.12.14 9
Research of varieties	38 dys	Wed 17.12.14	Fri 23.01.15 10
configuring the tablet	3 dys	Sat 24.01.15	Mon 26.01.15 10;11
Preparation on templates, specifications and research work finished	0 dys	Mon 26.01.15	Mon 26.01.15 12
▲ Study on feasibility concepts	103 dys	Sat 20.12.14	Wed 01.04.15
Testing selected varieties	35 dys	Tue 27.01.15	Mon 02.03.15 13
Creating the final concept	9 dys	Tue 24.03.15	Wed 01.04.15 15;18
Creating research paper	94 dys	Sat 20.12.14	Mon 23.03.15 11SS+3 dys
Creating research paper completed	0 dys	Mon 23.03.15	Mon 23.03.15 17
Configuring prototype	14 dys	Tue 10.03.15	Mon 23.03.15 12
Configuring prototype finished	0 dys	Mon 23.03.15	Mon 23.03.15 19
▲ Evaluation of selected concepts for prototyping	18 dys	Thu 02.04.15	Sun 19.04.15
Performing final quality check	6 dys	Thu 02.04.15	Tue 07.04.15 14
Handover of final results to the partner	2 dys	Sat 18.04.15	Sun 19.04.15 25
Acceptance test by teachers	5 dys	Wed 08.04.15	Sun 12.04.15 22
Acceptance test by BMUKK	5 dys	Mon 13.04.15	Fri 17.04.15 24
Result presentation	0 dys	Fri 17.04.15	Fri 17.04.15 25

15 Sep '14 06 Oct '14 27 Oct '14 17 Nov '14 08 Dec '14 29 Dec '14 19 Jan '15 09 Feb '15 02 Mar '15 23 Mar '15 13 Apr '15 04 May '15
T W T F S S M T W T F S S M T W T F S S M T W T F



1.13 Project Communication

KTI – Kapsch Tablet Infrastructure 001		PROJECT- COMMUNICATION		
Title	Objectives, Content	Participants	Schedule	Location
Project owner meeting	<ul style="list-style-type: none"> • project status • decisions • acceptance of progress report 	Project owner, Project manager, Project team	2 x month	Kapsch BusinesswCom AG
Project controlling meeting	<ul style="list-style-type: none"> • project status • controlling of tasks, schedule, resources, costs • controlling of project environments • social controlling • prepare proposal for decision 	Project manager, Project team, Project coach	1 x week	HTBLVA Spengergasse
Project status report	<ul style="list-style-type: none"> • progress documentation • planned actions 	Project manager, Project team, Project coach, Project management officer (PMO)	1 x month	HTBLVA Spengergasse

1.14 Project „Rules“

1.1 Documents

1.1.1 File name

Every created Document needs to be saved with a specific file name and into a specific directory.

The filename exists out of the

<title>_<creation date>_<version number>.<file extension>

For example:

A MS-Project file defining the Gantt-Chart has been updated on the fifth of December.

So the file name is:

gantt_20141205_1.1.msproj

1.1.2 File Directory

For every type of project relevant document there is a directory containing all versions of this file. The directory name results out of the document. So for the Work breakdown structure the directory name will be WBS.

1.1.3 Distribution

The project team will share and work in a Dropbox-directory, saving files under the rules mentioned above. After finishing a certain document, the final version will be copied to the SharePoint-Server provided by Kapsch. The directory structure on the SharePoint will be the same, as it is in the Dropbox.

1.2 Activity report

Every project member has to leave an entry in their own activity report, after working on a project related topic. The members have to use the provided template in the Dropbox and are not allowed to change it, unless it has been discussed with the rest of the project team. In the template there is a single line for each entry. This line contains columns for an ID, WBS-Code, Date, Time from, Time to, Time Total and a description. Every column needs to be filled out!

1.3 Minutes of meeting

During a meeting one of the project members needs to take notes of the conversation. These should be kept as normal notices in a file of any form, but need to be summarized in a minutes of meeting afterwards. There standardised template, provided in the Dropbox, must be used, so that the files have the same structure.

1.4 Hand-ins

All hand-ins will be done by the project manager. In case of an absence of the manager this will be performed by the member Sebastian Götze or Konstanze Müller.

1.15 Project Risk Analysis

1.16 Project Documentation

Area	Description
File	<ul style="list-style-type: none"> • Kapsch SharePoint-Server • Dropbox
Access Authorisation	Provided by the project owner
Naming convention	Auto-generated by the Kapsch PMO
Rules	

2 Project Co-ordination

2.1 Approval of Work-packages

KTI – Kapsch Tablet Infrastructure 001		APPROVAL OF WORK-PACKAGES			
WBS- Code	Work-package	WP-Owner	Date	Approval by	Signature
1.1.1	Assignment of work packages and responsibility	Hammer	16.10.2014	Project coach	
1.1.2	Creating Project relevant Documents	Steinhäuser		Project coach	
1.1.3	Creating Timetable	Hammer	01.11.2014	Project coach	
1.1.4	Enumerating work packages	Steinhäuser	05.11.2014	Project manager	
1.2.1	Creating of documentation templates	Steinhäuser	23.11.2014	Project manager	
1.2.2	Selection of specifications	Kaufmann	16.12.2014	Project manager	
1.2.3	Research of varieties	Müller	23.01.2015	Project manager	
1.2.4	Configuring the tablet	Götze	26.01.2015	Project manager	
1.3.1	Testing the selected varieties	Kaufmann	02.03.2015	Project manager	
1.3.2	Creating the final concept	Müller	01.04.2015	Project manager	
1.3.3	Creating research paper	Kaufmann	23.03.2015	Project manager, KAPSCH	
1.3.5	Configuring prototype	Götze	23.03.2015	Project manager, KAPSCH	
1.4.1	Perform final quality check	Steinhäuser	07.04.2015	Project manager	
1.4.2	Handover of the final results to the partner	Steinhäuser	14.04.2015	KAPSCH	
1.4.3	Acceptance test by teachers	Steinhäuser	16.04.2015	Project coach	
1.4.4	Acceptance test by BMUKK	Steinhäuser	17.04.2015	BMUKK	

2.2 Minutes – Project Co-ordination

Besprechungsprotokoll Projekt KTI-Kapsch Tablet Infrastructur

Besprechungsprotokoll Projekt KTI-Kapsch Tablet Infrastructur				
Besprechungsprotokoll: HTL Spengerstrasse		Verteiler:		
Besprechungsdatum: 09.09.2014				
von: 17:00 bis: 18:20 Protokollführer: Samuel Hammer				
anwesend		zeitweise anwesend		
Name:		Name	von – bis	Unterschrift
Mag. Dipl.-Ing. Bernhard Bruckner Dipl.-Ing. Harald Swoboda Sebastian Götze Samuel Hammer Michael Kaufmann Konstanze Müller Philip Steinhäuser				
Lfd. Nr.	Arbeitspaket Nr. Kurzbezeichnung	Ergebnisse	erledigt Code	durch bis
01	Sprache der Dokumentation	Englisch		
02	Sprache der Verwaltungsoberfl.	Hinfällig		
03	Ziele definieren	geklärt		
04	Kooperationsvertrag besprechen	Wird unterschrieben		
05	Werksvertrag besprechen	Hinfällig		
06	SharePoint Zugang	in die Wege geleitet	T	KW 37
07	Projekt Ergebnis	geklärt		
08	Rechere des MDM	wird in den nächsten Wochen durchgeführt		
Code: A = Auftrag, B = Beschluss, E = Empfehlung, F = Feststellung, T = Termin				
Version:1.1		Datum: 18.09.2014	Ersteller/in: Müller Konstanze	Seite 1 von 1

Besprechungsprotokoll Projekt KTI-Kapsch Tablet Infrastructur					
Besprechungsprotokoll: Kapsch BusinessCom AG Besprechungsdatum: 13.10.2014 von: 15:25 bis: 17:30 Protokollführer: Konstanze Müller Datum des Protokolls: 13.10.2014			Verteiler:		
anwesend Name:	zeitweise anwesend Name	von – bis	Unterschrift		
Mag. Dipl.-Ing. Bernhard Bruckner Jürgen Krammer Sebastian Götze Samuel Hammer Michael Kaufmann Konstanze Müller Philip Steinhäuser					
Lfd. Nr.	Arbeitspaket	Ergebnisse	erledigt		
01	Allgemeine Besprechung des weiteren Vorgehens		Code	durch	bis
01	Brainstorming zur Feststellung der Betrachtungsobjekte des	Ein provisorischer OSP			
03	Besprechung der verschiedenen Varianten zur Absicherung eines Tablets	<ul style="list-style-type: none"> • Standard MDM+ • Stock Android MDM + • Samsung Knox • Root • Container 			
04	Besprechung von Anforderungen für erfolgreichen Abschluss des Projekts	<ul style="list-style-type: none"> • Untersuchungsbericht (Aufbereitung der Ergebnisse der Recherche) • Prototyp auf Tablet 			
05	Besprechung weiterer Aufgaben	OSP, PSP, Meilensteinliste bis zum nächsten Meeting	A	24.10.2014	
06	Festlegung des nächsten	24.10.2014, 09:00 Uhr	T		
Code: A = Auftrag, B = Beschluss, E = Empfehlung, F = Feststellung, T = Termin					

subject: Kapsch Kick-Off Meeting

date: 13.10. page: 1

Sicherheitssystem durch alle Schichten

10.000.000 Sicherheitsfunktionen

Samsung → **KNOX** reicht 8000

Liste von Anforderungen

Evaluierung von Varianten

Standard MDM + Android stock

→ FMDM + Samsung Knox

Dokumente: Abschlussbericht

Template: Evaluierungsdocument erstellen

Umgebungsbedingungen Varianten:

→ Technik

Knox arbeitet in der Zwischenwelt

Varianten

↳ Rooten → rechtliche Themen
↳ Knox

Das gesamte System am Ende

Android Tablet

• Device Specification

• Test Apps

• Managementplattform

• Managementcharakteristik der verschiedenen Android-Versionen

Konzept

- Roll-out
- Betriebsführung
- Applikationsverteilung
- Sicherheitskonzept

Standard

• FMDM + Samsung Knox

• (Root)

• Container für App



Container

OS

Tablet



To Do

~~OSP~~

PSP

Mitarbeiterstammlist

9⁰⁰ 24.10.

↳ schicken 22.10.



Abbildung 1-Mitschrift

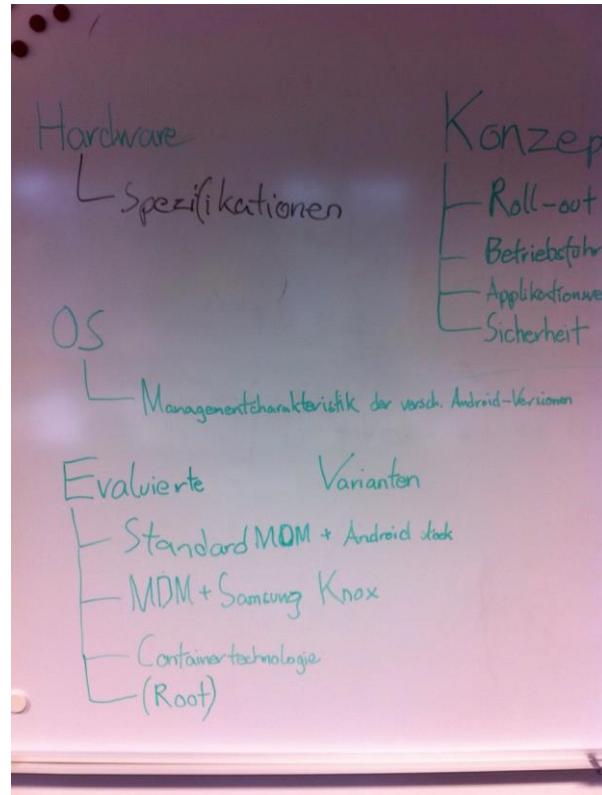


Abbildung 2-OSP #1

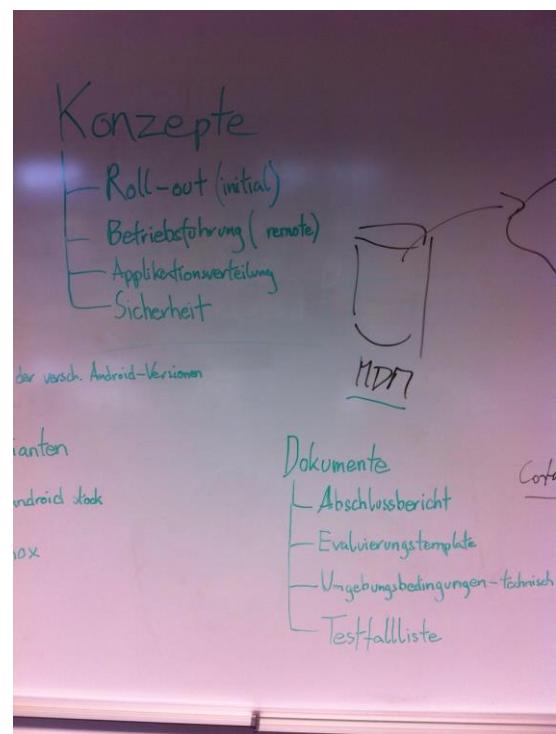


Abbildung 3-OSP #2

Besprechungsprotokoll Projekt KTI –Kapsch Tablet Infrastructur

Besprechungsprototyp: Kapsch BusinessCom AG Verteiler:

Besprechungsdatum: 24.10.2014

von: 09:00 bis:

10:00 Protokollführer/in: Philip

Steinhäuser

Datum des Protokolls: 24.10.2014

anwesend	zeitweise anwesend	Unterschrift
Name:	Name von – bis	
Mag. Dipl.-Ing. Bernhard Bruckner Jürgen Krammer Sebastian Götze Samuel Hammer Michael Kaufmann Konstanze Müller Philip Steinhäuser		

Lfd. Nr.	Arbeitspaket Nr. Kurzbezeichnung	Ergebnisse	erledigt	
			Cod e	durch bis
01	Allgemeine Besprechung des weiteren Vorgehens			
02	Klärung einiger aufgetretener Fragen	Tablet? → nach Abschluss der Projektplanung	F	
03	Anfertigen einer TODO –LISTE	Hilfestellung für Planung		
04	OSP besprochen	-		
05	PSP besprochen	Zeitplan in MS Project erstellen	F	
06	Punkt 1 des Projekthandbuchs durchgegangen und ausgefüllt	Ziele, Nicht-Ziele, Kann-Ziele, Projectphases, Pre project phase, Resourcen plan, Project communication, Risks, Project documentation	F	
07	Festlegung des nächsten Meetings	24.11.2014	T	24.11.2014

