

Pflichtenheft

- SecureMessage -



*„Entwicklung einer Software für die verschlüsselte Kommunikation
innerhalb des Unternehmensnetzwerkes“*

Stand: 16.01.2018

Auftraggeber: MoneyGroup AG
Hauptstraße 28
D-20095 Hamburg
Ansprechpartner: Ariane Freithaler, Tel. 040 / 62306-165

Auftragnehmer: Allsafe - Cybersecurity
Ringstraße 25
D-10365 Berlin
Ansprechpartner: Florian Jeßner

Inhaltsverzeichnis

1 Zielbestimmung	2
1.1 Muss-Kriterien	2
1.2 Kann-Kriterien	2
1.3 Abgrenzungskriterien	2
2 Produkteinsatz	2
2.1 Anwendungsbereich	2
2.2 Zielgruppen	3
2.3 Produktumgebung	3
2.3.1 Architektur	3
2.3.2 Technologie	3
2.3.3 Komponenten	3
2.3.4 Schnittstellen	3
2.4 Betriebsbedingungen	3
3 Produktfunktionen/Anforderungen	3
3.1 Funktionale Anforderungen	3
3.1.1 Beschreibung der FAs mit Rollen innerhalb der Geschäftsprozesse	3
3.1.2 Aktivitäten mit Benutzerschnittstelle (UI)	4
3.1.3 Fachliches Klassendiagramm („domain model“) / Produktdaten	4
3.2 Nichtfunktionale Anforderungen	5
3.2.1 Benutzbarkeit	5
3.2.2 Zuverlässigkeit	5
3.2.3 Effizienz	5
3.2.4 Softwarewartung	5
3.2.5 Sicherheit	5
3.2.6 Normen	5
4 Testung	5
5 Monitoring/Support bei Übergabe oder ähnliche Leistungen	5
6 Dokumentation	5
6.1 Anwenderdokumentation	5
6.2 Administratordokumentation	5
6.3 Entwicklerdokumentation	5
6.4 Weitere referenzierte Dokumente	6
7 Vorgehen (Wie?)	6
8 Entwicklungsumgebung (Womit?)	7
9 Glossar	7

1 Zielbestimmung

Für das Unternehmen MoneyGroup AG soll ein Verschlüsselungstool mit grafischer Benutzeroberfläche in Java geschrieben werden. Der Benutzer kann seine Nachricht eingeben und mit seinem Key und dem Drücken der Encrypttaste die Nachricht verschlüsseln. Anschließend kann diese verschlüsselte Nachricht über jedes handelsübliche Emailprogramm versendet werden.

1.1 Muss-Kriterien

MK-IO-01	STD-Output	Das System soll eine grafische Benutzeroberfläche besitzen
MK-IO-02	STD-Input	Der Benutzer kann in ein Textfeld seine Nachricht eingeben und durch Drücken des Encrypt-Buttons die Nachricht verschlüsseln.
MK-IO-03	STD-Input	Der Benutzer kann in ein Textfeld eine verschlüsselte Nachricht eingeben und durch Drücken des Decrypt-Buttons die Nachricht entschlüsseln
MK-BS-01	Exit Code	Durch schließen des Fensters wird das Programm beendet.
MK-SYS-01	OO-Analyse	Die Analyse des Systems soll objektorientiert erfolgen.
MK-SYS-02	UML2	Für Modellierung und Dokumentation soll UML2 genutzt werden.
MK-IMPL-01	Java Code	Die Implementierung soll in Java erfolgen.

1.2 Kann-Kriterien

KK-BS-01	Anzeige Hilfe	Der Benutzer kann, bei laufendem Programm, durch Drücken der F1-Taste eine Hilfe aufrufen
----------	------------------	---

1.3 Abgrenzungskriterien

AK-IO-01	Non-GUI	Das System soll keine grafische Bedienoberfläche haben.
AK-T-01	Testung	Das System soll keinen Usability-Test durchlaufen.

2 Produkteinsatz

2.1 Anwendungsbereich

Das Verschlüsselungstool wird von dem Unternehmen MoneyGroup AG im täglichen Geschäftsprozess benutzt. Das System dient zur Verschlüsselung der internen Kommunikation

2.2 Zielgruppen

Programm soll allen Mitarbeitern der MoneyGroup AG bereitgestellt werden.

2.3 Produktumgebung

Das System benötigt mindestens eine installierte Java Runtime ab Java-Version 1.0. Um Java einfach starten zu können, sollte die Pfad-Variable auf den bin-Ordner der Javaumgebung gesetzt sein. Es bestehen minimalste Hardwareanforderungen, sowie ein Windows- oder Linuxbetriebssystem.