Code: A = Auftrag, B = Beschluss, E = Empfehlung, F = Feststellung, T = Termin

Version:1.0 Datum: 25.10.2014 Ersteller/in: Samuel Hammer Seite 1 von 1

Besprechungsprotokoll Projekt KTI		Logo		
Besprechungsdatum: 24.11.2014		Verteiler:		
von: 16:00				
bis 17:00				
Protokollführer/in: Samuel Hammer				
Datum des Protokolls: 25.11.2014				
anwesend	zeitweise anwesend	Unterschrift		
Name:	Name von – bis			
Samuel Hammer Philip Steinhäuser Michael Kaufmann Konstanze Müller Sebastian Götz Jürgen Krammer Bernhard Bruckner				
Lfd. Nr.	Arbeitspaket Nr. Kurzbezeichnung	Ergebnisse	erledigt	
			Cod e	durch bis
01	Besprechung Struktur der Diplomarbeit	Erste Struktur festgelegt	F	
02	Besprechung der Inhalte des Pflichtenhefts	Struktur für Pflichtenheft festgelegt	F	
Code: A = Auftrag, B = Beschluss, E = Empfehlung, F = Feststellung, T = Termin				
Version: 1.0	Datum: 25.11.2015	Ersteller/in: Samuel Hammer	Seite 1 von 1	

Besprechungsprotokoll Projekt KTI-Kapsch Tablet Infrastructur

Besprechungsprototypsort: Kapsch BusinessCom AG

Verteiler:

Besprechungsdatum: 15.12.2014

von: 16:30 bis:

17:30 Protokollführer/in: Philip

Steinhäuser

Datum des Protokolls: 16.12.2014

anwesend		zeitweise anwesend	Unterschrift		
Name:		Name	von – bis		
Mag. Dipl.-Ing. Bernhard Bruckner Jürgen Krammer Sebastian Götze Samuel Hammer Michael Kaufmann Konstanze Müller Philip Steinhäuser					
Lfd. Nr.	Arbeitspaket Nr. Kurzbezeichnung	Ergebnisse	erledigt	Cod e	durch bis
01	Allgemeine Besprechung des weiteren Vorgehens				
02	Erhalt von Informationen des Projektpartners zu den einzelnen Absicherungsmöglichkeiten	<ul style="list-style-type: none"> • Linux Manipulation • MDM Only • MDM + Container 	E		
03	Strukturierung der Diplomarbeit		E		
04	Erhalt von Infos zum Thema Android als Teilthema der Diplomarbeit	http://www.eetimes.com/document.asp?doc_id=1279698 http://net.cs.uni-bonn.de/fileadmin/user_upload/ploehmann/2012-Schulz-Android_Security_Common_Attack_Vectors.pdf https://source.android.com/devices/tech/security/index.html https://www.isc2cares.org/uploadedFiles/wwwisc2cares.org/Content/Android-Security-Report-FrostSullivan.pdf	E		
05	Festlegung des nächsten Meetings	19.01.2015	T		

Code: A = Auftrag, B = Beschluss, E = Empfehlung, F = Feststellung, T = Termin

Version:1.0 Datum: 16.12.2014 Ersteller/in: Samuel Hammer Seite 1 von 1

Besprechungsprotokoll Projekt KTI		Logo		
Besprechungsort: Kapsch BusinessCom		Verteiler:		
Besprechungsdatum: 19.01.2015				
von: 16:30				
bis: 17:30				
Protokollführer/in: Samuel Hammer				
anwesend		zeitweise anwesend		Unterschrift
Name:		Name	von – bis	
Samuel Hammer Konstanze Müller Philip Steinhäuser Sebastian Götze Michael Kaufmann Bernhard Bruckner		Jürgen Krammer	16:30 – 17:00	
Lfd. Nr.	Arbeitspaket Nr. Kurzbezeichnung		Ergebnisse	
01	Übergabe des Tablets		Tablet übergeben	T 19.01.2015
02	Besprechen der allgemeinen Vorgehensweise der			E
Code: A = Auftrag, B = Beschluss, E = Empfehlung, F = Feststellung, T = Termin				
Version: 1.0		Datum: 22.01.15	Ersteller/in: Samuel Hammer	Seite 1 von 1

3 Project Controlling

3.1 Project Status Report

Project Status Report Template

Project Name: KTI-Kapsch Tablet Infrastructure				
Prepared By: Philip Steinhäuser, Sebastian Götz				Date: 12/04/14
Reporting Period: From: <u>11/20/14</u> To: <u>12/04/14</u>	Type of Report: Team Report	Conclusions: Manageable issues exist.		
PLANNED TASKS FOR THIS REPORTING PERIOD				
Task Description	Start Date	Target End Date	Percent Complete	Task Status
1.Functional specification document	<u>11/24/14</u>	<u>12/20/14</u>	<50%	On Plan
2.Creating Time Table	<u>11/24/14</u>	<u>12/04/14</u>	50 - 74%	Not On Plan
3.Project management handbook	<u>11/12/14</u>	<u>12/04/14</u>	100%	On Plan
4.Creating documentation templates	<u>11/17/14</u>	<u>11/24/14</u>	100%	On Plan
5.	_____	_____	<50%	On Plan
6.	_____	_____	<50%	On Plan
Variance Details: The MS-Project Gantt-Chart is already created, but needs working associations between the workpackages. Also the resource-usage needs to be implemented.				
Corrective Actions: More intense work on the Gantt-Chart over the next days, to correct the mistakes.				
Objectives for the Next Reporting Period		From: <u>12/04/14</u>	To: <u>12/18/14</u>	
Functional specification document Implement ressource-usage into the project time table				
Notes: /				

Project Status Report Template

Project Name: KTI-Kapsch Tablet Infrastructure				
Prepared By: Sebastian Götze, Michael Kaufmann, Konstanze Müller				Date: 08/01/15
Reporting Period: From: <u>12/04/14</u> To: <u>01/08/15</u>	Type of Report: Team Report	Conclusions: Manageable issues exist.		
PLANNED TASKS FOR THIS REPORTING PERIOD				
Task Description	Start Date	Target End Date	Percent Complete	Task Status
1.Selection of specifications	<u>12/12/14</u>	<u>12/16/14</u>	100%	On Plan
2.Research of Varieties	<u>12/27/14</u>	<u>01/25/14</u>	<50%	Not On Plan
3. Creating research paper	<u>12/20/14</u>	<u>01/31/15</u>	<50%	Not On Plan
4.	_____	_____	Select	On Plan
5.	_____	_____	Select	On Plan
6.	_____	_____	Select	On Plan
Variance Details: According to Microsoft Project, task 2 and 3 are behind. This means the amount of time spent on the research work doesn't match it's actual completion state.				
Corrective Actions: Invest more time into the variety research				
Objectives for the Next Reporting Period		From: <u>01/08/15</u>	To: <u>01/22/15</u>	
Research of Varieties				
Notes: Our time table is now up to date. Also we have started with the system-research of the Samsung KNOX, MDM and MDM + Container and filled in our findings in the research template. Apart from that we searched for the hardware and software specification. Work on the research of varieties to finish it in time. We look forward to begin with the next workpackage "Configuring the Tablet" in the projectweek in the end of January.				



Project Status Report Template

Project Name: KTI-Kapsch Tablet Infrastructure				
Prepared By: Sebastian Götze, Samuel Hammer, Philip Steinäuser				Date: 01/22/15
Reporting Period: From: <u>01/08/14</u> To: <u>01/22/15</u>	Type of Report: Team Report	Conclusions: Manageable issues exist.		
PLANNED TASKS FOR THIS REPORTING PERIOD				
Task Description	Start Date	Target End Date	Percent Complete	Task Status
1.	_____	_____	Select	On Plan
2.	_____	_____	Select	On Plan
3.	_____	_____	Select	On Plan
4.	_____	_____	Select	On Plan
5.	_____	_____	Select	On Plan
6.	_____	_____	Select	On Plan
Variance Details: -				
Corrective Actions: -				
Objectives for the Next Reporting Period		From: <u>01/22/15</u>	To: <u>02/12/15</u>	
Research of Varieties Creating research paper setting up the Tablet				
Notes: reserved Tablet from our Project Partner first Tests with MobileIron (MDM)				

Project Status Report Template

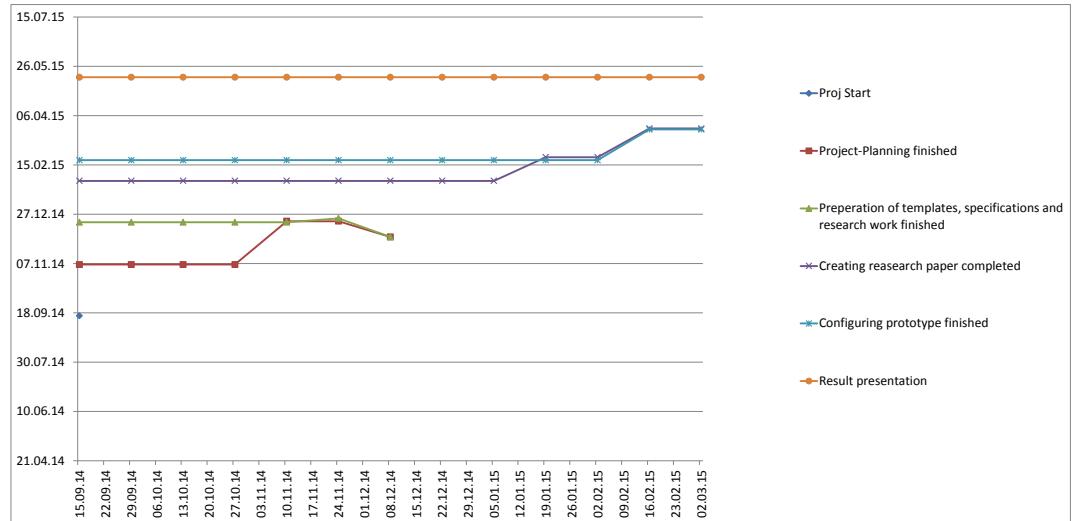
Project Name: KTI-Kapsch Tablet Infrastructure				
Prepared By: Konstanze Müller, Philip Steinhäuser				Date: 02/19/15
Reporting Period: From: <u>01/22/15</u> To: <u>02/19/15</u>	Type of Report: Team Report	Conclusions: Manageable issues exist.		
PLANNED TASKS FOR THIS REPORTING PERIOD				
Task Description	Start Date	Target End Date	Percent Complete	Task Status
1.Research of varieties	<u>12/17/14</u>	<u>01/23/15</u>	100%	On Plan
2.Configuring the tablet	<u>01/24/15</u>	<u>01/26/15</u>	100%	Ahead of Plan
3. Testing selected varieties	<u>01/27/15</u>	<u>02/05/15</u>	50 - 74%	Not On Plan
4.Creating research paper	<u>12/20/14</u>	<u>01/31/15</u>	<50%	Not On Plan
5.	_____	_____	Select	On Plan
6.	_____	_____	Select	On Plan
Variance Details: According to Microsoft Project, task 3 and 4 are behind. This means the amount of time spent on the Testing work as well as the creating of the research paper doesn't match it's actual completion state.				
Corrective Actions: Invest more time into testing and creating our research paper.				
Objectives for the Next Reporting Period		From: <u>02/19/15</u>	To: <u>03/19/15</u>	
Finish the testing of the selected varieties, Creating the final concept, Creating the research paper completed, Configuring prototype finished				
Notes: The Diploma thesis is finished about 60 %				



Project Status Report Template

Project Name: KTI-Kapsch Tablet Infrastructure				
Prepared By: Konstanze Müller, Philip Steinhäuser				Date: 03/05/15
Reporting Period: From: <u>02/19/15</u> To: <u>03/05/15</u>	Type of Report: Team Report	Conclusions: Manageable issues exist.		
PLANNED TASKS FOR THIS REPORTING PERIOD				
Task Description	Start Date	Target End Date	Percent Complete	Task Status
1. Testing selected varieties	<u>01/27/14</u>	<u>02/03/15</u>	100%	Not On Plan
2. Creating the final concept	<u>03/24/15</u>	<u>04/01/15</u>	50 - 74%	Ahead of Plan
3. Creating research paper	<u>12/20/14</u>	<u>03/23/15</u>	75 - 89%	Ahead of Plan
4.	_____	_____	Select	On Plan
5.	_____	_____	Select	On Plan
6.	_____	_____	Select	On Plan
Variance Details: According to Microsoft Project, task 1 is behind but due to the fact that this task is already finished it is not that important.				
Corrective Actions: -				
Objectives for the Next Reporting Period		From: <u>03/05/15</u>	To: <u>03/19/15</u>	
Finish the Creating the final concept, Creating the research paper completed and Configuring prototype finished.				
Notes: Adjusted our projectplan. Due to MSProject our Project is about 80 % finished.				

Contr.-Punkts	15.09.14	29.09.14	13.10.14	27.10.14	10.11.14	24.11.14	08.12.14	22.12.14	05.01.15	19.01.15	02.02.15	16.02.15	02.03.15	16.03.15	30.03.15	13.04.15	27.04.15	11.05.15
Proj Start	15.09.14																	
Project-Planning finished	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	06.11.14	
Preparation of templates, specifications and research work finished	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	
Creating research paper completed	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	
Configuring prototype finished	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	
Result presentation	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	

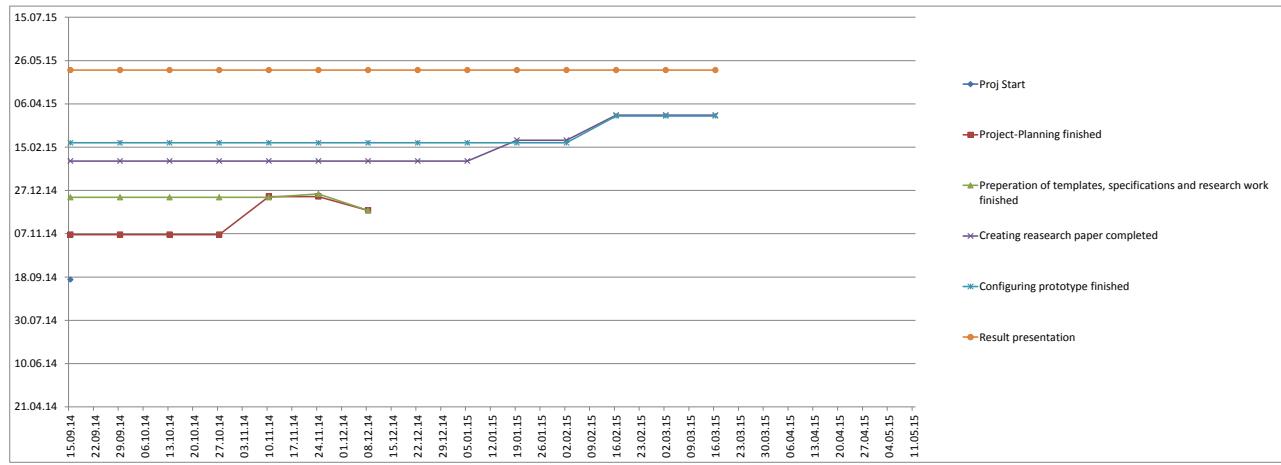




Project Status Report Template

Project Name: KTI-Kapsch Tablet Infrastructure				
Prepared By: Philip Steinhäuser				Date: 03/19/15
Reporting Period: From: <u>03/05/15</u> To: <u>03/19/15</u>	Type of Report: Team Report	Conclusions: Manageable issues exist.		
PLANNED TASKS FOR THIS REPORTING PERIOD				
Task Description	Start Date	Target End Date	Percent Complete	Task Status
1.Creating the final concept	<u>03/24/15</u>	<u>04/01/15</u>	50 - 74%	On Plan
2.Creating research paper	<u>12/20/14</u>	<u>03/23/15</u>	75 - 89%	On Plan
3.	_____	_____	Select	On Plan
4.	_____	_____	Select	On Plan
5.	_____	_____	Select	On Plan
6.	_____	_____	Select	On Plan
Variance Details: -				
Corrective Actions: -				
Objectives for the Next Reporting Period		From: <u>03/19/15</u>	To: <u>04/13/15</u>	
Finish the Creating the final concept, Creating the research paper completed and Configuring prototype finished.				
Notes: Due to MSProject our Project is about 80 % finished.				

	Conir-Punkt	15.09.14	29.09.14	13.10.14	27.10.14	10.11.14	24.11.14	08.12.14	22.12.14	05.01.15	19.01.15	02.02.15	16.02.15	02.03.15	16.03.15	30.03.15	13.04.15	27.04.15	11.05.15
Proj Start		15.09.14																	
Project-Planning finished		06.11.14	06.11.14	06.11.14	06.11.14	20.12.14	20.12.14	04.12.14											
Preparation of templates, specifications and research work finished		19.12.14	19.12.14	19.12.14	19.12.14	19.12.14	23.12.14	04.12.14											
Creating research paper completed		30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	30.01.15	23.02.15	23.02.15	24.03.15	24.03.15	24.03.15			
Configuring prototype finished		20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	20.02.15	23.03.15	23.03.15	23.03.15	23.03.15			
Result presentation		15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15	15.05.15		



4 Project Close Down

4.1 Project Close Down Report

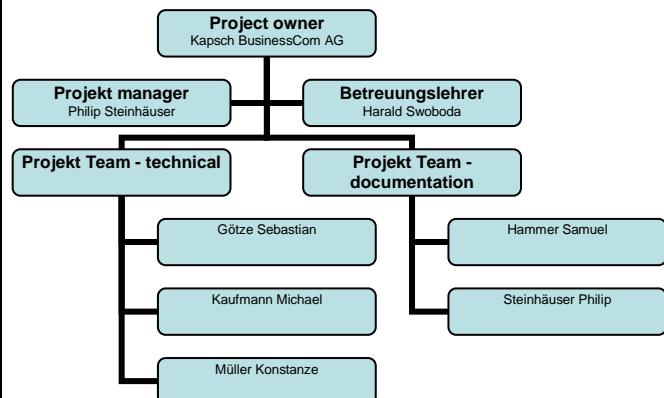
KTI – Kapsch Tablet Infrastructure 001	PROJECT CLOSE DOWN REPORT																																								
1) Overall impression <div style="display: flex; align-items: center; justify-content: space-around;"> between GOOD and OK </div> <div style="text-align: center; margin-top: 10px;">  /  </div>																																									
2) Reflection: Fulfilment of objectives <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Milestones</th> <th>Plane date</th> <th>Revised date</th> <th>Actual date</th> <th>Revised date</th> <th>Actual date</th> </tr> </thead> <tbody> <tr> <td>Project-Planning finished</td> <td>06.11.2014</td> <td>20.12.2014</td> <td>04.12.2014</td> <td></td> <td></td> </tr> <tr> <td>Preperation of templates, specifications and research work finished</td> <td>19.12.2014</td> <td>23.12.2014</td> <td>04.12.2014</td> <td></td> <td></td> </tr> <tr> <td>Creating research paper completed</td> <td>30.01.2015</td> <td>23.02.2015</td> <td>-</td> <td>23.03.2015</td> <td>23.03.2015</td> </tr> <tr> <td>Configuring prototype finished</td> <td>20.02.2015</td> <td>23.02.2015</td> <td>23.03.2015</td> <td></td> <td></td> </tr> <tr> <td>Result presentation</td> <td>15.05.2015</td> <td></td> <td>17.04.2015</td> <td></td> <td></td> </tr> </tbody> </table> <p> Green ... finished without replanning Orange ... finished after replanning Red ... did not finished </p>						Milestones	Plane date	Revised date	Actual date	Revised date	Actual date	Project-Planning finished	06.11.2014	20.12.2014	04.12.2014			Preperation of templates, specifications and research work finished	19.12.2014	23.12.2014	04.12.2014			Creating research paper completed	30.01.2015	23.02.2015	-	23.03.2015	23.03.2015	Configuring prototype finished	20.02.2015	23.02.2015	23.03.2015			Result presentation	15.05.2015		17.04.2015		
Milestones	Plane date	Revised date	Actual date	Revised date	Actual date																																				
Project-Planning finished	06.11.2014	20.12.2014	04.12.2014																																						
Preperation of templates, specifications and research work finished	19.12.2014	23.12.2014	04.12.2014																																						
Creating research paper completed	30.01.2015	23.02.2015	-	23.03.2015	23.03.2015																																				
Configuring prototype finished	20.02.2015	23.02.2015	23.03.2015																																						
Result presentation	15.05.2015		17.04.2015																																						
3) Reflection: Deliverables / Schedule <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Deliverables</th> <th>Schedule</th> </tr> </thead> <tbody> <tr> <td>Researchpaper</td> <td>16.04.2015</td> </tr> <tr> <td>Prototype</td> <td>16.04.2015</td> </tr> <tr> <td>Diplomathesis</td> <td>15.05.2015</td> </tr> </tbody> </table>						Deliverables	Schedule	Researchpaper	16.04.2015	Prototype	16.04.2015	Diplomathesis	15.05.2015																												
Deliverables	Schedule																																								
Researchpaper	16.04.2015																																								
Prototype	16.04.2015																																								
Diplomathesis	15.05.2015																																								

4) Reflection: Resources / Costs

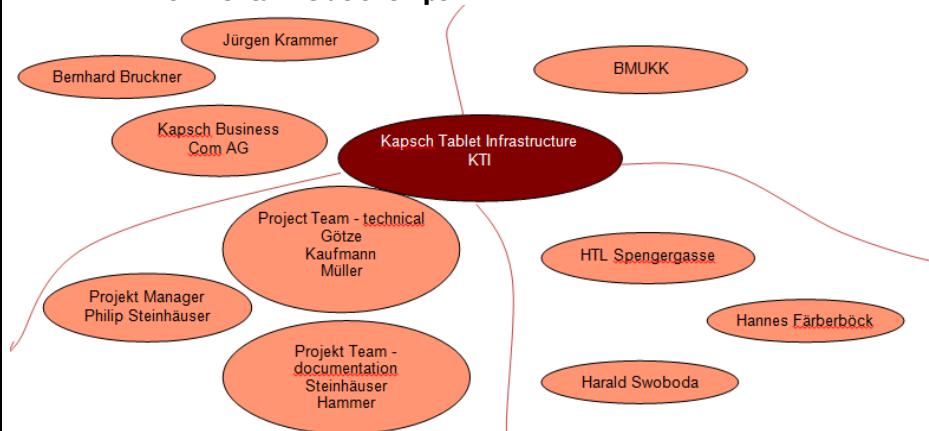
-

5) Reflection: Internal Organisation / Environmental Relationships

- Internal Organisation



- Environmental Relationships



6) Performance appraisal (Project owner, Project manager, Project member)

Stakeholder	Appraisal (x out of 3 Smileys)
Project owner – KAPSCH BusinessCom AG	3 smileys
Project manager – Philip Steinhäuser	3 smileys
Project member – Sebastian Götze	3 smileys
Project member – Samuel Hammer	2 smileys
Project member – Michael Kaufmann	2 smileys
Project member - Konstanze Müller	3 smileys

7) Lessons learned (Summary of Experiences and suggestions for improvement)

- Plan enough time buffer
- Don't rely on whitepapers
- Communication between sub teams is important
- Reading the installation manual, instead of working trial and error
- The amount of work in this project was hard to calculate

kapsch >>>

SPENGERGASSE 
ausbildung mit zukunft



UNTERSUCHUNGSBERICHT

KTI – Kapsch Tablet Infrastructure

Sebastian Götze
Samuel Hammer
Michael Kaufmann
Konstanze Müller
Philip Steinhäuser

HTBLVA Spengergasse, Spengergasse 20, 1050 Wien

1 Inhaltsverzeichnis

2 Einleitung	3
3 Linux Manipulation	4
3.1 Allgemein	4
3.2 Schlussfolgerung	4
4 Mobile Device Management (MDM).....	5
4.1 Allgemein	5
4.2 Mobile Device Management Standard	5
4.3 Informationen der getesteten Software.....	6
4.4 Installation	6
4.5 Features	7
4.5.1 Key-Features	7
4.5.2 Zusatzfeatures.....	8
4.6 Testphase und -Ergebnisse	9
4.6.1 Erfüllte Anforderungen	9
4.6.2 Nicht erfüllte Anforderungen	12
5 MDM + Container	14
5.1 MobileIron Containertechnologie	14
5.2 Allgemeine Information der getesteten Software.....	14
5.3 Installation	14
5.4 End User Products	15
5.4.1 Docs@Work	15
5.4.2 Web@Work.....	15
5.4.3 Apps@Work.....	16
5.5 Übereinstimmung mit den Anforderungen des Projektpartners	17
5.5.1 Zusätzlich erfüllte Anforderungen	17
6 Samsung Knox	18
6.1 Allgemeine Infos zu Samsung Knox	18
6.2 Samsung Knox Bestanteile.....	18
a. Plattform Security.....	18
b. Applikation Security.....	20
c. Management (MDM)	21
6.3 Unterstützte Geräte	23
6.4 Erfüllte Funktionen.....	24
6.4.1 Kernfunktionen.....	24
6.4.2 Zusätzliche Funktionen	25
6.5 Fehlende Funktionen	26
6.5.1 Inhaltseinstellungen(Apps,Websiten, Marketing)	26

6.5.2	Benutzer und Geräte	26
6.5.3	Managementseitige Anforderungen	26
7	Endergebnis und Empfehlung	27
7.1	Evaluierung	27
7.2	Auswertung der Evaluierung.....	30
7.2.1	Allgemein	30
7.2.2	Inhaltseinstellungen.....	30
7.2.3	Statistik.....	30
7.2.4	Benutzer und Geräte	30
7.2.5	Managementseitige Anforderungen	30
7.2.6	Zusätzliche Anforderungen	30
7.3	Empfehlung	31

2 Einleitung

Dieser Untersuchungsbericht dient dazu der Firma Kapsch BusinessCom AG eine Empfehlung für ein bei deren Kunden einzusetzendes Managementsystem für Tablets auf Basis Android auszusprechen. Diese Plattform muss gewissen Vorgaben des Unternehmens entsprechen, welche in der Anfangsphase des Projekts definiert wurden. Nach anfänglichen Untersuchungen konnte das Projektteam vier Kandidaten für Betriebsplattformen in die engere Auswahl ziehen. Diese vier Systeme wurden bezüglich ihres Nutzes für die Absichten der Firma Kapsch untersucht, dokumentiert und in diesem Untersuchungsbericht zusammengefasst. Bei diesen potentiell einsetzbaren Systemen handelt es sich um:

- Linux Manipulation
- Mobile Device Management
- Mobile Device Management + Container Technologien
- Samsung Knox

Diese werden in diesem Bericht detailliert dargestellt und es wird auf deren spezifische Stärken und Schwächen eingegangen. Am Ende stellt eine Nutzwertanalyse einen Vergleich der Systeme dar, an Hand derer das abschließende Fazit und die damit einhergehende Empfehlung erfolgen. Damit kann die Kapsch BusinessCom AG dann entscheiden, ob und wenn wie sie das System für ihre zukünftigen Projekte einsetzen können.

3 Linux Manipulation

3.1 Allgemein

Dieser Teil befasst sich mit der Veränderung des Grundsystems eines jeden Android Geräts. Dieses Grundsystem basiert auf dem Open Source Betriebssystem Linux und existiert dabei in einer für Mobilgeräte angepassten Form. In seiner Standardimplementation bietet es zwar einige Funktion zur Erhöhung der Gerätesicherheit, jedoch nicht genügend, um es in einem betrieblichen Umfeld einzusetzen. Da Linux ein Open Source System ist, darf der Source Code von jedem angesehen und nach eigenen Wünschen verändert werden. Und genau hier setzt die Methode der Linuxmanipulation an. In dem man den Source Code so verändert, dass bestimmte Teile des Betriebssystems unzugänglich gemacht oder verschlüsselt werden, ist es möglich den späteren Benutzer vor unabsichtlichen Änderungen am System zu bewahren, welche den reibungslosen Betrieb stören könnten. Das bedeutet, dass dadurch ein vollkommen an die Bedürfnisse des Kunden angepasstes Betriebssystem möglich wäre. Um sich einen besseren Eindruck davon zu verschaffen, wie diese Manipulationen letztendlich aussehen, lohnt es sich auf die diversen frei am Markt erhältlichen Derivate zu werfen. Bekannte Beispiele dafür wären:

- CyanogenMod
- Android AOSP
- Paranoid Android
- Dirty Unicorns
- Etc.

Zwar sind diese nicht mit securitytechnischen Absichten entwickelt worden, aber sie zeigen trotzdem auf was mit einer Menge an Entwicklungsaufwand möglich ist. Da aber durch die tiefgreifenden Eingriffe in das System eines Android Gerätes auch die Garantieansprüche verloren gehen, ist das Projektteam zu einem schnellen Fazit zu kommen.