2.3.1 Architektur

Es ist keine spezielle Architektur vorgesehen, es ist eine typische Java-Anwendung.

2.3.2 Technologie

Zur Darstellung der Benutzeroberfläche wird Java-AWT verwendet.

2.3.3 Komponenten

Das Projekt muss aufgrund der geringen Komplexität nicht in Komponenten zerlegt werden.

2.3.4 Schnittstellen

Die einzige Schnittstelle ist die grafische Benutzeroberfläche des Programms. Es sind keine weiteren Schnittstellen vorhergesehen.

2.4 Betriebsbedingungen

Das System ist nur für die Anwendung in den Geschäftsstellen der MoneyGroup AG vorgesehen. Hier nutzen die Mitarbeiter dieses Programm täglich auf den Arbeitsrechnern. Das Tool ist vorinstalliert und wird in regelmäßigen Abständen, von den Entwicklern aktualisiert.

3 Produktfunktionen/Anforderungen

3.1 Funktionale Anforderungen

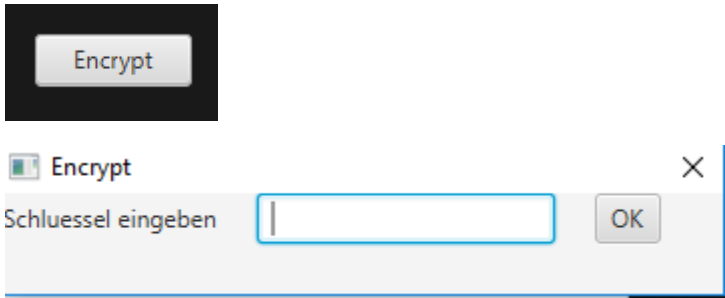
3.1.1 Beschreibung der FAs mit Rollen innerhalb der Geschäftsprozesse

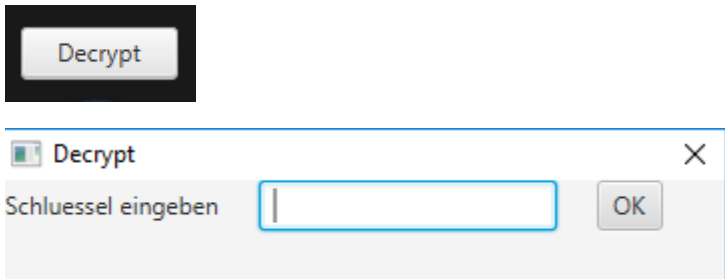
Der Benutzer hat die Möglichkeit zwischen der Encrypt-, Decrypt- und Hilfefunktion.
Als Nebenfunktionen ist der Import von Keys und Suche nach einem Empfänger gedacht.

AF-01	Encrypt-Funktion	Die zuverschlüsselnde Nachricht wird in das obere Textfeld geschrieben oder kopiert, anschließend wird diese dann durch Drücken des Encrypt-Buttons und Eingabe des eigenen Schlüssels verschlüsselt. Die verschlüsselte Nachricht wird im unteren Textfeld angezeigt.
AF-02	Decrypt-Funktion	Die zuentschlüsselnde Nachricht wird in das obere Textfeld kopiert, anschließend wird diese durch Drücken des Decrypt-Buttons und Eingabe des eigenen Schlüssels entschlüsselt. Die entschlüsselte Nachricht wird im unteren Textfeld angezeigt.
AF-03	Import Key	Der Benutzer hat die Möglichkeit seinen Public-Key anzeigen zu lassen
AF-04	User Suche	Der Benutzer hat die Möglichkeit einen Empfänger zu suchen und diesen als Empfänger auszuwählen.
AF-05	Hilfefunktion	Der Benutzer kann durch Drücken der „?“-Funktion die Anwenderdokumentation einsehen.
AF-06	Anzeige der Projektdaten	Durch Drücken des About-Buttons wird in einem neuen Fenster die Projektdaten angezeigt
AF-07	Anzeige einer Fehlermeldung	Der Benutzer wird darauf hingewiesen, dass das Programm nicht richtig benutzt wurde. Hier wird zwischen verschiedenen Fehlern unterschieden und auch gleich der Grund genannt.

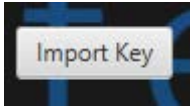
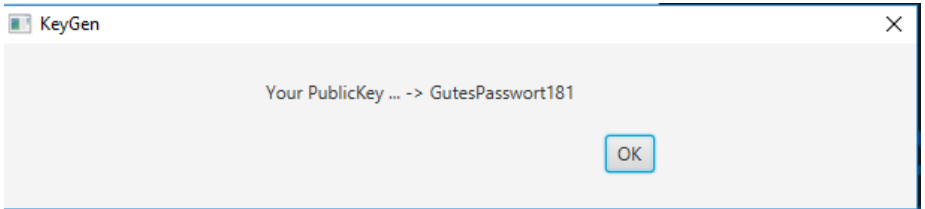
3.1.2 Aktivitäten mit Benutzerschnittstelle (UI)

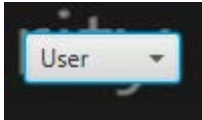
Anwendungsfall ID	AF-01
AF Name	Encrypt-Funktion
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Das Programm wurde gestartet
Auslösendes Ereignis	Durch Drücken des Encrypt Buttons wird die eingegebene Nachricht verschlüsselt. Davor wird nach dem eigenen Key gefragt.
Nachbedingung Erfolg	Die verschlüsselte Nachricht wird im unteren Textfeld angezeigt.
Nachbedingung Fehlschlag	Programm konnte nicht gestartet werden. Eigener Key wurde falsch eingegeben.
Ablauf	<ul style="list-style-type: none">- Programm öffnen- Auswahl des eigenen Users im Dropdown-Menü- beim erstmaligen Verwenden, eigenen Key einsehen

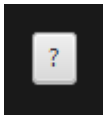
	<ul style="list-style-type: none"> - Eingabe der Nachricht ins obere Textfeld - Drücken des Encrypt-Buttons - Eingabe des eigenen Keys - Ausgabe der verschlüsselten Nachricht im unteren Textfeld
Benutzerschnittstelle	

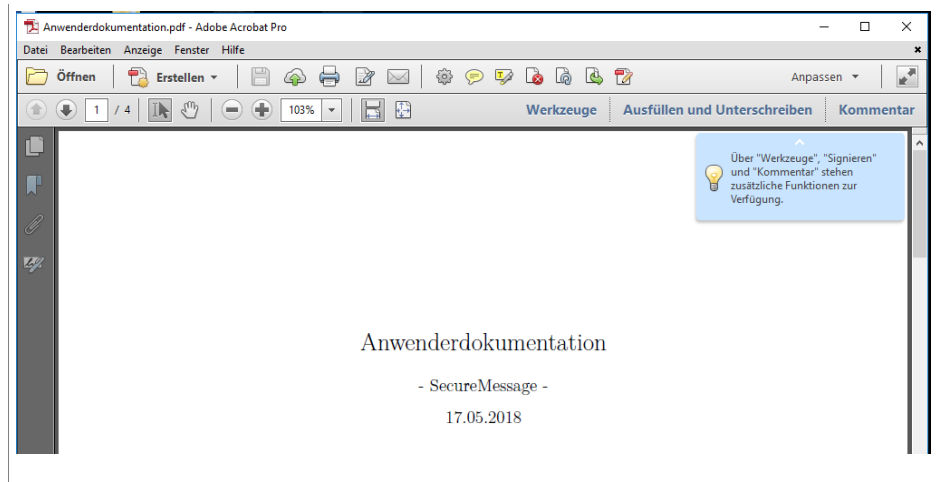
Anwendungsfall ID	AF-02
AF Name	Decrypt-Funktion
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Das Programm wurde gestartet
Auslösendes Ereignis	Durch Drücken des Decrypt-Buttons wird die eingegebene und verschlüsselte Nachricht entschlüsselt. Davor wird nach dem Key des Absenders gefragt.
Nachbedingung Erfolg	Die verschlüsselte Nachricht wird entschlüsselt im unteren Textfeld angezeigt.
Nachbedingung Fehlschlag	Programm konnte nicht gestartet werden. Eigener Key wurde falsch eingegeben.
Ablauf	<ul style="list-style-type: none"> - Programm öffnen - Den Absender der Nachricht im Dropdown-Menü auswählen - Den Key des Absenders einsehen via Import-Key - Eingabe der verschlüsselten Nachricht ins obere Textfeld - Drücken des Decrypt-Buttons - Eingabe des Absender-Keys - Ausgabe der entschlüsselten Nachricht im unteren Textfeld
Benutzerschnittstelle	


Anwendungsfall ID	AF-03
AF Name	Import Privat-Key
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Das Programm wurde gestartet
Auslösendes Ereignis	Durch Drücken des Buttons „Import Privat-Key“ öffnet sich ein Fenster indem man den Key des ausgewählten Users sieht.
Nachbedingung Erfolg	Der Privat-Key wird erfolgreich angezeigt.
Nachbedingung Fehlschlag	Der Key konnte nicht importiert werden.