3.2 Schlussfolgerung

Die Methode der Linuxmanipulation ist für die Zwecke der Kapsch leider absolut nicht einsetzbar da sich bei ihr gewisse Konflikte bezüglich Garantieanspruch und Kosten ergeben. Zwar wären über diese Methode sämtliche Anforderungen an das Endprodukt erfüllbar, jedoch bedarf es dazu eines so großen Entwicklungsaufwands, dass dieser sich in einem dermaßen hohen Endkundenpreis niederschlagen würde, welcher von kaum einem Unternehmen zu bezahlen wäre. Denn nicht nur die Beschäftigung einer Vielzahl von Entwicklern über einen langen Zeitraum hinweg, sondern auch der anfallende Support für das Produkt würde sich selbst für ein großes Unternehmen wie Kapsch nicht rentieren. Ein weiteres großes Manko dieser Methode ist die verfallende Garantie für die Hardware. Egal mit welchen Android Tablet diese Software ausgeliefert werden würde, sobald eine Veränderung der darauf vorinstallierten Software stattfindet, gehen sämtliche Garantieansprüche an den Hersteller verloren. Bei der geplanten Menge an ausgelieferten Geräten durch die Kapsch, wäre dies nicht vertretbar. Für einen industriellen Einsatzzweck ist die Variante daher absolut nicht geeignet und der damit einhergehende Aufwand würde sich nur durch einen extrem hohen Verkaufspreis ausgleichen lassen. Somit bleibt dem Projektteam nichts anderes zu sagen, als, dass diese Variante nicht passend für die Absichten der Kapsch ist.

4 Mobile Device Management (MDM)

4.1 Allgemein

Die folgenden Zeilen beschäftigen sich mit den Einsatz von Mobile Device Management Systemen als Betriebsplattform für potentielle Kunden der Firma Kapsch. Durchgeführt wurden alle Untersuchungen am bereits am Markt etablierten MDM-System MobileIron. Es wird beleuchtet was der MDM-Standard ist, welche Funktionen für den alltäglichen Gebrauch unumgänglich sind und wie diese im System realisiert sind. Ein besonders wichtiger Punkt hierbei ist auch das Aufzeigen von nicht vorhandenen Funktionen, die jedoch für das Unternehmen Kapsch von fundamentaler Wichtigkeit sind. Des Weiteren wird auf die Bedienbarkeit und die Komplexität der Installation eingegangen und inwiefern dies für den Projektauftraggeber und dessen Kunden relevant ist. Abschließend wird ein Statement abgegeben, ob bzw. wie es möglich ist diese Form von System für die von Kapsch gedachten Zwecke einzusetzen.



Abbildung 1 Logo MobileIron

4.2 Mobile Device Management Standard

MDM bezeichnet einen von der Open Mobile Alliance (OMA) festgelegten industriellen Standard zur Verwaltung mobiler Endgeräte wie zum Beispiel Smartphones, Tablets oder Laptops. Es dient dazu die allgemeine Verwaltung einer Vielzahl von Geräten zu erleichtern und somit Zeit und Kosten zu sparen. Die mobile Hardware kann dabei vom Unternehmen zur Verfügung gestellt werden oder, sofern mit dem Mitarbeiter abgesprochen, von diesem selbst mitgebracht werden. „Bring Your Own Device“ (BYOD) nennt sich dieser Ansatz. Dieser Standard wird in der Software verschiedenster Hersteller implementiert, welche dann dieses Komplettsystem verschiedenen Unternehmen zur Verwaltung ihrer Geräte anbieten. Beispiele dafür sind.

- MobileIron
- Samsung EMM
- Cisco Meraki
- MaaS360
- AirWatch
- etc.

Bestandteile dieser Softwarelösung sind eine Serverkomponente und die verschiedenen mobilen Clients. Der Server dient dabei dazu die Konfigurationen und Statistiken für die Geräte zu halten und zu verwalten. Er sendet auch, auf dem MDM-Standard basierende, Management Kommandos an die Clients aus, wenn sich ein Parameter in deren Konfiguration verändert hat. War es am Anfang der MDM-Systeme noch notwendig das Gerät dazu physisch mit dem Server zu verbinden, geschieht dies heute vollautomatisch über Netzwerkverbindungen. Die implementierten Funktionen können zum Beispiel eine over-the-air (OTA) Verteilung von Applikationen, Daten oder Konfigurationen sein. So braucht ein Administrator nicht auf 100 Geräten das Wifi-Netzwerk einrichten, sondern kann per Knopfdruck diese Konfiguration auf alle im System registrierten Geräte verteilen. Auch in Punkto Sicherheit bieten MDM-Systeme einige Möglichkeiten und deshalb sind sie so interessant für die Zukunftspläne der Firma Kapsch. So bieten diese Systeme die Möglichkeit Passwortrichtlinien zu setzen oder sogar ganze Teile des Betriebssystems zu sperren, damit diese für den Benutzer nicht zugänglich sind. Dadurch soll die Anfälligkeit für Fehler im Berufsumfeld gesenkt und ein ordentlicher Arbeitsablauf genehmigt werden.

4.3 Informationen der getesteten Software

Name:	MobileIron EMM
Hersteller:	MobileIron, 415 East Middlefield Road, Mountain View, CA 94043
Version:	7.5.0
Datum:	27.1.2015
Preis:	/
Website:	http://www.mobileiron.com/
Dokumentation:	https://support.mobileiron.com/eval/

4.4 Installation

Die Installation von MobileIron gestaltet sich relativ einfach, wobei doch einige wichtige Dinge zu beachten sind. Nach dem Download einer Datei aus dem Online-Zugangsportal von MobileIron, kann über diese das Betriebssystem installiert werden. Diese gestaltet sich für erfahrene Nutzer sehr einfach, allerdings ist auf folgende Dinge acht zu geben:

Folgende Daten müssen bereit stehen bzw. eingerichtet werden:

- Lizenzerinformationen (Firma, Kontaktperson, E-Mail)
- IP-Adresse
- Externer Hostname (**Sehr wichtig, weil die mobilen Geräte den Server von außerhalb erreichen müssen**)
- Command Line Interface Passwort
- Administratorname und – passwort
- Mind. 1 physisches Interface
- Subnetzmaske
- Default Gateway
- Mind. 1 zu erreichender DNS-Server
- Wahlweise
 - SSH-Zugriff
 - Telnet-Zugriff
 - NTP

Ist die Einrichtung erfolgt, kann man das System nach einem Neustart bereits einsetzen. Während dem Evaluierungsprozess sind dem Projektteam allerdings einige wichtige Details aufgefallen. Ein funktionierender externer Hostname ist von höchster Wichtigkeit, weil ohne ihn zwar die Einrichtung der Software auf den mobilen Endgeräten funktioniert, leider jedoch die Verbreitung von Konfigurationen versagt. Da jedoch 99 Prozent aller modernen Unternehmen über solche Möglichkeiten verfügen sollten, dürfte dies im realen Betrieb weniger problematisch ausfallen. Hervorzuheben ist hierbei die hervorragende Dokumentation, die MobileIron für den Installationsprozess zur Verfügung stellt. Auf deren Website findet sich eine Sammlung an Dokumenten, welche den Administrator am Anfang zwar überwältigen könnten, aber sich als eine schnell zu durchforstende Sammlung an bebilderten Skripten zur Einrichtung sämtlicher Funktionen herausstellen. Generell lässt sich die Webplattform, welche MobileIron hier zur Verfügung stellt, gut bedienen und bietet Informationen zu den verschiedenen Implementierungsszenarien und Komponenten des Systems. So findet man sich nach kurzer Zeit bereits relativ gut zurecht und weiß wo man suchen muss, um die benötigte Information zu finden.

4.5 Features

4.5.1 Key-Features

In diesem Teil werden die wichtigsten von MobileIron gebotenen Features beleuchtet und erklärt. Die folgende Liste stellt die wichtigsten Funktionen dar, welche während des Evaluierungsprozesses festgestellt werden konnten:

- MDM System
- Management von verschieden vielen Geräten
 - Jeder Gerätetyp ist möglich, egal ob Smartphone, Tablet oder Laptop
 - Lokalisierung sämtlicher eingebundener Geräte
 - Leicht aufzusetzen und zu verwenden
 - Installation einer einzigen App ist notwendig, um das Gerät in die Plattform einzubinden
 - Verwendbar ab dem ersten Gerät
 - Mehrsprachig
 - Geringe Wartungskosten
- Konfiguration der eingebundenen Endgeräte
 - Vorkonfiguration von Email-Konten und sonstigem (Wifi, VPN, etc.)
 - Der Angestellte muss dies nicht selbst erledigen
 - Keine Chance einer Fehlkonfiguration
 - Verteilbar auf hunderte Geräte innerhalb von Sekunden
- Statistiken
 - Verfügbare Statistiken
 - Gerätestatus
 - Kompromitierungsstand
 - Betriebssystem
 - Betriebssystemversion
 - Zugehörigkeit (gehört dem Unternehmen oder dem Angestellten)
 - Netzbetreiber (3, A1, Telering, etc.)
 - Registrierungszustand
 - Logging von Events:

Device Actions	App -	Policy -	Space -	Status -
Register	Install	Activate	Add Space	Not started
Wipe	Uninstall	Modify	Remove Space	In progress
Lock	Set setting	Deactivate	Change Space Prioritization	Completed successfully
Retire	Unset setting		Assign Space Admin Delete Space Admin	Failed

- Policies
 - Dienen zur Erhöhung der Sicherheit von registrierten Mobilgeräten
 - Blockieren von Systemteilen oder Einstellungen
 - Zum Beispiel: Der Benutzer kann das WLAN oder GPS nicht mehr ausschalten
 - Diese Funktion wird benötigt, wenn die Chance besteht, dass der Anwender durch gewollte oder ungewollte Aktionen das Gerät in einen unbenutzbaren Zustand bringt.
 - Passwortpflicht
 - Der User ist dazu gezwungen, ein Passwort nach Unternehmensrichtlinien zum Sperren und Entsperren seines Gerätes zu setzen
 - Globaler Proxy
 - Sämtlicher Netzwerkverkehr wird durch einen Proxy-Server des Unternehmens geleitet, welche dazu dient ungewollten Websiteinhalte zu filtern oder um Unternehmensdaten vor dem Verlassen des Firmennetzwerks zu schützen.
 - Kiosk-Modus
 - Das Gerät wird in einen Zustand versetzt, in dem nur mehr das Benutzen einer einzelnen Applikation möglich ist.
 - Applikationen
 - Erlauben von spezifischen Apps
 - Verbieten –
 - Erfordern –

4.5.2 Zusatzfeatures

MobileIron bietet die Möglichkeit seine Vielfalt an Features zu erweitern, indem man es an eine sogenannte „Standalone Sentry“ anbindet. Diese ermöglicht es dem Administrator des MDM-Systems die von MobileIron entwickelten Applikationen auf allen Geräten zu installieren. Diese Applikationen dienen dazu den E-Mail-Verkehr, das Webbrowsing, die Installation von Applikationen und den Zugang zu Unternehmensdokumenten zu sichern. Sie heißen:

- Apps@Work
- Docs@Work
- Web@Work

Nachdem sich das Projektteam in diese vertieft hatte, hat es erkannt, dass diese Applikationen eine sogenannte Containertechnologie einsetzen und somit nicht Bestandteil eines standardmäßigen MDM-Systems sind. Daher werden diese in einem anderen Teil des Evaluierungsprozesses behandelt, welcher „MDM+Container“ heißt. Eine weitere Zusatzfunktion, die MobileIron bietet, sind „ActiveSync Policies“. Diese stellen dabei einfach nur durchgehend upgedateete Policies dar, welche bei einer Verletzung sofort Alarm schlagen.

4.6 Testphase und -Ergebnisse

Während der Testphase wurde das System eingehend getestet und festgestellt, ob es den Anforderungen der Firma Kapsch genügt. Hierbei wurden vom Projektteam verschiedenste Szenarien durchgespielt und diverse Arten von Konfigurationen getestet. Im folgenden Abschnitt werden die Ergebnisse der Untersuchungen und deren Auswirkungen präsentiert.

4.6.1 Erfüllte Anforderungen

1. Nutzung ohne öffentliche Cloud-Services

Das getestete Endgerät ist vollständig ohne öffentliche Cloud Services einsetzbar. Das bedeutet, dass kein Google-Konto oder ähnliches notwendig ist, um es zu verwenden. Natürlich entfallen dadurch auch bestimmte Features wie der Google Play Store zur Installation von Applikationen.

2. Keine Zuweisung zu einer spezifischen Person notwendig

Es war dem Projektteam möglich, das Tablet einzusetzen ohne es einem konkreten Projektmitglied zuzuweisen. Jeder aus dem Team konnte es gleichermaßen ohne jegliche Einschränkungen verwenden. Dabei bleibt natürlich zu beachten, dass die Passwörter und Zugangsdaten allen Benutzern mitzuteilen sind. Dies kann über Wege wie eine Textdatei, Notizzettel oder Ähnlichem geschehen.

3. Verhinderung der Installation von Applikationen

Über die verschiedenen Policies, welche das System bietet, ist es möglich die Installation einzelnen oder allen Applikationen zu blockieren. Dies verhindert, dass der spätere Anwender ungewollte Apps installiert, welche den Zustand des Systems verändern könnten und es somit für den Arbeitsgebrauch unbrauchbar machen.

4. Keine Zugriff auf die Einstellungen

Durch eine Policy war es dem Projektteam möglich, sämtlichen Zugriffe auf die Einstellungen zu sperren, egal ob über die App oder über die Taskleiste am oberen Bildschirmrand. Diese Möglichkeit ist deshalb so wichtig, um den Anwendern davon unerwünschte Änderungen durchzuführen.

5. Verstecken der Standard-Applikationen

Diese Anforderung konnte die MDM-Lösung nicht vollständig umsetzen. Dem Projektteam ist es zwar gelungen einige einzelne Standardapplikationen auszublenden (Google Play, YouTube, Browser), allerdings war es nicht für alle Systemapps möglich. Zu diesen nicht auszublenden Softwarepaketem gehören die Galerie, Gmail, alle weiteren Google Play Apps (Music, Movies, etc.), die Kontakte, Maps und noch einige weitere. Dies muss nicht unbedingt zu Probleme im Betriebseinsatz führen, stellt aber trotzdem ein nicht zu unterschätzendes Risiko dar.