Ablauf	<ul style="list-style-type: none"> - Programm öffnen - Drücken des „Import -Key“-Buttons - Neues Fenster wird dargestellt, hier wird der Key des zuvor ausgewählten Users angezeigt
Benutzerschnittstelle	 

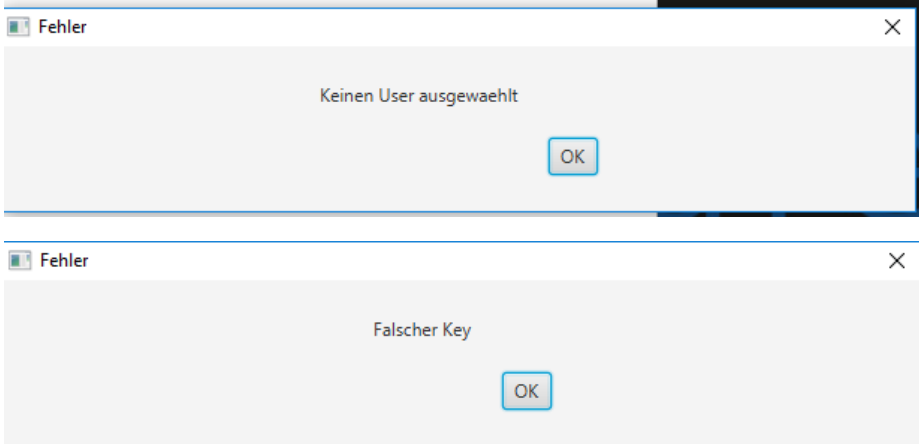
Anwendungsfall ID	AF-04
AF Name	User Suche
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Das Programm wurde gestartet
Auslösendes Ereignis	Der Benutzer wählt über das Dropdown-Menü den gewünschten User aus.
Nachbedingung Erfolg	Der User wurde erfolgreich ausgewählt. Der Key des ausgewählten Users wurde importiert und kann über „Import-Key“ eingesehen werden.
Nachbedingung Fehlschlag	Der Key konnte nicht importiert werden.
Ablauf	<ul style="list-style-type: none"> - Programm öffnen - Auswahl des Empfängers über das Dropdown-Menü - Key wurde erfolgreich importiert
Benutzerschnittstelle	

Anwendungsfall ID	AF-05
AF Name	Hilfefunktion
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Das Programm wurde gestartet
Auslösendes Ereignis	Durch Drücken der „?“-Taste wird die Anwenderdokumentation angezeigt.
Nachbedingung Erfolg	Die Anwenderdokumentation wird erfolgreich in einem extra Fenster angezeigt.
Nachbedingung Fehlschlag	Die Anwenderdokumentation wird nicht angezeigt.
Ablauf	<ul style="list-style-type: none"> - Programm öffnen - „?“-Taste drücken - Anzeige der Anwenderdokumentation
Benutzerschnittstelle	



Anwendungsfall ID	AF-06
AF Name	Anzeige der Projektdaten
Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Das Programm wurde gestartet
Auslösendes Ereignis	Durch Drücken des Buttons „About“ öffnet sich ein Fenster indem man die Projektdaten einsehen kann.
Nachbedingung Erfolg	Die Projektdaten werden angezeigt.
Nachbedingung Fehlschlag	Die Projektdaten werden nicht angezeigt
Ablauf	<ul style="list-style-type: none"> - Programm öffnen - Drücken des „About“-Buttons - Die Projektdaten werden in einem Fenster angezeigt
Benutzerschnittstelle	

Anwendungsfall ID	AF-07
AF Name	Anzeigen der Fehlermeldung

Akteur	am Betriebssystem angemeldeter Nutzer
Vorbedingung	Das Programm wurde gestartet
Auslösendes Ereignis	Keys wurden nicht importiert.
Nachbedingung Erfolg	Eine Fehlermeldung wird angezeigt.
Nachbedingung Fehlschlag	Die Fehlermeldung wird nicht angezeigt.
Ablauf	<ul style="list-style-type: none"> - Programm öffnen - Der Key wurde falsch eingegeben - Es wurde kein Empfänger ausgewählt - Ausgabe der Fehlermeldung
Benutzerschnittstelle	

3.1.3 Fachliches Klassendiagramm („*domain model*“) / Produktdaten

Für die Anwendung sind keine Daten dauerhaft zu speichern.

3.2 Nichtfunktionale Anforderungen

3.2.1 Benutzbarkeit

Das Programm sollte einfach und intuitiv gestaltet sein.

3.2.2 Zuverlässigkeit

Die Anwendung muss jederzeit tadellos funktionieren da es für die tägliche Kommunikation unter den Mitarbeitern verwendet wird.

3.2.3 Effizienz

Die Effizienz wird durch einfache Bedienung und schlanke Software erreicht.

3.2.4 Softwarewartung

Das Programm wird durch den Hersteller gewartet.

3.2.5 Sicherheit

Das Programm muss ein ausreichendes Verschlüsselungsprotokoll verwenden.

3.2.6 Normen

NF-B1	Benutzung	Die Anwendung soll als grafische Benutzeroberfläche dargestellt werden.
NF-E1	Effizienz	Die Ausgabe der Nachrichten soll unmittelbar erfolgen.
NF-W1	Softwarewartung	Es liegen keine Anforderungen vor.
NF-S1	Sicherheit	Eine ausreichende Verschlüsselung muss gewährleistet sein.
NF-N1	Normen	Die Anwendung braucht keine besonderen Normen zu erfüllen.

4 Testung

Es wird ein ausreichender Funktionstest für die obengenannten Anwendungsfälle durchgeführt.

5 Monitoring/Support bei Übergabe oder ähnliche Leistungen

Im Rahmen unserer Anforderungen an uns Selbst, stellen wir einen umfangreichen Support für die Anwendung bereit:

- Bereitstellung des Repositories
- individueller Support und Schulung der Mitarbeiter
- Rufbereitschaft 8x5 per E-Mail.

6 Dokumentation

6.1 Anwenderdokumentation

Die Anwenderdokumentation wird als typische „readme.txt“-Datei in deutscher Sprache im Repository zur Verfügung gestellt.

6.2 Administratordokumentation

Eine Administratordokumentation ist nicht vorgesehen.

6.3 Entwicklerdokumentation

Die Entwicklerdokumentation wird als Ent_Doku.txt in deutscher Sprache im Repository zur Verfügung gestellt.

6.4 Weitere referenzierte Dokumente

Das Pflichtenheft wurde mit Bezug auf das bereitgestellte Lastenheft erstellt.

7 Vorgehen (Wie?)

Für das Programm wird ein Prototyp erstellt, der nach Absprache mit dem Kunden noch geändert werden kann. Der letzte Prototyp gilt als Release Candidate.

Meilensteine sind:

Datum	Meilenstein
27.05.2018	Auftakttreffen
30.05.2018	Projektplan und Pflichtenheft
05.06.2018	Erstellen der Entwicklungsumgebung
15.06.2018	Prototyp
16.06.2018	Funktionstest
01.07.2018	Release Candidate
10.07.2018	Übergabe

Die Fortschrittskontrolle erfolgt anhand folgender Indikatoren:

Indikator		Auftakt-treffen	Projektplan und Pflichtenheft	Erstellen der Entwicklungs-umgebung	Prototyp	Funktions-test	Release Candidate	Übergabe
Pflichtenheft [% erledigte Gliederungs-Punkte]	Soll		100					
	Ist							
Umgebung	Soll			2				

[Tools]	Ist							
Diagramme [Anzahl]	Soll		2					
	Ist							
Quellcode [LOC]	Soll				600			
	Ist							
Verhältnis LOC/ Kommentare	Soll				4:1			
	Ist							
Tests [Fälle/ Methoden]	Soll					1		
	Ist							
Anwenderdoku [Wörter]	Soll					500		
	Ist							
Entwicklerdoku [Wörter]	Soll					500		
	Ist							
Release [Artefakte]	Soll						5	5
	Ist							

8 Entwicklungsumgebung (Womit?)

Für die Entwicklung der Anwendung wird Eclipse und der aktuelle Java Compiler verwendet. Der Quellcode wird ausreichend kommentiert.

9 Glossar

<i>Public-Key:</i>	<i>Öffentlicher Schlüssel</i>
<i>Encrypt:</i>	<i>Verschlüsseln</i>
<i>Decrypt:</i>	<i>Entschlüsseln</i>
<i>Repositories:</i>	<i>Quellen</i>
<i>Release Candidate:</i>	<i>Entwicklungsstadium vor Release</i>
<i>Release</i>	<i>Fertiges Programm zum Stichtag</i>