6. Filtern von Browserinhalten

Diese Funktionalität ist über die Nutzung eines globalen Proxy-Servers gewährleistet. Dieser ist gesondert vom Unternehmen einzurichten und bietet alle Möglichkeiten wie Sie ein herkömmlicher Proxy bietet.

7. Remote-Zugriff

8. Verwalten von Applikation per Remote

MobileIron bietet die Möglichkeit zu überprüfen, welche Apps auf einem Gerät installiert sind und es ist bei Bedarf möglich bestimmte Applikationen zu sperren bzw. sie als erforderlich einzustufen.

9. Festlegen von Hintergrund und Lockscreen per Remote

Diese Funktionalität ist auch nur teilweise gegeben. Folgendes ist dabei aus den Tests des Projektteams hervorgegangen: Es ist möglich einen Lockscreen und ein entsprechendes Passwort vom User zu verlangen. Auch das Sperren von Widgets am Lockscreen ist ohne weiteres möglich. Das Festlegen und Ändern des Hintergrundbilds ist mit MobileIron jedoch nicht möglich.

10. Logs der UP-/Downtime

Über die Weboberfläche des MDM-Systems ist es möglich zu überprüfen, wie lange ein Gerät mit dem System verbunden war oder auch nicht.

11. Logs sortierbar nach Gerät und Ort

Für jedes im System registrierte Mobilgerät werden separate Logs generiert, über die die letzten Vorgänge verfolgt werden können. Diese sind sehr leicht abzurufen und werden auch übersichtlich dargestellt.

12. Zurücksetzen des Geräts auf einen spezifischen Zustand durch den Benutzer

Diese Funktion ist in MobileIron nur begrenzt implementiert, weil es zwar die Möglichkeit gibt das Gerät auf den Werkszustand zurückzusetzen, aber nicht auf einen vom Unternehmen definierten Zustand. Gerade dies wäre für größere Kunden von der Kapsch sehr wünschenswert, weil so die Mitarbeiter bei einer Fehlfunktion des Geräts, dieses einfach wieder auf einen funktionstüchtigen Zustand zurücksetzen können. Somit ist die Funktionalität leider nur teilweise gegeben.

13. Management mehrerer Gerätestandorte

Das Management mehrerer Standorte ist ohne weiteres möglich, weil im MDM-System Funktionen eingebettet sind Geräte nach Standort zu sortieren oder ihnen spezielle „Tags“ zu verleihen, über die sie bzw. ihr Standort identifiziert werden können.

14. Ortung (inkl. Status) der Geräte über ein Dashboard

Auch die Ortung einzelner Endgeräte stellt mit MobileIron kein Problem dar. Zu jedem eingetragenen Tablet, Smartphone oder Tablet gibt es per Rechtsklick einen Eintrag „Locate“, über welchen das Gerät ausfindig gemacht werden kann. Dies funktioniert natürlich nur, wenn das Gerät mit dem Internet oder GPS verbunden und dabei eingeschalten ist.

15. Fehlerberichte pro Endgerät und Standort

Für sämtliche Geräte ist es möglich einen Fehlerbericht zu erhalten, sollte z.B. bei der Einrichtung etwas fehlschlagen oder das Mobilgerät nicht den Unternehmensrichtlinien entsprechen. So lässt sich genau erkennen, warum es zu dem jeweiligen Fehler gekommen ist.

16. Zurücksetzen des Geräts auf einen spezifischen Zustand per Remote-Zugriff

Wie bereits in Punkt 12 beschrieben lassen sich in das System eingebundene Geräte leider nur auf den Werkszustand zurücksetzen. Dies trifft ebenfalls auf das Management per Remote zu. Dem Administrator ist es somit über die Weboberfläche nur möglich das Endgerät auf den Auslieferungszustand zu setzen.

17. Zugriff auf die Statistiken über Remote

Da sämtliche Statistiken im Onlineportal gesammelt werden und nicht auf den Geräten selbst, ist es selbstverständlich, dass diese auch dort abgerufen werden können. Die Informationen können somit von jedem Rechner, mit Zugriff auf Management-Backend von MobileIron, aufgerufen werden.

18. Nutzbar ohne Kiosk-Modus

Um die Geräte im System nutzen zu können ist es nicht erforderlich sie in einem Modus schalten zu müssen, in dem nur eine einzige Applikation verwendbar ist. Die Haptik entspricht exakt der eines normalen Devices.

19. Nicht zugeschnitten auf eine spezifische Hardware

Das gesamte System ist speziell im Hinblick auf Android mit jedem Gerät ab Version 2.3 verwendbar. Jedoch ist dabei zu beachten, dass nicht sämtliche Funktionen auf allen Geräten zur Verfügung stehen. So ist dem Projektteam bei seinen Untersuchung aufgefallen, dass ein LG Tablet weniger Konfigurationsmöglichkeiten bietet, als ein Samsung Gerät. Dies ist darauf zurück zu führen, dass das Samsung Tablet einen speziellen Chip eingebaut hatte, welcher es erlaubte einen tieferen Eingriff in das System durchzuführen. Somit sollte beim Einsatz dieses Systems darauf geachtet werden mit welchen Endgeräten es betrieben wird.

20. Geringe Wartungskosten

Ein wahrer Vorteil dieses MDMs sind die nahezu nicht vorhandenen Wartungskosten. Ist das System einmal installiert und eingerichtet so läuft es praktisch ohne jegliche weitere Wartung. Lediglich die Lizenzkosten an das Unternehmen sind jährlich abzutreten. Dies stellt natürlich eine große Kostensparnis da.

21. Filtern von expliziten Inhalten

Wie bereits in Punkt 6 beschrieben ist dies über den Einsatz eines globalen Proxy möglich. Somit können die Endgeräte frei von Gewalt, Extremismus, Pornographie, etc. gehalten werden.

22. Mehrsprachig

MobileIron bietet sein MDM in verschiedenen Sprachen an, jedoch ist es trotzdem zu empfehlen das System in Englisch zu verwenden, da auch die Dokumentation in dieser Sprache geschrieben ist und somit die Installation und Einrichtung dadurch deutlich vereinfacht wird.

4.6.2 Nicht erfüllte Anforderungen

1. Benutzer können Apps nicht deinstallieren

Aus den Untersuchungen des Projektteams ist hervorgegangen, dass das Deinstallieren von Applikationen nicht ausreichend verhindert werden kann. Es ist zwar möglich die Installation von Unternehmensapplikationen zu blockieren, jedoch werden diese automatisch wieder installiert, sobald sie deinstalliert wurden. Dies könnte jedoch zum Verlust von Daten führen. Für Applikationen, welche nicht vom Unternehmen bereitgestellt werden, ist es nicht möglich deren Deinstallation zu verhindern. Für einen Einsatz in einem professionellen Arbeitsumfeld ist das leider nicht ausreichend, weil dadurch eine große Anzahl von Problemen auftreten kann. Dies ist ein kritischer Punkt in der Systemauswahl und deshalb wurde diese Funktion vom Projektteam als nicht erfüllt bewertet.

2. Websites/Links können per Remote verwaltet werden

MobileIron bietet ohne Einsatz von Containertechnologien leider nicht die Möglichkeit die Webseiten welche, auf dem Gerät geöffnet sind, über Remote zu verwalten. Auch das Öffnen eines speziellen Links lässt sich nicht realisieren. Unter gewissen Umständen könnte das zu größeren Komplikationen in einem Unternehmensumfeld führen. In einem Lokal könnte somit nicht gewährleistet werden, dass immer die Website des Lokalbetreibers geöffnet ist.

3. Inhalte können per Remote verwaltet werden

Die Möglichkeit Medien und andere Daten auf dem Gerät von außerhalb zu verwalten, ist leider nicht gegeben. Dies ist jedoch ein Punkt von äußerster Wichtigkeit, weil diese Funktion für die Einsatzzwecke eines Tablets im Businessalltag von Nöten ist. Ob es nun dazu gedacht ist um einen Manager die Umsätze des letzten Monats zu liefern oder einem Baumeister die Baupläne zur Verfügung zu stellen, diese Funktionalität muss gewährleistet sein, um das Gerät professionell einsetzen zu können.

4. Logs von Benutzersessions, Nutzungszeit, App-Aufrufen und Webseiten

Das Mitschreiben der Dauer von Benutzersessions oder der Nutzungsdauer von Apps, etc. wird von MobileIron leider nicht unterstützt. Zwar ist dieser Punkt nicht so kritisch wie die vorherigen, aber zu Analysezwecken wäre es für Unternehmen doch sehr interessant zu wissen, wie sich ihre User verhalten, um Rückschlüsse auf Probleme oder Unklarheiten zu ziehen.

5. Möglichkeit zur Löschung der Benutzerdaten durch den User

Auch eine Funktion zur Sicherheit der Anonymität eines Benutzers bringt MobileIron leider nicht mit. Zwar ist es dem User zwar möglich seinen Browserverlauf durch das umständliche Suchen in Untermenüs zu löschen, allerdings wäre das für einen Laien nicht praktizierbar. Andere Nutzerdaten hingegen sind nicht durch den Benutzer löschar und könnten nur durch ein Zurücksetzen auf den Ausgangszustand unauffindbar gemacht werden. Daher erachtet das Projektteam diese Funktionalität als nicht gegeben.

6. Zurücksetzen auf einen definierten Zustand nach einem bestimmten Intervall an Inaktivität

Da bereits das Zurücksetzen auf einen anderen Zustand außer dem Werkszustand nicht vorhanden war, ist diese Möglichkeit leider auch nicht gegeben. Diese Funktion wäre besonders in öffentlichen Verkaufsräumen von Nutzen, wenn ein Gerät sich alle 2 Stunden wieder auf den Ausgangszustand zurücksetzen könnte und somit für Besucher in einem einheitlichen Zustand befindet.

7. Einsetzbar ab einem Endgerät

Leider bedarf es bei MobileIron einer gewissen Anzahl an Geräten, damit man eine gültige Lizenz erhält. Eine Testversion gibt es zwar schon ab einem Client, diese kann jedoch leider nicht in einem produktiven Umfeld eingesetzt werden. Deshalb kann MobileIron diesen Punkt leider nicht erfüllen.

5 MDM + Container

5.1 MobileIron Containertechnologie

MobileIron Content Management (MCM) ist die Containertechnologie der gleichnamigen Firma MobileIron. Die Containertechnologie basiert auf dem MobileIron MDM System, welches in diesem Projekt separat analysiert wird.

Der Hauptnutzen dieser bestimmten Containertechnologie ist die Verwaltung von Benutzerressourcen. Ursprungsansatz der Entwicklung ist der fortschreitende Trend zur Arbeit auf mobilen Geräten wie Tablets. Das MCM System ist auf folgende Module aufgeteilt: Docs@Work, Apps@Work, Web@Work.

5.2 Allgemeine Information der getesteten Software

Name:	MobileIron Mobile@Work
Manufacturer:	MobileIron 415 East Middlefield Road, Mountain View, CA 94043
Current Version:	
Date:	26.01.2015
Price:	-
Website:	https://www.mobileiron.com/en/products/product-overview
Documentation:	https://support.mobileiron.com/eval/

5.3 Installation

Die Konfiguration der Containertechnologie geschieht über ein Webinterface, welches man ebenso für das Standard MDM System von MobileIron verwendet. Dokumentation wie zum Beispiel ein Benutzerhandbuch über die Konfigurationsschritte von MobileIron direkt ist zwar vorhanden, wobei dieses erst größtenteils während unserer Projektzeit erschienen ist. Somit versuchte das Projektteam zunächst dies alleine zu bewältigen und scheiterten, da die gesamte Benutzeroberfläche nur gering selbsterklärend ist. Mittels dem Benutzerhandbuch gelang es uns nach mehreren Versuchen die Containertechnologien Web@Work und Apps@Work funktionstüchtig zu bekommen. Für mehr Aufwand sorgten Zertifikate und Konfigurationsdateien, welche man für die Konfiguration benötigt und nur sehr oberflächig in den bereitgestellten Dokumenten beschrieben worden sind. Weiters beschäftigte uns die nicht konstante und auffallend lange Synchronisationszeit zwischen dem Server und dem Client Device. Der Grund dafür war für uns nicht klar ersichtlich und hätte für eine genaue Analyse zu viel Zeit in Anspruch genommen. Jedoch deutet vieles darauf hin, dass die Ursache in unserem selbstaufgebauten Testnetzwerk entstanden ist, welches sehr klein und minimalistisch gehalten wurde.

5.4 End User Products

5.4.1 Docs@Work

Docs@Work stellt den Usern unternehmensinternen Content für deren tägliche Arbeit zur Verfügung. Dies basiert auf einem Cloud-Content-Management-System und wird am Gerät als eigene App dargestellt. In erster Linie gewährt die Containertechnologie einen sicheren Verbindungsauflauf zu „Content Repositories“. Unter „Content Repositories“ versteht man im Allgemeinen gewöhnliche unternehmensinterne Fileserver (SharePoint), welche als Netzwerkressourcen zur Verfügung stehen. Besonders dabei ist, dass auch empfangende Daten via Email in dem gleichen Ausmaß wie normale „Content Repositories“ den Sicherheitsrichtlinien unterworfen sind. Dem User steht die Möglichkeit des Downloads dieser Dokumente offen. Diese Funktion, welche offiziell unter dem Namen „Secure Email Attachment“ geläufig ist, basiert auf MobileIron AppContent. Diese Technologie stellt eine konsistente, sichere Umgebung auf dem Android Gerät zur Verfügung. D.h. Unternehmensdaten welche auf dem Mitarbeiter-Gerät abrufbar sind, sind verschlüsselt.

Aus Administrator-Sicht gewährt Docs@Work neben der zentralen Verwaltung von Ressourcen für einzelne User ebenfalls die Möglichkeit bei nicht gewünschten Tätigkeiten eines Mitarbeiters seine Berechtigung einzuschränken. Administratoren können bestimmte Geräte von Mitarbeitern in Quarantäne verschieben bzw. diese ganz entfernen. Die betroffenen Geräte sind dann nicht mehr in der Lage sich mit dem Server zu verbinden.

Als gewissen Vorteil dieser Technologie kann man den Wegfall des üblichen zusätzlichen VPN auf den Geräten bezeichnen. Die Bedingung ist für den Anwender kompakter und schneller. Sofern der User Zugriff auf den „Content Repository“ hat, gilt die Regelung des Single-Sign-On. Dies gewährt einer nahezu einmaligen Anmeldung per einem User Account und garantiert Zugriff auf alle verknüpften Anwendungen ohne weitere Log-in Session.

Die oben genannte Secure Email Attachment Funktion ist nach Informationen von der offiziellen MobileIron Homepage bis zum heutigen Stand, nur mittels Email Applikationen namens Divide und Email+ funktionstüchtig.

Ebenso gibt es nur ausgewählte „Content Repositories“, welche Docs@Work unterstützen. Zu diesen zählen:

- Microsoft Sharepoint 2007/2010/2013
- CIFS Windows 2008 R2 SPI
- CIFS Samba CentOS 6.2
- Apache-based WebDAV content repositories
- IIS-based WebDAV content repositories

5.4.2 Web@Work

Web@Work garantiert Unternehmen einen sicheren, mitunter auch beschränkten Internetzugriff ihrer Mitarbeiter. Es basiert auf zwei Technologien namens AppTunnel und MobileIron Sentry, welche bei Nutzung von Web@Work auch konfiguriert werden müssen. Das Zusammenspiel dieser zwei Technologien gewährt eine Zugangsbeschränkung/Kontrolle sowie den verschlüsselten Datenaustausch. Die Administratoren sind in der Lage gewisse, in den Augen des Unternehmens wichtige Websites für die Mitarbeiter frei zu schalten und somit den Besuch dieser zu erlauben. Unter diesen Websites werden auch interne Unternehmens Websites verstanden. Daten wie der Zwischenspeicher des Browsers, Cookies, die Web History als auch Daten von anderen Websites werden alle verschlüsselt übertragen. Sofern ein Android-Gerät eines Mitarbeiters den Zugangsbestimmungen nicht mehr entspricht, werden all diese Daten aus Sicherheitszwecken gelöscht.

Laut der Dokumentation von MobileIron ist es möglich die User-/Geräte-Verwaltung dieses Systems mit dem Enterprise Directory des Unternehmens zu koppeln. Somit ist das Aktivieren und Zulassen von Websites basierend auf bestimmten Gruppen von Usern möglich.

Die Gegenmaßnahme von MobileIron zur Prävention von DLP (Data Loss Prevention) ist die Deaktivierung vom Erstellen von Screenshots des Users.

Beim Einsatz dieser Technologie steht der Vorteil bezüglich des VPN im Vordergrund. Um Mitarbeitern einen sicheren, abgeschirmten Zugriff auf Webressourcen zu gewähren war bisher eine mögliche Variante, eine VPN Verbindung einzurichten. Web@Work ersetzt das VPN und gewährt weitere Möglichkeiten wie bereits oben beschrieben.

5.4.3 Apps@Work

Diese Technologie stellt dem Benutzer des Devices die benötigten Apps zur Verfügung. Dabei kann der User selbst nicht entscheiden, welche Apps er installiert, sondern der IT Administrator. Somit wird das Verwenden des Devices für nicht unternehmensinterne Angelegenheiten verhindert. Der IT Administrator deklariert alle Apps, die laut Unternehmensführung genehmigt sind. Alle anderen Apps welche nicht angeführt sind, sind somit nicht für die Installation zulässig. Diese Technologie basiert auf AppConnect und AppTunnel.

Die erstangeführte Technologie AppConnect ist zuständig für die Implementierung eines Containers um die eigentliche App. Das Resultat daraus ist ein Schutz gegen „data-at-rest“ Daten des Devices bzw. Apps. Die vorhandenen Daten werden verschlüsselt und vor unberechtigten Zugriff geschützt. Auf dem jeweiligen Device sind alle App Container der Apps verbunden und kommunizieren miteinander. Dabei werden Informationen, wie zum Beispiel Richtlinien und Single sign-on Daten, ausgetauscht.

AppTunnel ist für die Sicherheit und den Schutz der data-in-motion Daten zuständig. Diese Technologie stellt sicher, dass die einzelnen Container um die Apps vom restlichen System abgeschirmt sind und keine Verbindung von außen über das Android Basissystem auf die Container stattfinden kann. Die Verbindung wird einzig und alleine zu autorisierten Apps, Usern und Devices aufgebaut. Die „certificate-based session authentication“ verhindert man-in-the-middle Attacken.

Bei der Benutzung von Apps@Work unterscheidet man zwischen 3 verschiedenen Möglichkeiten Apps in das System einzubinden und dem User bereitzustellen.

Die wohl geläufigste Art Apps auf ein Android Device zu implementieren ist das Herunterladen mittels dem Google Play Store.

Der theoretische Ablauf ist folgender: Der IT Administrator fertigt einen Vorschlag für eine ausgewählte App des Google Play Store und hat somit eine Freigabe für den Download dieser App. In unserem Anwendungsfall trifft diese Lösung nicht zu, da der Google Play Store einzig und allein bei Verwendung eines Google Accounts auf dem Device funktioniert. Die Verwendung eines Google Accounts ist in unserem Fall jedoch nicht zielführend, da weitere Sicherheitsprobleme auftreten würden. Daraus ergibt sich die zweite Variante zum Management der auf den Devices installierten Apps
anzuwenden.

Die In-house Apps basieren hauptsächlich auf selbst entwickelten Applikationen, welche zum Beispiel das Unternehmen selbst in Auftrag gegeben hat. Für die Freischaltung zum Download dieser durch die User benötigt man die APK der App, welcher der IT Administrator zunächst selbst hochladen muss. Weiters kann man mittels Benutzung von In-house Apps ebenfalls zum Teil Apps installieren, welche im Google Play Store verfügbar sind. Voraussetzung dafür ist der rechtmäßige Besitz der APK Files der Apps.

5.5 Übereinstimmung mit den Anforderungen des Projektpartners

Die von unserem Projektpartner gestellten Anforderungen, welche bestmöglich erfüllt werden sollten, werden durch diese Containertechnologie zusammen mit dem gleichnamigen MDM System nur sehr geringfügig verbessert. Grund dafür ist in erster Linie, dass diese Containertechnologie - wie bereits oben erwähnt - sich auf die Verwaltung der Inhalte und Dokumente der User konzentriert. Unser Projektpartner stellt zwar ebenfalls 3 Anforderungen in diesem Bereich, diese kann man jedoch mittels dieser Technologie nicht wie gefordert umsetzen. Der Knackpunkt liegt in diesem Bereich bei der Verwaltung über Remote. Wir als Projektteam haben uns darauf geeinigt, dass wir unter diesem Punkt eine komplette Remotesteuerung verstehen. Ein entsprechendes Beispiel wäre das Auswählen eines Dokumentes zentral und ein sofortiges Öffnen dieses Dokumentes am Clientdevice. Für dieses Anliegen stellen die Produkte der Produktlinie Mobile@Work keine Funktion zur Verfügung und können somit nicht komplett als erfüllend angenommen werden.

Die meisten Anforderungen sind bereits durch das MDM System alleine abgedeckt worden. Wenige zusätzliche können einerseits durch Web@Work und andere durch Apps@Work realisiert werden.

5.5.1 Zusätzlich erfüllte Anforderungen

1) User Daten (zB. Browser History, Spielstände, Bookmarks) können gelöscht werden

Web@Work lässt dem Administrator die Möglichkeit offen, bisherige Daten, wie zum Beispiel die History und die Bookmarks, zu löschen. In der Regel passiert dies sobald die Zugangserfordernisse nicht erfüllt sind. Benutzer des Devices selbst können das Tablet jedoch nicht auf den ursprünglichen Standard zurücksetzen. Somit ist die Anforderung für den Gebrauch nicht ganz erfüllt.

2) Ressourcen (Dokumente, Videos, Bilder, etc) kann man per Remote verwalten/verändern

Docs@Work gewährt einen Verbindungsauflauf mit einem Content Repository eines Unternehmens. Mit Hilfe dieser Verknüpfung können Dokumente einzelnen Usern oder auch Gruppen zur Verfügung gestellt werden. Jedoch ist wie bereits oben genauer beschrieben ein kompletter Remote Zugriff nicht möglich.

4) Apps können per Remote verwaltet werden

Administratoren können Apps zur Installation freischalten und somit deren Installation erlauben.

6 Samsung Knox

Dieser Abschnitt beschäftigt sich mit dem Einsatz von Samsung Knox als System für potentielle Kunden von der Firma Kapsch. Hier findet sich sowohl Informationen zu Samsung Knox im Allgemeinen, als auch darüber, welche der von der Firma Kapsch erwünschten Features mit Samsung Knox realisierbar sind und welche nicht. Außerdem wird noch auf die Bedienbarkeit und die Installation eingegangen.

6.1 Allgemeine Infos zu Samsung Knox

Name:	Samsung Knox
Manufacturer:	Samsung
Current Version:	2.1
Publishing Date:	08.07.2014
Price:	<ul style="list-style-type: none"> • Express (E): free but limited to 250 seats • Premium (P) : USD \$1 MSRP per device/month • Workspace (W) : USD \$3.60 MSRP per device/month
Website:	https://www.samsungknox.com/de
Documentation:	

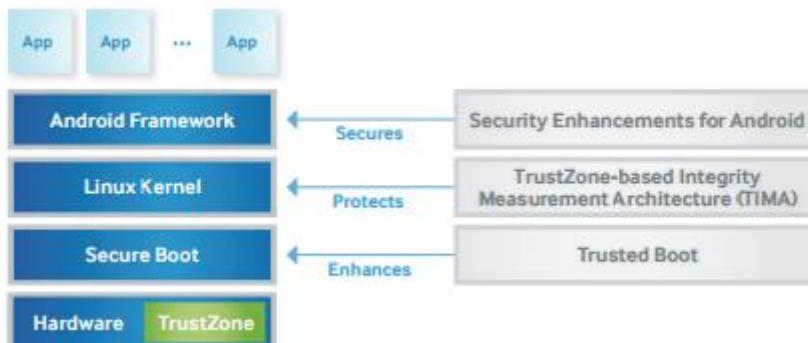
Samsung Knox ist eine Sammlung Business-orientierter Sicherheitsfunktionen für Android-Geräte von Samsung. Das System, welches auf SE for Android basiert, ist speziell auf die Bedürfnisse von Unternehmen in Hinsicht auf die Sicherheit der Endgeräte ausgerichtet.

6.2 Samsung Knox Bestanteile

Um wirklich erklären zu können was Samsung Knox ist, muss man es zuerst einmal in seine 3 Hauptbestandteile unterteilen.

a. Plattform Security

In jedem Samsung Gerät ist ein physischer Hardware Chip eingebaut, welcher automatisch einen höheren Schutz für Samsung Geräte bieten soll. Mittels diesem Chip ist es Samsung möglich, bis in die tiefste Software-Schicht eines auf Android basierenden Geräts einzugreifen.



In der Grafik links sieht man den groben Aufbau eines Android Systems. In der Grafik rechts sieht man die Technologien, mit deren Hilfe es Samsung möglich ist in die Schichten einzugreifen.

a.1. Kerntechnologien der Plattform Security

a.1.1. SE for Android

a.1.1.1. So funktioniert es

SE for Android basiert auf SELinux-Technologie und definiert die Zugriffskontrolle auf Linux-Ebene. Mandatory Access Control (MAC) und Discretionary Access Control (DAC) überwachen und verwalten welche Dateien und Apps auf das System des Geräts zugreifen können.

a.1.1.2. Sicherheit

SE for Android erzwingt MAC, wobei Apps nur genau die Rechte zugewiesen erhalten, die sie für den Zugriff auf das System des Geräts benötigen. Sollte ein böswilliger Benutzer oder eine bösartige App also Zugriff auf das Gerät erhalten, dann betrifft der Schaden immer nur einen bestimmten Bereich, während die restlichen Bereiche des Geräts geschützt bleiben.

Wenn SE for Android ausgelöst wird, sendet es eine Präventionsinformationsmeldung an den Benutzer des Geräts, die in der **Informationsleiste** erscheint. Die Präventionsinformationsmeldung gibt an, welche Anwendung versucht hat, auf Daten des Geräts zuzugreifen, und sie wird die Deinstallation der betreffenden Anwendung empfehlen. Beachten Sie dabei jedoch, dass SE for Android auch von autorisierten Anwendungen ausgelöst werden kann. Dies kann vorkommen, wenn die App einen von dem in der SE for Android-Richtlinie vorgegebenen Pfad verwendet. Wenn dies der Fall ist, sollten Sie die Datei der Sicherheitsrichtlinie entsprechend aktualisieren.

a.1.2. ARM TrustZone-basierte Integrity Measurement Architecture (TIMA)

a.1.2.1. So funktioniert es

TIMA schützt Ihr Gerät auf zwei verschiedene Weisen. Erstens prüft sie in regelmäßigen Abständen, ob der Kernel des Geräts geändert wurde, indem sie den gegenwärtigen Zustand mit dem Original-Kernel vergleicht. Zweitens authentifiziert TIMA Kernel-Module, wenn diese geladen werden, sodass Geräte nie ungeschützt sind.

a.1.2.2. Sicherheit

Die TIMA TrustZone ist ein manipulationssicherer Sektor des ARM-Prozessors in Ihrem Gerät. Sie authentifiziert und verifiziert den Linux-Kernel über regelmäßige Messungen. Wenn TIMA for Android ausgelöst wird, sendet es eine Ermittlungsinformationsmeldung an den Benutzer, die in der Informationsleiste erscheint. In der Meldung werden Sie normalerweise zum Neustart des Geräts aufgefordert.

a.1.3. Secure Boot und Trusted Boot

a.1.3.1. So funktioniert es

Secure Boot verhindert, dass unbefugte Bootloader und Kernels in das Gerät geladen werden. Dies bedeutet, dass Ihr Gerät nicht manipuliert wurde und der KNOX-Container geladen werden kann.

Trusted Boot vergleicht den Bootloader und den Kernel des Betriebssystems mit den originalen Werksversionen. Dies wird dadurch erreicht, dass die Originaldaten des Geräts aufgezeichnet werden und das Gerät permanent beim Systemstart mit diesen gegengeprüft wird, um sicherzustellen, dass sich diese Daten nicht geändert haben.

a.1.3.2. Sicherheit

Es gibt 4 Bootloader auf dem Gerät. Jeder Bootloader prüft die Gültigkeit des vorhergehenden Bootloaders oder Kernels. Trusted Boot sichert diese Bootloader. Die Funktion ist in der ARM TrustZone, einem manipulationssicheren Sektor des ARM-Prozessors, eingebettet. Trusted Boot verwendet kryptografische Schlüssel, um sicherzustellen, dass die Messungen am Gerät dem Original entsprechen. Diese Schlüssel werden erst von Trusted Boot freigegeben, wenn SE for Android bestätigt, dass genehmigte Firmware auf dem Gerät ausgeführt wird.

b. Applikation Security

Der Bereich Application Security bietet folgende drei Sicherheitsfunktionen:

b.1. Application Containers

Container sind quasi ein Android im Android und bezeichnet einen gesicherten, separaten Bereich (Container) auf dem Gerät. Dieser Bereich hat einen eigenen Homescreen, eigene Anwendungen und eigene Daten. Diese Funktion ist damit vergleichbar mit einer Art Dual-Boot-Variante. Anwendungen außerhalb des Containers haben dabei keinerlei Zugriff auf die Daten oder Prozesse innerhalb des Containers. Anwendungen innerhalb des Containers haben grundsätzlich keinen Zugriff auf Daten außerhalb des Containers.

Mittels Richtlinienkonfiguration kann der IT-Verwalter des Gerätes einen read-only (nur lesen) Zugriff für bestimmte Anwendungen im Container auf Daten außerhalb des Containers einrichten. Umgekehrt besteht diese Möglichkeit allerdings nicht. Daten innerhalb des Containers werden dabei mit einem Verschlüsselungsalgorithmus und einem AES-256 Bit Schlüssel geschützt. Ein Zugriff ist erst nach Eingabe eines Passwortes möglich.

b.2. On-device Data Encryption (kurz ODE)

ODE ist ein im Knox enthaltenes Feature zur Verschlüsselung von Daten. Dabei kann sowohl der sichere Container als auch der normale Bereich, sowie interner als auch externer Speicher verschlüsselt werden. Verwendet wird ein AES Algorithmus mit einem 256 Bit starken Schlüssel. Die Verschlüsselungsfunktion kann vom User selbst unter den Einstellungen oder vom IT Administrator durch eine Richtlinie aktiviert werden.

b.3. Virtual Private Network Support

VPN-Verbindungen verwendet man um sicherzustellen, dass Daten bei der Übertragung geschützt sind und um den Netzwerkverkehr nicht mit den Daten von persönlichen Apps zu belasten.

b.3.1. So funktioniert es

KNOX Workspace bietet 3 VPN-Optionen für den Schutz Ihrer Daten bei der Übertragung. Ein geräteweites VPN kann von Benutzern konfiguriert werden, sofern sie den entsprechenden Servernamen und die dazugehörigen Informationen zur Hand haben. VPN pro App oder ein containerweites VPN kann von IT-Administratoren über die MDM-Konsole eingerichtet werden. Auf der MDM-Konsole kann man bis zu 5 verschiedene VPN-Profile einrichten und diese einzelnen Apps zuweisen, um VPN pro App zu implementieren.

b.3.2. Sicherheit

VPN-Verbindungen sind die sicherste Methode, um die Daten bei der Übertragung zu schützen. Der KNOX VPN-Client kann ein bestehendes VPN-Gateway für den Schutz von Daten verwenden. KNOX VPN ist über Ihre MDM-Konsole im FIPS-Modus konfigurierbar. Zu den Sicherheitsfunktionen gehören NSA-Suite-B-Algorithmen, Unterstützung für X.509 mit Zertifikatsprüfung auf OCSP-Basis und 256-Bit-AES-Verschlüsselung. Wenn Ihr Unternehmen SmartCards verwendet, können diese mit VPN-Anmeldedaten konfiguriert werden.

c. Management (MDM)

Mit MDM (Mobile Device Management) bietet Samsung Knox der IT Abteilung des Unternehmens die Möglichkeit eine integrierte Lösung zur Verwaltung und Administration von Samsung-Geräten vorzunehmen, ohne dabei auf Drittanbieter zurückgreifen zu müssen. Der Nachteil dabei ist die ausschließliche Verwendung mit Samsung-Geräten, welche über Samsung Knox verfügen.

Samsung Knox gibt es in 2 Varianten, wobei beide Varianten ein MDM als Basis benötigen um zu funktionieren.

c.1. Knox Lösungen

c.1.1. Knox Express

Diese Lösung ist für kleine bis mittelständische Unternehmen gedacht. Sie ist gratis, jedoch limitiert auf 250 Geräte.

Falls die 250 Plätze nicht mehr reichen, kann man Knox Express problemlos auf Knox Premium updaten.

Das ist die Variante, die die Projektgruppe empfehlen würde, da sie gratis ist und da laut Angaben von der Firma Kapsch es eher unwahrscheinlich ist, dass ein Kunde mehr als 250 Geräte hat und falls doch ist es ja problemlos auf Premium erweiterbar.

c.1.2. Knox Premium

Diese Komplettlösung eignet sich ideal für Unternehmen, in denen Sicherheit oberste Priorität hat.

Knox Premium kostet pro Gerät pro Monat 1 \$ und kann um zusätzliche Add-ons erweitert werden.

Weiters bietet Knox Premium einige zusätzliche Features für Android und eine bessere IOS Integration. In Hinblick auf die von Kapsch erwünschten Funktionalitäten bringen diese zusätzlichen Features jedoch keine Vorteile.

c.1.2.1. Add-ons

c.1.2.1.1. Knox Workspace

Knox Workspace ist eine Erweiterung im Bereich Container.

Dieses Add-on bietet eine einfache Konfiguration, zwei Container pro Gerät, zusätzliche Apps und verbesserte Sicherheitsfunktionen.

Die Hauptfeatures, die Workspace mit sich bringt, sind:

- Erweiterte Containerverwaltung mit sicheren Richtlinien
- Datenverschlüsselung bei jedem Entsperren des Containers
- Pro-App-VPN für sichere und schnelle Verbindung
- Support für zwei separate Container für maximale Produktivität und für Trennung von arbeitsbezogenen und privaten Daten

All diese Einstellungen, die mit Knox Workspace dazu kommen, gelten nur innerhalb des Containers. Da es jedoch keine Einstellung gibt um zu verhindern, dass der Benutzer den Container verlässt und somit alle Workspace-Einstellungen umgeht, ist dieses Add-on in unserem Fall ungeeignet.

Jedoch soll in Zukunft eine Art Container-only-mode implementiert werden. Sobald dieser vorhanden ist, wäre Knox Worspace auf jeden Fall ein Addon das in Betracht gezogen werden sollte

c.1.2.1.2. Knox IAM

Knox IAM ist eine SSO-Lösung(Single Sign-On-Lösung).

Das heißt mit dem IAM Add-on hat man nur mehr einen einzigen Account, mit dem man sich in allen Apps anmelden kann.

c.2. MDMs

Knox bietet zur Verwaltung der Geräte ein eigenes MDM, das EMM, hat aber auch genügend Partner, deren MDMs Samsung Knox Funktionen implementiert haben.

Samsung Knox funktioniert also mit jedem MDM aus der [nachfolgenden Liste](#).

c.2.1. Liste an Kompatiblen MDMs

- [Samsung EMM](#)
- [MobileIron](#)
- [Absolute Software](#)
- [AirWatch](#)
- [CA Technologies](#)
- [Centrify](#)
- [Citrix](#)
- [FancyFon](#)
- [MaaS360](#)
- [NQ Mobile](#)
- [Samsung SDS](#)
- [SAP](#)
- [SOTI](#)

c.2.2. Samsung EMM

Samsung KNOX EMM ist eine cloudbasierte Verwaltungslösung für Unternehmen. IT-Administratoren können damit Benutzer, Apps und plattformübergreifende Geräte über eine cloudbasierte Konsole verwalten. Außerdem bietet KNOX EMM Single Sign-On (SSO) und eine starke Authentifizierung für eine benutzerfreundliche und sichere Arbeitsumgebung für Mitarbeiter.

Es ist einfach zu installieren und sehr übersichtlich gestaltet. Falls in der Firma noch keine andere MDM Lösung installiert sein sollte, ist EMM zu empfehlen.

6.3 Unterstützte Geräte

Ein riesiger Nachteil an Samsung Knox ist, dass ungefähr 90% aller Funktionen bei folgenden Geräten einstellbar sind:

- **ab KitKat 4.4.4**
 - Galaxy Note 4
- **ab KitKat 4.4.2**
 - Galaxy S5
 - Galaxy Avant
- **ab Jelly Bean 4.3**
 - Galaxy S4
 - Galaxy S3
 - Galaxy Note 3
 - Galaxy Note 2
 - Galaxy Express 2
 - Galaxy Grand
 - Galaxy NotePro 12.2
 - Galaxy Note 10.1
 - Galaxy Note 8.0
 - Galaxy Tab S
 - Galaxy TabPro
 - Galaxy Tab 4
 - Galaxy Tab 3
- **ab Jelly Bean 4.2.2**
 - Galaxy Mega
- **In jeder Version**
 - Galaxy Alpha
 - Galaxy Note Edge
 - Galaxy Ace
 - Galaxy Core

6.4 Erfüllte Funktionen

Dieser Abschnitt beschäftigt sich damit, welche für unser Projekt wichtigen Funktionen Samsung Knox mit sich bringt.

6.4.1 Kernfunktionen

Unter Kernfunktionen fallen die Funktionen die benötigt werden um den Anforderungen von der Firma Kapsch gerecht werden zu können.

6.4.1.1 Allgemeine Funktionen

- Nutzbar ohne Public Cloud Services
 - Samsung Knox ist so konfigurierbar, dass man keine Public Cloud Dienste verwenden muss.
- Keine speziellen Person zugewiesen
 - Samsung Knox ist so konfigurierbar, dass das Gerät anonym bleibt, also keiner speziellen Person zugewiesen ist.
- Der Sperrbildschirm der Geräte kann fernverwaltet werden
 - Hintergrund des Sperrbildschirms kann geändert werden
 - Passwort für den Sperrbildschirm kann gesetzt werden

6.4.1.2 Inhaltseinstellungen (Apps, Webseiten, Marketing)

- Installieren von Apps kann verboten werden
- Dem Benutzer kann das Recht zur Verstellung der Systemeinstellungen genommen werden.
- Browserinhalt und bestimmte URLs können gefiltert werden
 - Für die Geräte ist es möglich einen Proxy-Server einzustellen. Durch das Konfigurieren dieses Servers kann man dann den Inhalt, den man filtern will, filtern.
- Remote Zugriff ist möglich
 - Das heißt man kann über das Internet mit Hilfe der Admin Konsole auf die Geräte zugreifen

6.4.1.3 Statistiken

- Es ist möglich sogenannte Berichte generieren zu lassen
 - Der einzige Hacken daran ist, dass man sie dafür erst selbst mittels SQL „programmieren“ muss.
 - Genaue Informationen über die Funktionsweise findet man unter <https://emm3.samsungknox.com/vfslow/lib/docs/samsung/adminref/wwhelp/wwhimpl/js/html/wwhelp.htm#href=cloud-admin-reports.html>
 - Informationen zum Syntax der SQL-Queries gibt es unter <https://emm3.samsungknox.com/vfslow/lib/docs/samsung/adminref/wwhelp/wwhimpl/js/html/wwhelp.htm#href=cloud-admin-sql-func-exa>.

6.4.1.4 Benutzer und Geräte

- Geräte können zum Auslieferungszustand zurückgesetzt werden

6.4.1.5 Managementseitige Anforderungen

- Geräteverwaltung ist von mehreren Standorten möglich
- Geräte inklusive Status können angezeigt werden
- Geräte können per remote auf den Auslieferungszustand zurückgesetzt werden
- VPN Einstellungen können mittels remote geändert werden
- Passwort Richtlinien können definiert werden
- Die Statistiken sind per remote auslesbar

6.4.1.6 Zusätzliche Anforderungen

- Das Gerät ist benutzbar ohne Kiosk-Modus
 - Im Kiosk-Modus kann man nur genau eine einzige App zu verwenden.
- Die Lösung ist nicht auf nur eine einzige Hardware beschränkt
 - Dies funktioniert jedoch nur mit den zuvor genannten unterstützten Geräten.
- Bestimmter Inhalt kann gefiltert werden
 - Es ist möglich für die Geräte einen Proxy-Server einzustellen. Durch das Konfigurieren dieses Servers, kann man dann den gewünschten Inhalt filtern.
- Mehrsprachen Unterstützung
 - Samsung Knox unterstützt folgende Sprachen:
 - Englisch
 - Spanisch
 - Französisch
 - Deutsch
 - Italienisch
 - Portugiesisch
 - Brasilianisches Portugiesisch
 - Einfaches Chinesisch
 - Traditionelles Chinesisch
 - Koreanisch
 - Japanisch

6.4.2 Zusätzliche Funktionen

Darunter fallen Funktionen, die zwar nicht explizit von der Firma Kapsch erwünscht wurden, jedoch eventuell für die Firma Kapsch interessant seinen könnten.

- Verschlüsselung
 - Geräte Verschlüsselung
 - Interne Speicher Verschlüsselung
 - SD Karten Verschlüsselung
- Management von Geräten
 - Mittels remote kann man
 - das Gerät zum Auslieferungszustand zurücksetzen
 - das Passwort des Geräts ändern
 - das Gerät lokalisieren
 - feststellen, ob die SIM-Karte im Gerät getauscht wurde
 - maximale Anzahl an falsch eingegebenen Passwörtern setzen
 - Firewall Einstellungen ändern
 - Whitelists und Blacklists einrichten für
 - Bluetooth
 - Wlan
 - Protokollierung von Anrufinformationen aktivieren
 - Protokollierung von Netzwerkstatistiken für Mobilfunkdaten aktivieren
 - Protokollierung von WLAN-Netzwerkstatistiken aktivieren
- Funktionen des Geräts für den Benutzer erlauben/verbieten
 - App Benachrichtigungen
 - Kamera
 - Mikrofone
 - USB-tethering
 - Bildschirm Übertragungen

- Screenshots
- USB-Debugging
- S-Voice
- Aufnahmen
 - Video
 - Audio
- Datum und Zeit Änderungen
- Roaming
- SMS
 - Senden
 - Empfangen
- Bluetooth-Einstellungen ändern
- Wlan-Einstellungen ändern

6.5 Fehlende Funktionen

Folgende von der Firma Kapsch erwünschte Funktionen sind mit Samsung Knox nicht realisierbar:

6.5.1 Inhaltseinstellungen(Apps,Webseiten, Marketing)

- Deinstallieren von Apps kann verboten werden
- Standard Apps können ausgeblendet werden
 - Bei Samsung Knox Premium wäre das möglich, da man dort einstellen kann, welche Apps vom Benutzer geöffnet werden können.
- App Content kann fernverwaltet werden
- Per remote können bestimmte Webseiten am Gerät geöffnet werden
- Der Inhalt der Geräte (Bilder, Videos, etc.) kann fernverwaltet werden.
- Der Sperrbildschirm der Geräte kann fernverwaltet werden
 - Hintergrund des Sperrbildschirms kann geändert werden
 - Passwort für den Sperrbildschirm kann gesetzt werden
- Der Hintergrund der Geräte kann fernverwaltet werden

6.5.2 Benutzer und Geräte

- Es ist möglich das Gerät auf einen beliebigen zuvor definierten Zustand zurückzusetzen.
- Es ist möglich gewisse Benutzerdaten(History, Highscores, Lesezeichen, etc.) zu löschen
- Das Gerät setzt sich nach einer gewissen Zeit automatisch auf einen bestimmten Zustand zurück.

6.5.3 Managementseitige Anforderungen

- Fehler Berichte sind per Gerät und per Standort auslesbar.
 - Per Gerät
 - Per Standort

7 Endergebnis und Empfehlung

Nach Abschluss des Evaluierungsprozesses ist das Projektteam an diesem Punkt in der Lage eine Empfehlung an das Unternehmen Kapsch auszusprechen. In diesem Teil wird nun erläutert, welches evaluierte Konzept mit den Anforderungen des Auftraggebers übereinstimmt und wodurch dies zu Stande gekommen ist. Nachdem die Linuxmanipulation aus Garantietechnischen Gründen bereits als ausgeschieden gilt, befasst sich der folgende Abschnitt nur mehr mit den Konzepten MDM, MDM + Container und Samsung Knox.

7.1 Evaluierung

Um feststellen zu können welche der 3 Lösungen am besten die Anforderungen des Auftraggebers erfüllen kann, hat das Projektteam eine Nutzwertanalyse erstellt, mit derer Hilfe das Projektteam die Funktionen der Systeme direkt miteinander vergleichen konnten.

		max. Punkte	Gewicht	MDM	Bewertung	MDM+ Container	Bewertung	Samsung Knox	Bewertung
Allgemein		8							
	Nutzbar ohne Public-Cloud-Services	3	10	30	10	30	10	30	
	Keiner speziellen Person zugewiesen	5	10	50	10	50	10	50	
Nutzwert Allgemein		80		80		80		80	
Inhaltseinstellung		44							
	Installieren von Apps kann verboten werden	6	7	42	7	42	10	60	
	Deinstallieren von Apps kann verboten werden	6	3	18	3	18	0	0	
	Kein benutzerseitiger Zugang zu Systemeinstellungen	8	10	80	10	80	10	80	
	Standard Apps können ausgeblendet werden	2	2	4	2	4	2	4	
	Browser Inhalt kann gefiltert werden	4	10	40	10	40	10	40	
	Remote-Zugriff ist möglich	8	10	80	10	80	10	80	
	Apps können per Remote verwaltet werden	5	6	30	8	40	10	50	
	Websiten können per Remote verwaltet werden	2	0	0	0	0	0	0	
	Inhalt (Dokumente, Videos, Bilder) können per Remote verwalten können	1	0	0	8	8	0	0	
	Hintergrund und Speerbildschirm können per Remote verwalten können	2	5	10	5	10	5	10	
Nutzwert Inhaltseinstellungen		440		304		322		324	
Statistik		7							
	Up/Down-Time einsehbar	3	10	30	10	30	10	30	
	Stastiken sind pro Device und Standort auslesbar	3	10	30	10	30	10	30	
	Statistiken über Aktivitätszeit, App aufrufe und Websiten	1	4	4	4	4	10	10	
Nutzwert Statistik		70		64		64		70	
Benutzer und Geräte		10							
	Geräte können zu einem spezifischen Status zurückgesetzt werden	5	3	15	3	15	3	15	
	Benutzerdaten (history, Spielstände,Lesezeichen) können gelöscht werden	3	3	6	4	12	3	9	
	Automatischen zurücksetzen der Geräte nach einer Zeiteinheit	2	0	0	0	0	0	0	
Nutzwert Benutzer und Geräte		100		24		27		24	

Managementseitige Anforderungen		20						
	Geräteverwaltung von mehreren Standorten möglich	3	10	30	10	30	10	30
	Geräte (inkl. deren Status) können angezeigt werden	4	10	40	10	40	10	40
	Fehlerberichte sind pro Gerät und pro Standort verfügbar	4	7	28	7	28	9	36
	Geräte können zu einem spezifischen Status per Remote zurückgesetzt werden	5	3	15	3	15	3	15
	Statistiken können per Remote ausgelesen werden	3	10	30	10	30	10	30
	Inhalt (Dokumente, Videos, Bilder) können per Remote geändert können	1	0	0	8	8	0	0
Nutzwert Managementseitige Anforderungen		200		143		151		151
Zusätzliche Anforderungen		11						
	Gerät ohne Kiosk-Mode verwendbar	5	10	50	10	50	10	50
	Hardwareunabhängig	1	10	10	10	10	10	10
	Geringerer Wartungsaufwand	2	10	20	10	20	10	20
	mehrsprachen Unterstützung	2	10	20	10	20	10	20
	Einsetzbar ab dem ersten Gerät	1	0	0	0	0	10	10
Nutzwert Zusätzliche Anforderungen		110		100		100		110

Nutzwert Gesamt	100		715		744		759
Rang Nutzwert			3		2		1
Kosten	10		6		4		10
Rang Kosten			2		3		1
Kosten/Leistungsverhältnis - Kosten je Pkt.			118,67		186,00		75,90
Rang Kosten/Leistungsverhältnis			2		3		1

7.2 Auswertung der Evaluierung

Um eine aussagekräftiges Ergebnis zu erhalten, wurden die Anforderungen des Gesamtsystems in verschiedene Unterpunkte unterteilt, welche auf ihre Funktionstüchtigkeit hin untersucht wurden.

Die Lösungen wurden dann je nach Erfüllungsgrad der einzelnen Punkte bewertet, was eine objektive Vergleichbarkeit schaffen sollte.

7.2.1 Allgemein

Da alle 3 Systeme sowohl ohne pulic-cloud-service nutzbar waren als auch keiner speziellen Person zugewiesen werden müssen, haben die alle 3 Lösungen volle Punkte erreicht und sind somit was diesen Punkt angeht gleich auf.

7.2.2 Inhaltseinstellungen

In diesem Bereich hat sich herausgestellt ist Samsung Knox kapp aber doch noch vor der MDM+Container Lösung das am besten geeignete System. Denn auch wenn es mit MDM+Containern möglich ist, die Deinstallation von betriebsinternen Apps zu verbieten und man auch Inhalte der Geräte wie zum Beispiel Bilder, Dokumente, etc per remote verwalten kann, ist Samsung die hier die bessere Alternative, da es eine bei weitem bessere Verwaltung von Apps per Remote bietet.

Eine Anforderung die jedoch keines der drei Systeme erfüllen konnte war das Verwalten von Websites per remote.

7.2.3 Statistik

Auch den Unterpunkt Statistiken kann Samsung Knox wieder für sich entscheiden. Diesmal jedoch mit einem Größeren Vorsprung. Zurückzuführen ist das darauf, dass Samsung Knox die Möglichkeit bietet, sich jede beliebige Statistik die man für sein Unternehmen haben will, einfach mittels SQL Code selbst zu erzeugen(siehe Punkt 8.3.1 Statistiken).

7.2.4 Benutzer und Geräte

In diesem Bereich kann die MDM+Container Lösung punkten. Denn im Gegensatz zu den anderen zwei Systemen, kann man mit dieser Lösung am Gerät gespeicherte Lesezeichen löschen.

Auch bei diesem Unterpunkt gibt es wieder eine Anforderung die mit keinem unserer getesteten Systeme umsetzbar war. Und zwar war es nicht möglich die Geräte so zu konfigurieren, dass sie sich nach einer gewissen Zeit von alleine auf einen definierten Stand zurücksetzen.

7.2.5 Managementseitige Anforderungen

Bei der Erfüllung der Managementseitigen Anforderungen sind Samsung Knox und die MDM+Container-Lösung gleich auf. Denn den Vorsprung den Knox gewinnt durch besseres Handling der Fehlerberichte, kann die MDM+Container-Lösung dadurch weg machen, das sie im Gegensatz zu Samsung Knox eine Möglichkeit bietet, Geräteinhalt per remote zu ändern.

7.2.6 Zusätzliche Anforderungen

Da Samsung Knox das einzige System ist, welches bereits ab dem ersten Gerät einsetzbar ist und sonst alle Punkte von allen Systemen erfüllt werden ist auch hier Knox der klare Sieger.

7.3 Empfehlung

Nach eingehender Analyse ist das Projektteam zu dem Schluss gekommen, dass für die geplanten Projekte der Firma die Betriebsplattform Samsung Knox am ehesten geeignet ist. Es implementiert die meisten der benötigten Features, aber lässt dennoch einige fundamentale Punkte aus. Deshalb ist es hier für das Projektteam auch nicht möglich eine hundert prozentige Empfehlung zu geben. Den Informationen der Dokumentation von Samsung Knox zur Folge werden einige benötigte Features in kommenden Versionen eingebaut. Allerdings ist nicht absehbar ob und wann diese erscheinen. Für die beiden anderen Systeme kann deshalb keine Empfehlung ausgesprochen werden, weil sie weniger und besonders im Einsatz mit Nicht-Samsung-Geräten signifikant weniger Funktionen bieten.



ACCEPTANCE TESTING PROTOCOL

KTI – KAPSCH TABLET INFRASTRUCTURE: Spengergasse V – 5CHIF

Overview

The Acceptance Testing Protocol (ATP) defines the procedure of the project's acceptance by the project owner.

Team

Sebastian Götze

Samuel Hammer

Michael Kaufmann

Konstanze Müller

Philip Steinhäuser (Project Manager)

Author

Philip Steinhäuser

ste14288@spengergasse.at

1. Test Cases

Case ID:	1
Name:	Introduction

Case ID:	2
Name:	Method One – Linux Manipulation
Description:	Contains general information, fulfilled features and requirements which are provided and fulfilled by the solution.

Case ID:	4
Name:	Method Two – Mobile Device Management
Description:	Contains general information, fulfilled features and requirements which are provided and fulfilled by the solution.

Case ID:	8
Name:	Method Three – MDM + Container
Description:	Contains general information, fulfilled features and requirements which are provided and fulfilled by the solution.

Case ID:	12
Name:	Method Four - Samsung Knox
Description:	Contains general information, fulfilled features and requirements which are provided and fulfilled by the solution.

Case ID:	16
Name:	Fulfilled Features
Description:	A list of Features which were fulfilled by the solutions.

Case ID:	17
Name:	Conclusion
Description:	The conclusion contains a recommendation of the system which was considered best by the result of our project.

Case ID:	18
Name:	Usable without public cloud service
Description:	No connection to a cloud service (google, Dropbox,etc.) is needed.

Case ID:	19
Name:	Not assigned to a specific person
Description:	The user does not need a google account for using the system.

Case ID:	20
Name:	User cannot install apps
Description:	The user must not install applications.

Case ID:	21
Name:	User cannot access the settings
Description:	The user must not access the settings of his device.

Case ID:	22
Name:	Devices (incl. status) can be located over a dashboard
Description:	Administrators can see the status and the location of the registered devices.

Case ID:	23
Name:	Remote accessibility is given
Description:	Administrators can access the device over the internet.

Case ID:	24
Name:	Device can be reset to a specific state
Description:	It is possible to change the state of the tablet to a specific state which was saved before.

Case ID:	25
Name:	The device is usable without a kiosk mode (single app mode)
Description:	The device is not limited to one single application.

Case ID:	26
Name:	Multilanguage support
Description:	The system can be used in different languages.

2. TESTING AGENDA – PROJECT OWNER

2.1. Research Paper

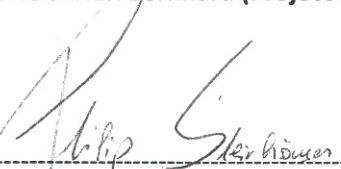
CASE ID	CASE NAME	ACCEPTED
1	Introduction	✓
2	Methode One - Linux Manipulation	✓
3	• General Information	✓
4	Method Two - Mobile Device Management (MDM)	✓
5	• General Information	✓
6	• Features	✓
7	• Requirements	✓
8	Method Three - MDM + Container	✓
9	• General Information	✓
10	• Features	✓
11	• Requirements	✓
12	Method Four – Samsung Knox	✓
13	• General Information	✓
14	• Features	✓
15	• Requirements	✓
16	Fulfilled Features	✓
17	Conclusion	✓

2.2. Prototype

CASE ID	CASE NAME	ACCEPTED
18	Usable without public cloud service	✓
19	Not assigned to a specific person	✓
20	User cannot install apps	✓
21	User cannot access the settings	✓
22	Devices (incl. status) can be located over a dashboard	✓
23	Remote accessibility is given	✓
24	Device can be reset to a specific state	✓
25	The device is usable without a kiosk mode (single app mode)	✓
26	Multilanguage support	✓


14.04.15, Wien

BRUCKNER Bernhard (Project Owner), Vienna, on


14.05.2015

STEINHÄUSER Philip (Project Manager), Vienna, on

3. TESTING AGENDA – PROJECT Coach

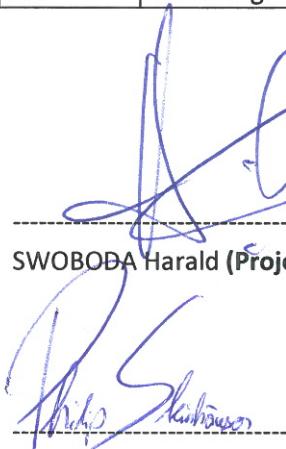
3.1. Research Paper

CASE ID	CASE NAME	ACCEPTED
1	Introduktion	✓
2	Methode One - Linux Manipulation	✓
3	• General Information	✓
4	Method Two - Mobile Device Management (MDM)	✓
5	• General Information	✓
6	• Features	✓
7	• Requirements	✓
8	Method Three - MDM + Container	✓
9	• General Information	✓
10	• Features	✗
11	• Requirements	✗
12	Method Four – Samsung Knox	✓
13	• General Information	✓
14	• Features	✓
15	• Requirements	✓
16	Conclusion	✓
17	Fulfilled Features	✓

3.2. Prototype

CASE ID	CASE NAME	ACCEPTED
18	Usable without public cloud service	✓
19	Not assigned to a specific person	✓
20	User cannot install apps	✓
21	User cannot access the settings	✓
22	Devices (incl. status) can be located over a dashboard	✓
23	Remote accessibility is given	✓
24	Device can be reset to a specific state	✓
25	The device is usable without a kiosk mode (single app mode)	✓
26	Multilanguage support	✓

SWOBODA Harald (Project Coach), Vienna, on



20.04.2015

STEINHÄUSER Philip (Project Manager), Vienna, on